

Article

Proposing a Small-Scale Digital Twin Implementation Framework for Manufacturing from a Systems Perspective

Jonatan H. Loaiza , Robert J. Cloutier  and Kari Lippert

Department of Systems Engineering, University of South Alabama, Mobile, AL 36688, USA

* Correspondence: jhl1921@jagmail.southalabama.edu

Abstract: Due to the fourth industrial revolution, manufacturing companies are looking to implement digital twins in their factories to be more competitive. However, the implementation of digital twins in manufacturing systems is a complex task. Factories need a framework that can guide them in the development of digital twins. Hence, this article proposes a small-scale digital twin implementation framework for manufacturing systems. To build this framework, the authors gathered several concepts from the literature and designed a digital twin subsystem model using a model-based systems engineering (MBSE) approach and the systems engineering “Vee” model. The systems modelling defines the digital twin components, functionalities, and structure. The authors distribute most of these concepts throughout the framework configuration and some concepts next to this general configuration. This configuration presents three spaces: physical, virtual, and information. The physical space presents a physical layer and a perception layer. The information space has a single layer called middleware. Finally, the virtual space presents two layers: application and model. In addition to these layers, this framework includes other concepts such as digital thread, data, ontology, and enabling technologies. This framework could help researchers and practitioners to learn more about digital twins and apply it to different domains.

Keywords: digital twin; manufacturing systems; model-based systems engineering; Vee model; framework



Citation: Loaiza, J.H.; Cloutier, R.J.; Lippert, K. Proposing a Small-Scale Digital Twin Implementation Framework for Manufacturing from a Systems Perspective. *Systems* **2023**, *11*, 41. <https://doi.org/10.3390/systems11010041>

Academic Editor: Mark Austin

Received: 7 December 2022

Revised: 5 January 2023

Accepted: 10 January 2023

Published: 11 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Manufacturing companies must follow the Industry 4.0 trends to survive in a highly competitive market. The fourth industrial revolution uses modern technology to transform manufacturing and make it smart [1,2]. Smart factories digitize the physical layout, business processes, and products, and integrate them in the digital world [3,4]. This improves the efficiency of processes [5], create better products [6], and enables data-driven decisions [7]. Smart manufacturing encompasses recent technologies such as the Internet of Things (IoT), artificial intelligence (AI), augmented reality (AR), big data analytics, and so on to enable the digital transformation of factories [8–11]. The integration of some of these technologies result in the development of digital twins [12,13].

Digital twin (DT) is a tool of Industry 4.0 that helps factories achieve digital transformation. DT mirrors the physical system in the virtual world. It has a continuous communication with its counterpart in the real world through an information channel [14]. DT uses several components such as sensors, actuators, software, databases, etc. DT collects, transfers, and stores data [15]. Factories could use digital twins for monitoring, data analysis, product development, and so on. In manufacturing, DT find its application in, for example, predictive maintenance, process planning, product, and factory design [16,17].

Nevertheless, the implementation of DT brings challenges to factories. DT implementation represents a huge investment of physical and human resources. Some researchers believe that a digital twin must use sophisticated technologies such as artificial intelligence and machine learning to operate. Furthermore, DT is a complex system that includes many

concepts and processes. Researchers have different definitions for a digital twin and their capabilities for manufacturing. Finally, there are not clear examples of a digital twin for manufacturing because of data property. Companies do not share their DT models to the public [18–20].

Therefore, factories need a framework that can guide them in the development of digital twins. Frameworks help to study a new concept or phenomena such as digital twins. Frameworks are an effective way to gather main concepts of the topic of interest and integrate them in a defined structure [21]. Even though there are many concepts related to DT in the current literature review, a framework can help to explain and resolve conflict between DT concepts. Frameworks can have a specific application that differentiate it from other frameworks. A digital twin implementation framework could enable manufacturing systems to implement digital twins in a secure, easy, and fast manner.

Existing DT literature present frameworks and related works for different manufacturing processes, and types with distinct characteristics, methods, and objectives. For instance, there are frameworks that focus on simulation and optimization of digital twins. Guo et al. [22] proposed a DT framework to optimize factory layout designs and solve hidden design flaws. Zhang et al. [23] proposed a DT framework using discrete-event simulation models for production planning and layout design. Marmolejo-Saucedo [24] developed a DT framework using optimization models for large-scale problems in supply chains. This framework considers the use of big data analytics but does not include artificial intelligence. Some focuses on collecting and enabling data throughout the digital twin models. The Kumbhar et al. [25] framework proposed a DT data-driven framework for detection and diagnostics of flaws. It executes a DT simulation to identify bottlenecks and improve bottlenecks throughput in complex manufacturing systems. Some frameworks do not focus on modelling the digital twin. Friederich et al. [26] focused on developing a framework to improve the simulation functionality of DT using machine learning and process mining techniques. Some researchers suggest a standardized framework. Shao and Helu [27] developed the scope and requirements for a generalized DT framework. However, their proposed framework just focuses on the use of DT in factories, not on the implementation. The International Organization for Standardization (ISO) developed the ISO 23247 which presents an overview, definitions, principles, and requirements for a DT framework [28]. Nonetheless, a generalized framework may be incompatible for factories with different context or applications.

Consequently, after analyzing the digital twin (DT) concept in a previous article [29], the next research step is to propose a small-scale digital twin implementation framework for a manufacturing system. This framework aims to design a digital twin for discrete manufacturing processes. Discrete manufacturing put together tangible components into a final product in an assembly line [30]. This work focuses on small size manufacturing companies that do not have resources such as sophisticated technology and skilled workers. It provides a comprehensible step-by-step implementation process. Furthermore, the authors used a methodology with a systems perspective to build this framework. This article gathers relevant digital twin concepts from the literature and models a digital twin for manufacturing systems. This model provides a high-level perspective about the development of digital twins and posterior implementation in factories. It defines essential digital twin functionalities, components, and structure. The system's modelling follows a model-based systems engineering (MBSE) approach and uses the "Vee" model developed by the U.S. Federal Highway Administration [31], which are based on the ISO 15288 systems and software engineering - system life cycle processes [32].

This study presents some limitations and assumptions in terms of scope to build a digital twin implementation framework for manufacturing systems. This work looks to develop small-scale digital twins from a systems perspective. This study does not focus on a specific technology or functionality such as cloud computing or artificial intelligence, nor does this framework develop a digital twin for a specific task or activity such as model fidelity design or product development. This study generalizes the development of digital

twin models in a manufacturing domain. The concepts presented in this article are not comprehensive. Nevertheless, they could help practitioners to implement digital twins in their factories.

This article has the following structure. Section 2 presents a literature review, including the most relevant topics for the development of this article. Section 3 models a digital twin subsystem of a manufacturing system. In Section 4, the authors propose a small-scale digital twin implementation framework for manufacturing systems. Section 5 presents a discussion about the digital twin model and proposed framework. Finally, Section 6 presents the conclusions and proposes future research on digital twin technology for manufacturing systems.

2. Literature Review

2.1. Digital Twin Components

Digital Twin has three main components: products in the physical space, products in the virtual space, and the connections of data and information that unifies both spaces [14]. Currently, a digital twin presents more behavioral characteristics due to the knowledge of information technology and operations technology. These technologies allow DT to model processes, machines, products, and so on, and perform specific functionalities such as testing a product's performance capabilities. Figure 1 shows the original concept of a digital twin as an information mirroring model that display the product in the virtual space [33].

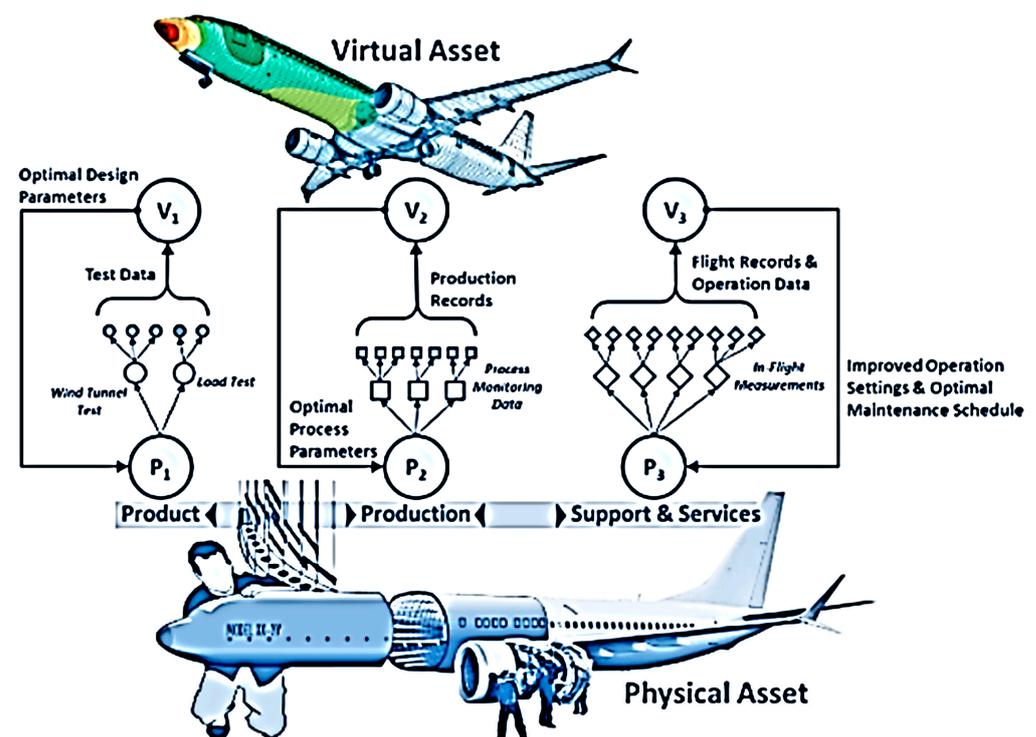


Figure 1. Representation of the Digital Twin Concept [33]. Reproduced with permission from AIAA Digital Engineering Integration Committee. “Digital Twin: Definition and Value—An AIAA and AIA Position Paper.” American Institute of Aeronautics and Astronautics (AIAA), 2020.

Loaiza and Cloutier’s article [29] studied the selection of the digital twin components for a manufacturing system. Each digital twin configuration space presents different components or concepts. These components are necessary to enable the digital twin capabilities. The authors used the conceptagon to distribute them into internal and external components. The internal components belong to the digital twin system. These components are RFID readers, RFID tags, PLC, transducers, software, electric actuators, switches, MEMS

sensors, and RFID transceivers. The external components belong to the manufacturing system. These components are artificial intelligence, digital thread, cloud computing, and smart machines. Moreover, the conceptagon describes the digital twin's behavior and functionalities such as data collection, data analysis in real-time, simulation of objects, and so on. Finally, it also considers the relationship between the system's components and how they transform inputs into outputs. Digital twin inputs are images, procedures, data, algorithms, and energy. The outputs are images, results, data, diagnosis, and energy. The authors concluded that the digital twin's goal is to give information and data that support manufacturing system's operations.

Loaiza and Cloutier's article [29] also presents a systemigram, which is a systems thinking tool. This tool studies the relationship between a digital twin, manufacturing system, and other concepts such as system readiness level, systems engineering, and digital transformation for the DT implementation. It provides an insight to several components within different concepts. The systemigram shows how data flows from the physical system to the digital twin, which send feedback to the system closing the loop.

2.2. Characteristics of Digital Twin Technology

According to DI SPRING [34], a company that promotes the Industry 4.0, DT present the following characteristics: connectivity, homogenization, smart programmability, traceability, and modularity. These characteristics make DT different from other technologies. The connectivity of DT is one of its most distinguishing features. It is the foundation of DT to connect the real world with the virtual world. It is crucial for the development of DT functionalities. Connectivity is a feature that will change over time as DT evolves. Homogenization is a feature that allows DT to collect and share data with other digital platforms. DT gathers information from the physical source to mirror it into a virtual model. Homogenization brings benefits such as low-cost ways to manage data and enhances the user experience to collaborate on a single digital source. DT is a smart technology that can program its functions automatically. The characteristic of smart programmability makes it possible to control physical objects. DT uses sensors, actuators, and artificial intelligence. DT can manage factory's processes and program them according to learning experience. This gives introduction to servitization in manufacturing. Smart programmability improves DT services to meet customer's needs. The DT characteristic of traceability enables it to perform functionalities such as simulation. DT can check past information of processes or products for diagnosis due to its digital thread. Digital tread implements traceability from the system requirements to the design, production, distribution, and disposal stage of a system of interest. It improves the communication and relationship of the DT with other systems. Finally, modularity is a system characteristic adopted by DT to separate and reorganize components. It gives flexibility and variety to DT models, reducing the complexity of systems by arranging a system in modules. The complexity of system's components is not visible at this level of model abstraction. The benefit of modularity is to understand and look on the right problem.

Barricelli et al. [35] believe that a DT has the following characteristics: connection with multiple devices, a high amount of data storage, and the ability to make smart decisions. The physical and virtual spaces should have a seamless connection to Internet and other networking devices to allow data sharing. This connection sets up direct and indirect communication through physical devices and cloud computing. The DT process of communication links physical objects and the DT, the DT and other DTs in the surrounding environment, and the DT and domain experts. DT is also capable to gather distinct kinds of data and organize them in categories, concepts, areas, etc. It understands data sources through ontologies. It shows data properties and the relationship between them. A digital twin should implement a database or storage system to save historical data and current data. Data are key to the performance of DT functionalities. Finally, a digital twin has the characteristic of making predictions, prescriptions, and descriptions of situations, processes, tasks, status, etc. It uses artificial intelligence that allows DT to learn capabilities. This

is possible due to algorithms that work as a virtual cognitive brain that collects data and makes intelligent decisions.

2.3. The “Vee” Model of the Systems Engineering Process

Systems engineering is a disciplined approach that focuses on the design, implementation, operation, and retirement of systems [36]. The IEEE 15288, Systems and Software Engineering—System Life Cycle Processes, describe the processes to manage systems over their life cycles [32]. The “Vee” model is one of several system models that describe these system processes. The “Vee” model has a V-shape describing a system’s development from left to right. It is an iterative model that improves the system until its maturity. The left-side of the “Vee” starts in the abstraction level where the system is decomposed into several components. The right-side assemble these components to develop the final product [37].

Systems engineering literature presents different variants of the “Vee” Model with different terminology and levels of decomposition. However, these models have activities in common throughout the system’s development [38]. In general, the left-side of the “Vee” model presents the system’s definition and planning stage, and the right-side the system’s integration, test, and operational stage. The first activity in the left-side describes the system stakeholder’s needs. The second activity transforms these needs into the system’s requirements. The next activity decomposes the system in a lower level presenting a high-level architecture. The next step is a more detailed architecture of the system design. The bottom activity of the “Vee” model is the execution of the plan. It looks to create the system’s components. Then, going up to the right-side of the “Vee” model, the next activity aims to integrate the components and verify and validate that the system meets the stakeholder’s needs. Finally, the last activity is when users operate and maintain the developed system though several iterations of the “Vee” model. The following iterations use feedback such as data, tests, updates, and so on, to improve the system until its retirement or replacement [39].

2.4. Model-Based Systems Engineering

The model-based systems engineering (MBSE) approach is a graphical modelling language that enables the design of complex systems such as a digital twin system. This approach aims to create systems engineering domain models to save and exchange information different to the document-centric model. It uses computer modelling to define systems based on properties, specifications, and future behavior. Systems modelling are helpful for architectural design processes. It also supports the development and procurement of requirements in the system, subsystems, and components. The MBSE goal is to give precision, consistency, traceability, and integration to the entire system lifecycle [40,41].

According to Delligatti [42], MBSE looks for the integration, coherence, and consistency of system activities in one single model. This is an advantage over the traditional document-based approach. The MBSE approach aims to generate documents automatically based on the information provided to the system model. The benefits are noticeable when the designer wants to change a requirement or update it. Changes in requirements affect the entire system model. The MBSE approach keeps track of these changes and updates the system automatically. There is no need to examine and update all the models and documents that were affected by that change as in the traditional document-based approach. The MBSE approach supports the application of the systems engineering “Vee” model for project development. The “Vee” model presents the system requirements from the stakeholder’s needs, the system design, analysis, integration and test, and the verification and validation process. The Vee model activities begin in the conceptual design and finishes with the actual operation of the project.

2.5. Smart Manufacturing

The trend of manufacturing is to become smart due to IoT devices, and software that improves management decisions. Smart manufacturing (SM) involves and studies all the stages of production from suppliers to customers. Agencies such as the Department of Defense and the Department of Energy use this term to describe the use of intelligence to produce better products. Smart manufacturing implements intelligence along the supply chain manufacturing. It gives users a holistic perspective to study, plan, and manage manufacturing processes. This is possible through the implementation of IoT devices, and development of data analytics, modelling and simulation [43].

The National Institute of Standards and Technology definition of SM is that it integrates all the components of a manufacturing system. SM processes meet supply chain manufacturing needs in real-time such as factory conditions, customer needs, and supply networks [44]. Digital twins are a part of smart manufacturing. It connects physical assets to an industrial network and models them in the virtual space. It provides SM with tools to simulate, improve models, and predict physical objects status in real-time. DT is not the only technology that makes manufacturing smart. Big data, artificial intelligence, and cloud computing, along with DT, work together to enable automated processes and activities. This is the case of DT technology. It needs of other IoT technologies to perform its functionalities.

There are three applications of smart manufacturing. The first one is smart production. This application describes production with augmented intelligence to manufacture smart end-products. SM are capable of making production more flexible and efficient. It improves the human-machine interface towards collaboration. The second application is the smart production network. It puts SM in a bigger system perspective. It considers the integration of other systems in the supply chain management. The goal is to create a big manufacturing network that helps each other to satisfy the constantly changing customer needs. This network will ease production planning and create automated processes at distinct levels in the SM. The upsides are more revenue, production processes that respect the environment, and a socially responsible factory. The last one is mass personalization, which means that SM will focus on customized production. This changes mass production for a personalized one that allows users to create their own end-product [4].

2.6. Conceptual Framework

A conceptual framework is an analytical tool that studies different concepts. It allows researchers to make comparisons and organize ideas. It not only gathers concepts, but also integrates them into one single structure. The goal is to find factors, attributes, variables, behavior, processes, and so on that describe the new concept. Some researchers could mistake conceptual frameworks with conceptual models. The first one considers concepts alone. The second one considers factors and variables. It presents an approach to interpret the real world. However, it does not study cause-and-effect relationships. It helps to understand new concepts [45].

According to Levering [46], conceptual frameworks are a good start to explain a concept or phenomena. They allow problems to be understood, but cannot determine the specific outcome as quantitative models. Nevertheless, they can solve a problem based on external concepts or factors that are interrelated. Researchers follow a qualitative analysis process to develop conceptual frameworks. According to Jabareen [21], conceptual frameworks connect several concepts in a network to investigate a phenomena. They simplify ideas and organize them in a way that is easy to apply. It is the product of a qualitative process that explores theorization. It gathers several theories to build a new concept. A concept presents components which define the concept itself. Hence, these components are not separable, heterogeneous, and endo-consistent. These concepts have a background of other concepts. All these components and concepts form the conceptual framework of the new concept. Conceptual frameworks are ontology-, epistemology-, and methodology-based. Concepts that are part of a framework have an ontological and

epistemological structure or nature. The ontological nature defines concepts or things in the real world. The epistemological nature describes these things or concepts in an abstract or ideal world. The methodology explains how to build the framework and evaluates its contribution to the real world.

The conceptual framework analysis technique involves the research and analysis of concepts relevant to the new topic. It is a grounded theory technique that looks to find phenomenon or events, patterns, and relationships in theory. The selection of concepts is based on the number of occurrences in a text, their meaning, and importance. All of them are part of the new conceptual or theoretical framework. It is critical to evaluate and select data relevant to the new concept or phenomena. Data could come from several sources which are part of different disciplines. Hence, conceptual frameworks present a multidisciplinary approach to analyze data. The conceptual framework analysis is an interactive process that compares concepts and data, continuously. This process manages emerging theory based on the conceptual level and scope. The conceptual framework analysis presents the following process [21]:

1. Mapping the selected data sources.
2. Extensive reading and categorizing of the selected data.
3. Identifying and naming concepts.
4. Deconstructing and categorizing the concepts.
5. Integrating concepts.
6. Synthesis, resynthesis, and making it all make sense.
7. Validating the conceptual framework.
8. Rethinking the conceptual framework.

3. Modelling a Digital Twin for Manufacturing Systems

This section uses the systems engineering “Vee” model and a model-based systems engineering (MBSE) approach to design complex systems such as a small-scale digital twin subsystem of a manufacturing system. The system of interest (SOI) for this study is a manufacturing system with the digital twin as a subsystem. The authors use a MBSE tool such as Astah SysML for the system modelling. They also use the “Vee” model activities presented by the Federal Highway Administration (FHWA) [31] to develop the digital twin subsystem. Figure 2 shows the FHWA’s “Vee” model.

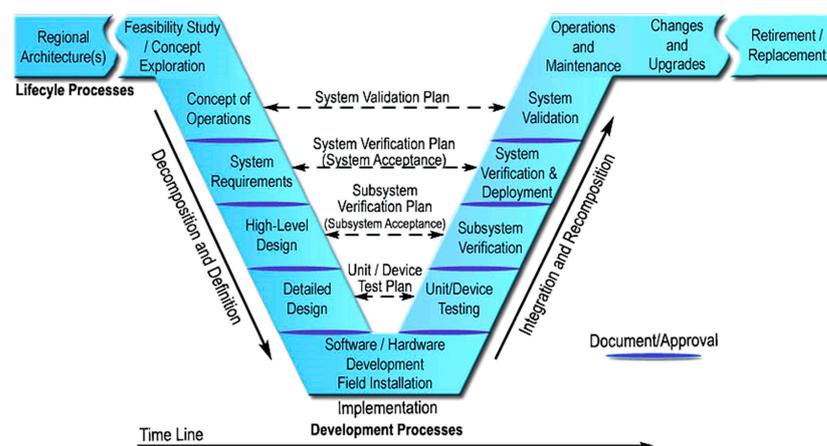


Figure 2. “Vee” model of the Systems Engineering Process [31].

In this case, the only interest of this article is the left-side of the “Vee” model. This side presents some crucial activities for the digital twin planning and design. These activities or steps are the concept of operations, system requirements, high-level design, and detailed design. This article employs a MBSE approach to perform and integrate all these activities. MBSE allows the creation, visualization, and traceability of each activity throughout the entire system.

3.1. Concept of Operations

This section defines the system’s concept of operations (CONOPS). The CONOPS is a document where stakeholders define the system needs and main operational goal from a high-level or systems perspective. This document helps to define the system requirements [31]. The system’s modelling starts with the operational need or top-level use case. The manufacturing system (MS) operational need is the implementation of digital twin technology in the manufacturing processes. To meet this goal, the system’s architects identify the MS stakeholders and develop a context diagram. These stakeholders interact with the MS as described in the use cases. Then, the system’s architects describe the system’s top-level use case which is the major usage scenario for the MS. Finally, it presents the concepts of operations and system domain which characterize the system needs.

3.1.1. Top-Level Use Case

Use cases are actions or events that define the interactions between an agent and a system to achieve a goal [42]. The system of interest (SOI) considers the digital twin technology implementation as a top-level use case. The system interacts with operators and the physical factory. The system needs operators to implement digital twin technology in the factory. These operators could be programmers, engineers, data analysts, and so on. Figure 3 shows the system’s top-level use case.

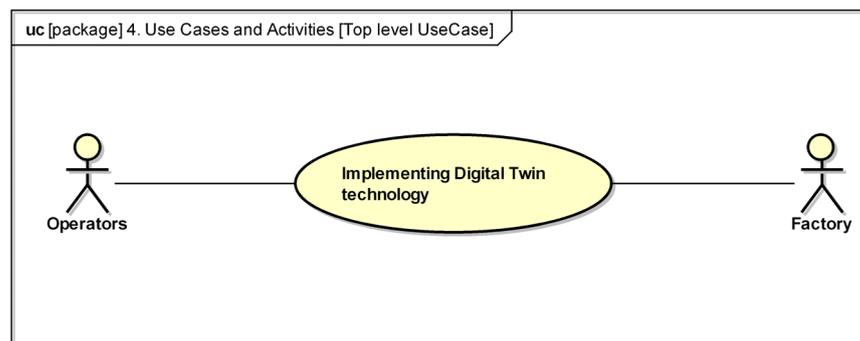


Figure 3. Top-Level Use Case.

3.1.2. Stakeholders

The MS stakeholders are all parts interested in the system because the system satisfies their operational need. The stakeholders’ identification is important to define the system’s requirements. The system’s life cycle decides who the SOI stakeholders are [31,42]. This system presents two groups of stakeholders: active and passive. Figure 4 shows the stakeholder’s diagram.

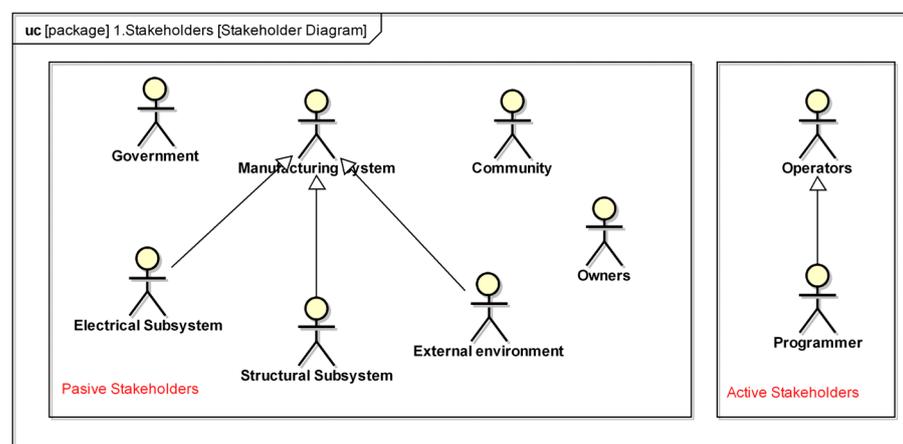


Figure 4. Stakeholder’s Diagram.

Active stakeholders are those who have a continuous participation with the SOI. They provide inputs and get outputs from the SOI. The active stakeholders for the digital twin subsystem of a manufacturing system are the following:

- **Operators/Programmer:** The MS needs operators to work on the DT subsystem and perform operational tasks such as simulating, monitoring, and controlling. Operators are key elements in the DT subsystem's life cycle from its conception to its retirement or replacement. In return, operators receive a salary for worked hours.

Passive stakeholders do not have a continuous interaction with the system. This does not mean they are not interested in the system. They are just not active participators in the system's lifecycle. The passive stakeholders are the following:

- **Owners:** The MS needs owners to put money or capital to develop new projects such as the digital twin implementation on the factory operations. Owners need to invest in the factory's structure, machines, equipment, material, labor costs, and so on.
- **External Environment:** The MS shall be responsible with its environment because the system gets energy and natural resources from it. Therefore, the system must be careful with waste emissions to the environment.
- **Electrical Subsystem:** The electrical energy allows the use of machines and other equipment, as well as the factory lighting.
- **Structural Subsystem:** The MS uses the factory facilities as its infrastructure to manage the business from the materials reception to the delivery of products. This subsystem is the physical space of the digital twin subsystem.
- **Community:** The community has similar interests in the SOI as the environment. Hence, the system shall be responsible with the community. The community shall accept the factory and support its operations. In return, the factory provides jobs, products, and services to the society.
- **Government:** The government regulates the SOI development and operations. It defines and enforces laws, norms, incentives, rules, regulations, and so. The factory will retribute the government by obeying the law and paying taxes.

Figure 5 presents the system of interest context diagram that shows the interaction of stakeholders with the manufacturing system:

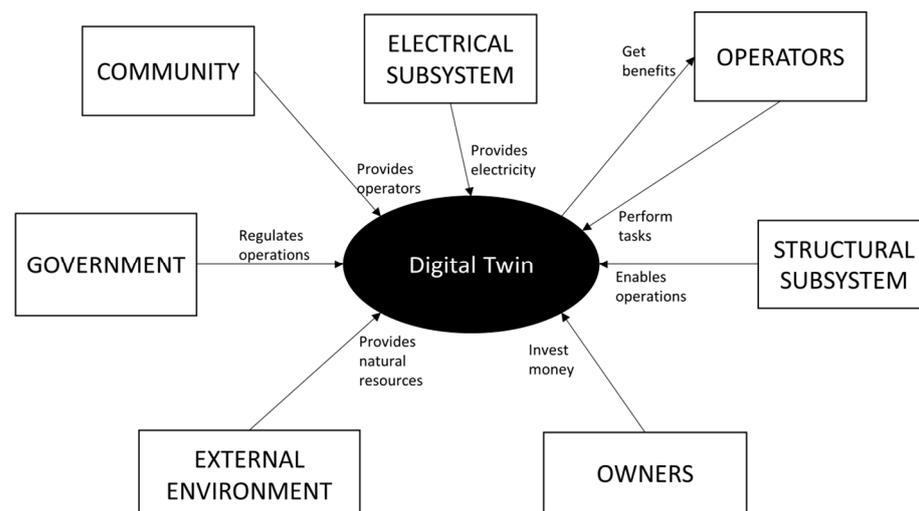


Figure 5. Context Diagram.

3.1.3. CONOPS and System Domain

The CONOPS and system domain describes the system's characteristics from a user perspective. A MS looks to implement digital twin technology in its regular operations. Digital twin is a smart technology that twin physical objects in the virtual world. DT simulates real-time data to make decisions. DT optimizes processes and objects in the

virtual world and applies the results in the real world. DT capabilities are monitoring, simulating, and controlling manufacturing processes. DT monitors production processes in real time. DT simulates ‘what-if’ scenarios to prevent or reduce risks and improve processes. Finally, DT controls the physical system to apply the simulation results [16,47]. Figure 6 shows the digital twin’s CONOPS diagram.

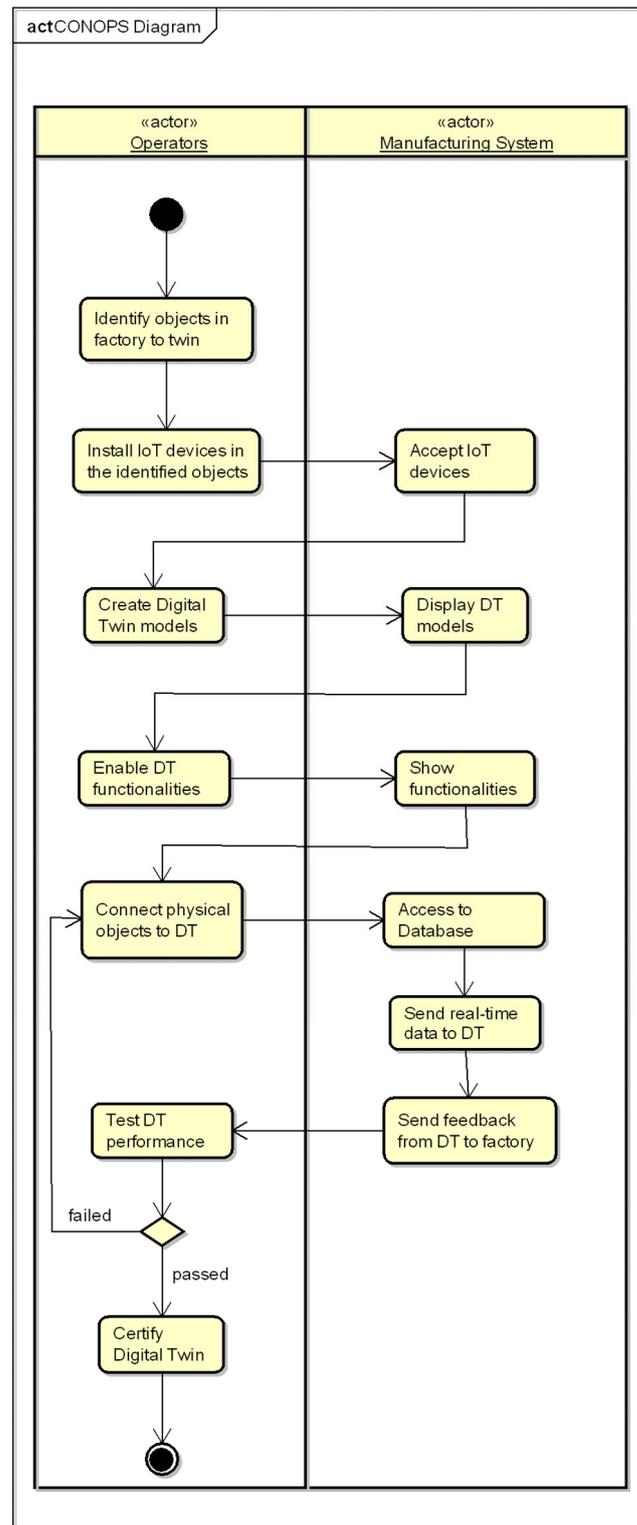


Figure 6. CONOPS diagram.

The CONOPS diagram shows the activities to implement DT in a factory. First, MS operators identify physical objects in the factory to twin in the virtual world. Then, they install IoT devices, such as sensors and actuators, in the factory to collect data from and operate the objects. The third step is to create digital twin models in the virtual world. The next step is to enable digital twin functionalities such as simulation, monitoring, and controlling objects. MS workers model the factory processes and add functionalities to the DT models. Then, the operators connect the DT models to the factory using real-time data. Finally, they test the DT performance, and certify it. Therefore, the high-level tasks involved in the CONOPS for the implementation of a digital twin for a manufacturing system are to install IoT devices, develop digital twin models, enable digital twin functionalities, and connect the factory to the digital twin. The authors describe these tasks in the use cases lines below.

3.2. System Requirements

Requirements describe the necessary operational outcomes to fulfill an operational need. They define the system's functions and features. From a high-level perspective, requirements focus more on what the system should do than how to do it. They do not get into details [31]. This is the case of this system of interest which needs requirements to fulfill its use cases. The SOI operational need is to implement digital twin technology in the manufacturing system. Therefore, the system of interest needs three high-level requirements such as resources, technology, and digital transformation. Figure 7 shows the system's high-level requirements in a SysML requirements diagram. Finally, this section divides the requirements into functional, non-functional, and interface requirements. Table 1 shows some of the digital twin requirements for a manufacturing system.

3.3. High-Level Design

After defining the system requirements, the system's architects describe high-level use cases and design the system's logical architecture. This section shows use cases with several tasks that enables the realization of the system's top-level use case [31]. It also shows an overall system's architectural design to satisfy the system requirements. This architectural design decomposes the system into subsystems and components [42].

Table 1. Some of the digital twin requirements for a manufacturing system.

ID Req.	Requirement	Description	Type of Requirement
1.1.1	Physical	The MS shall use physical resources to support the creation of digital twin models.	Non-Functional
1.1.1.1	Infrastructure	The MS shall use a physical infrastructure to operate the digital twin's "hardware."	Interface
1.1.2	Human	The MS owners shall hire employees to implement digital twin in the manufacturing system.	Non-Functional
1.2.1	Smart Machines	The operators shall install smart machines that works in a network setting and makes automated decisions.	Functional
1.2.2.3	Data Visualization	The DT software should use maps, graphics, dashboards, and so on to represent data and information.	Functional
1.3.1.1	Digital Thread	The MS workers shall create a digital thread to connect the physical space to the virtual space.	Functional
1.3.2.3	Objects	The MS workers shall digitize physical objects in the system according to the DT scope.	Functional

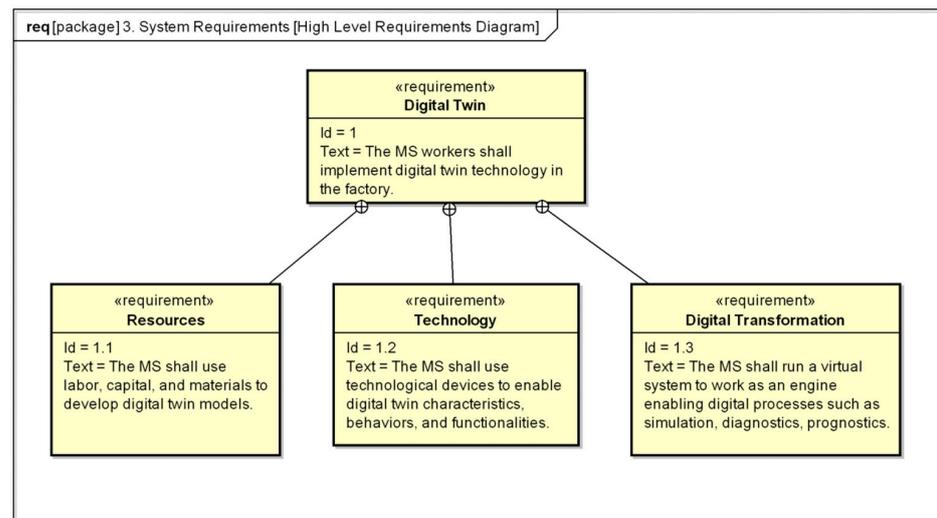


Figure 7. High-level requirements.

3.3.1. Use Cases

The system of interest considers the following use cases: install internet of things (IoT) devices, develop digital twin models, enable digital twin functionalities, and connect the factory to the digital twin. These use cases allow the top-level use case of implementing digital twin technology in the manufacturing system to be achieved. They are high-level tasks because they are composed of other tasks. Figure 8 shows the system's use cases.

1. **Install IoT devices:** This high-level task starts with the selection of IoT devices that are compatible with the manufacturing system. If they are not compatible, the factory must select other devices. If they are compatible, the operators proceed with installing the IoT devices to the factory. Then, the system operators must operate the IoT devices in the factory. Finally, the operators test the performance of these devices. If the IoT devices pass the test, they approve their installation. If not, they must be reinstalled.
2. **Develop digital twin models:** This high-level task starts with the installation of digital twin software. Then, the operators integrate the IoT devices to the digital twin software and set the configuration for their use. The next step is the digitization of the physical objects in the factory by creating computer-aided design (CAD) objects. The digital twin software must display these CAD objects. Finally, the operators must test and approve the virtual object's fidelity with respect to the real objects.
3. **Enable digital twin functionalities:** This high-level task starts with the operators mapping the factory processes to mirror them in the digital twin. Then, the operators define the digital twin functionalities and implement them to the digital twin software. The next step adds DT functionalities to the virtual models. Finally, operators display the digital twin functionalities and test their behavior. If the digital twin does not pass the test, operators must redefine the digital twin functionalities for the virtual models.
4. **Connect factory to digital twin:** This high-level task uses IoT devices to collect and centralize data in a database. Then, operators create the digital thread to integrate the physical manufacturing system and the digital twin. The digital thread enables the flow of data between the physical and virtual spaces. Finally, operators test the digital twin performance by feeding data from the factory to the digital twin and vice versa. They also test the digital twin functionalities. If the digital twin does not pass the test, operators must check and correct the integration between the physical and virtual spaces. If the digital twin passes the test, it is ready to be released and used in the factory's regular operations.

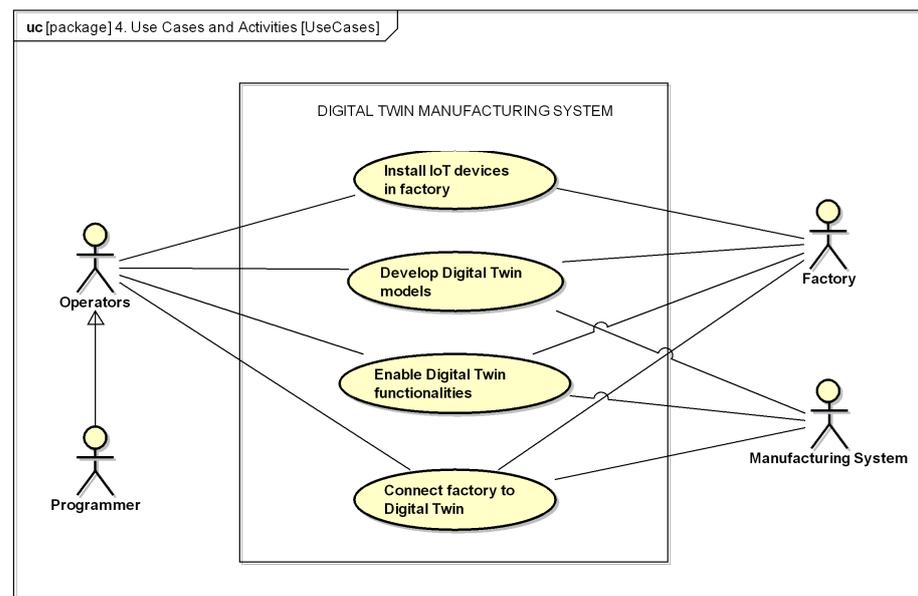


Figure 8. Use Cases.

3.3.2. System Logical Architecture

A logical architecture is an abstract representation of the requirements. It presents a structure design that defines functions, properties, and interfaces of logical components. It should be abstract and not give specific detail. Hence, it does not identify physical elements, but rather a baseline to start developing the physical system. In SysML, logical architecture uses block definition diagrams (BDD). These blocks or components distinguish from other diagram's blocks by the stereotype "Logical." The logical architecture divides the components in four categories or platforms: physical, virtual, information management, and business. This architecture enables the creation of the system's physical architecture [42]. The digital twin subsystem defines the logical components from the system's requirements in Figure 9.

3.4. Detailed Design

A detailed design shows the physical components that enable the realization of the system. It presents the system's physical architecture which derives from the system's logical architecture [31]. This section is the last activity of the system's planning and design.

System Physical Architecture

The physical architecture is a technical representation of the logical architecture. Physical architecture represents the structure design of the system's physical components. In SysML, physical architecture also uses block definition diagrams (BDD). The physical blocks or components distinguish from other diagrams by the stereotype "Physical." The physical architecture divides the components in three main categories: factory, data management, and DT software. Physical components are actual devices or software objects that realize logical components [42]. The "factory" physical components are the physical realization of the "physical platform" logical components. The "data management" components realize the "information management platform" logical component. Finally, the "DT software" realizes the "virtual platform" logical component. The digital twin subsystem model defines the physical components based on the previous logical architecture in Figure 10.

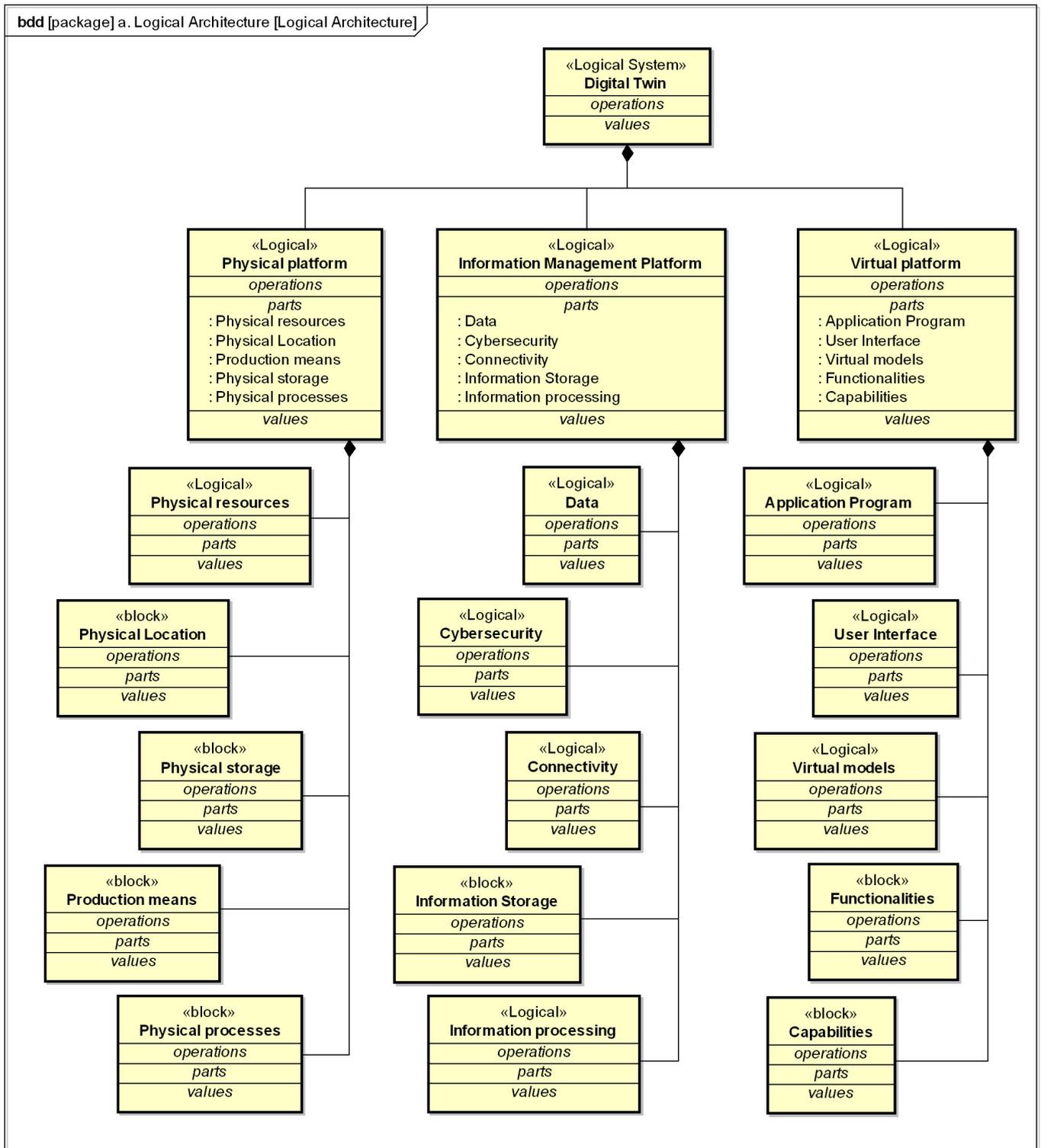


Figure 9. Logical Architecture.

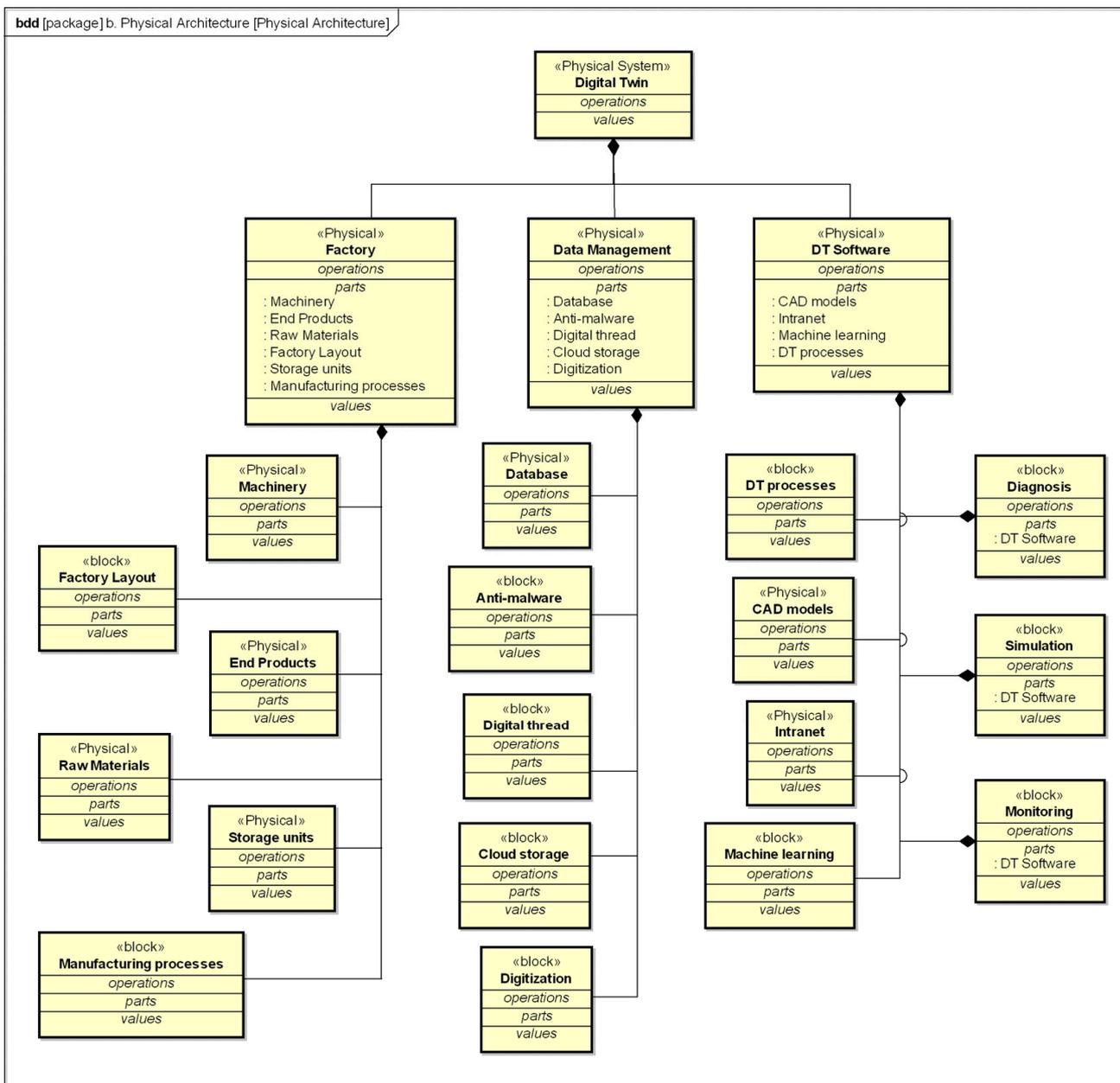


Figure 10. Physical Architecture.

4. Small-Scale Digital Twin Implementation Framework for Manufacturing Systems

This study proposes a framework to develop digital twins in manufacturing domains. This study considers the general configuration of digital twin technology and the digital twin components presented in the literature review. This configuration presents three main spaces or systems which are physical, virtual, and information. It allows the continuous dataflow through all spaces. It gives the digital twin updated information from the real world. It also helps to make better decisions and improve processes in the physical systems [4,12,14]. This framework also takes into consideration the system modelling presented above. It employs the use cases and system requirements to define the digital twin main goal, capabilities, and functionalities. It uses the system’s logical and physical architecture to define some key concepts part of a digital twin design. Figure 11 presents a framework to implement the digital twin in manufacturing systems.

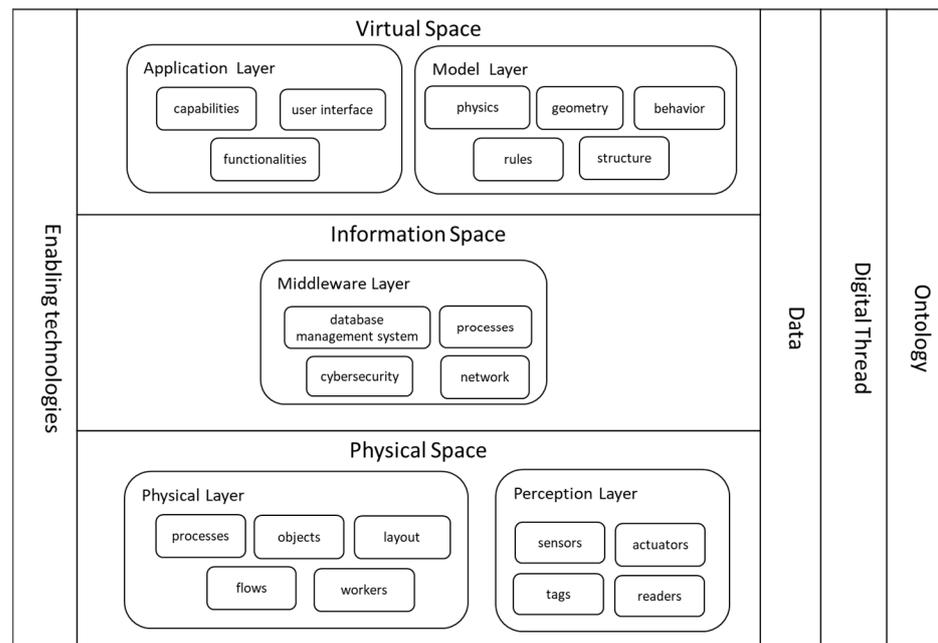


Figure 11. Small-Scale Digital Twin Implementation Framework.

This article develops a step-by-step process to use this framework and implement digital twins in a manufacturing system. The steps are the following:

1. Define what processes, products, or system to twin in the virtual space.
2. Install perception devices such as sensors, actuators, tags, and readers in the physical layer.
3. Create a database management system that give access to different type of databases.
4. Enable a digital thread that connects different types of data, devices, and systems.
5. Install a digital twin software that shows digital twin models and data and enables user's operation.
6. Create digital twin models with their properties in the DT software.
7. Enable digital twin functionalities such as feeding data to DT models.

There are some non-mandatory steps that users could follow to improve the digital twin maturity. One step is using a digital twin ontology to be familiar with DT concepts and relationships. Another step is integrating enabling technologies such as cloud computing, artificial intelligence, and so on to improve the system's digital twin.

After presenting the digital twin implementation framework for manufacturing systems, this study explains in detail the framework spaces, layers, and concepts lines below.

4.1. Physical Space

The physical space is a complex environment with many components interacting between each other. This article encompasses a discrete manufacturing system and its processes. It presents many processes such as product manufacturing, maintenance, logistics, product development, and so on. These processes have rules and a common physical constraint [48]. The system must install sensors over the physical asset that they want to resemble in the virtual space. This data helps to resemble the behavior and structure of objects and create digital twin models.

The physical space provides real-time data to enable digital twin capabilities such as simulation, control, and monitoring. These capabilities allow the physical objects and predict potential outcomes to be analyzed. Digital twin technology allows the physical space to control its objects automatically. It uses sensors and actuators in the real world to automatize processes. It will capture all the physical objects lifecycle in the virtual space through the information space [49]. This space presents two layers: thephysical layer and

the perception layer, with their respective components. The physical layer contains several physical objects that collect and send data to the virtual space for its analysis. It creates a network of objects that collaborate with each other to perform processes. The perception layer collects data and execute commands in the physical space.

The physical layer components for this framework are processes, objects, layout, workers, and flows. Physical processes are manufacturing activities to develop product from raw materials. They are a set of statements that assign certain behavior to a product. These processes transform inputs into end-products. Production processes create a system that interacts with other system to deliver products to customers [50]. Objects are entities such as machines, materials, parts, products, tools, and so on present in the physical space of the factory. These objects go through certain processes in a layout. Some of these objects (machines, parts, tools) are smart devices that connect to other devices creating a network or internet of things. They can provide real-time data about the manufacturing processes to the digital twin application and model layer. They can also receive feedback from these layers. The objects that are not smart need the perception layer components to send to and receive data from the virtual space. These objects share data such as position in the layout and status. Objects that use digital twins can enhance or develop an augmented perception of their physical environment [12,51]. The layout is the factory's floor that distributes the different objects, modules, and stations. It is the physical space where the processes transform resources into end-products. Factories layouts could be complex due to the high amount and variability of parameters. This complexity is related to the selection and positioning of objects. Digital twin can solve this complexity and improve a layout structure [7,52]. Workers are the human force that develop or assemble new products in a factory. Factories must match processes with skilled labor to fulfill production goals. This is a challenge that could decide a factory's productivity. Hence, productivity is related to the working layout and conditions in which workers perform their tasks. Workers are an essential component in the mechanical, physical, or chemical transformation of raw materials. Nevertheless, complex manufacturing systems such as additive manufacturing could replace workers with sophisticated but flexible machines [53,54]. Finally, flows are a sequence of processes which products follow in a manufacturing layout. They link different process parameters and organize them to finish at a certain time. They allow factories to design their work by defining the flow of people, materials, processes, and so on. This provides reliability and predictability to the factory's operations. They define the production lead time and capabilities that ensure the production of quality products [55,56].

The perception layer is in charge of collecting data from the physical space. It presents the following components: actuators, sensors, readers, and tags. A digital twin uses sensors to collect data of changes in the real world such as images, motion, pressure, and so on. Data are relevant for feeding digital twin models constantly in the virtual space. Sensors, being part of the IoT, can monitor and control processes. They can also upgrade standard devices into smart devices with network connectivity [57,58]. Actuators help machines, tools, or other devices to execute changes in the physical space. It uses electrical or hydraulic energy to command mechanical movements. Factories use them for opening doors, stop motions, execute motions, accelerate/deaccelerate, etc. Common types of actuators are pneumatic, electric, and electro-hydraulic [3,57,59]. Tags and readers are radio frequency identification (RFID) devices. Readers emit and receive signals from the tags. Tags communicate the location of the physical object [52].

4.2. Virtual Space

Digital twin uses the virtual space to show the physical objects in the virtual world. The virtual space is a copy of the physical space. It transforms physical objects into virtual objects. It resembles all characteristics of the physical counterpart. The virtual space displays the structure, behavior, information, and so on of the physical object [60,61]. It also displays diverse types of digital twin models based on components, assets, processes, and systems. IBM [62] explains that a digital twin for manufacturing systems could integrate

different digital twin models into a composite digital twin system. The virtual models allow the analysis of data and the improvement of the physical system. The digital twin goal is to implement post-analysis solutions in the physical system.

Digital twin recognizes every change on the physical asset. The virtual space mirrors its counterpart in the real world. It has different capabilities such as control, diagnostics, and prognostics [49]. It receives real-time data from the physical system to analyze it and give feedback to the physical system. It shows the system's current situation and potential cases. The virtual space allows the design and test of new models [63]. The simulation capability plays with the physical objects to propose potential changes for the benefit of the system. It can simulate the system's physics and structure. This capability allows operators to make better informed decisions throughout the system's lifecycle [7]. However, digital twin is more than a simulation tool [64]. Digital twin is a flexible and agile technology that works with real-time data under different use cases [22]. The user can monitor the physical space changes through the virtual space. It updates the virtual objects in real time. It can control physical objects from the virtual space using actuators in the physical space. These capabilities analyze data and provide information about the objects, processes, and services. All these add value and improve the system's operations.

The proposed framework presents two layers in the virtual space: application and model. The application layer analyzes data and sends useful information to the physical space. This layer helps employees to make better informed decisions. It analyzes short, medium, and long-term data. The application layer concepts are user interface, capabilities, and functionalities. User interface is the software that enables a human–digital twin technology interface. It shows the digital twin models and data. It allows users to operate the digital twin functionalities and capabilities. It helps operators to make data-driven decisions [65,66]. Digital twin presents main capabilities such as modelling, simulating, monitoring, and controlling. DT adds these capabilities to physical systems, improving their processes and functionalities. It also improves the efficiency and accuracy of physical objects. It makes physical systems smart with powerful communication and computing capabilities. DT capabilities enable a better simulation environment in terms of fidelity, speed, and granularity. A customized DT can choose the number and types of capabilities [3,33,49]. Finally, DT present some functionalities such as creating virtual models from physical objects, resembling the behavior of physical objects in the virtual space, using real-time data, providing feedback to the physical system, designing better products, solving complex problems, testing innovative ideas, and so on. These functionalities vary depending on the type of manufacturing system and the digital twin scope [67].

The model layer allows the DT to replicate physical objects in the virtual space. It presents all the characteristics of the physical objects. Based on the DT model level of fidelity, it could be indistinguishable from the physical object that it resembles. The model layer components are rules, physics, geometry, structure, and behavior. Digital twin rules are a group of triggers, conditions, and effects in the virtual models. These rules play an important role in deciding the digital twin system's architecture [68]. Physics-based digital twins are models that resemble the governing laws of nature such as space, time, and so on. These type of DT models require a great computational resource and processes. Currently, engineers use physics-based models for finite element analysis (FEA). Engineers should also consider the physics of objects for developing DT models. Physics-based models present benefits such as reliability and predictability [69]. Digital twin models must also consider geometry to design physical objects. Geometry describes the size, shape, position, and properties of physical objects. It could represent the digital twin in two-dimensional (2D) or three-dimensional (3D) form. Geometry elements are points, curves, lines, surfaces, bodies, patches, etc. These are necessary to form a solid geometric model. The collection of geometric objects leads to the creation of a mesh [67,70]. The structure concept organizes system components, elements, or parts, and presents its internal and external connections. It decomposes objects or classes into subcategories or subclasses. Then, it integrates them based on causal or correlational relationships. It describes value properties, interfaces, flows,

and constraints. The user interface displays the DT model's structure hierarchy [40,71]. The last concept in the model layer is behavior. Behavior defines the interaction between DT models such as activities, state machines, and sequences. There are two types of behaviors based on functions and state. The function-oriented behavior studies DT model activities, connections, and compositions. It shows the execution of activities such as the transformation of inputs into outputs. The state-based behavior studies the changing of models before and after function execution. DT models can show the history of objects through different transitions. Behaviors enable or realize DT capabilities. It can change the DT model's property values and structure [72,73].

4.3. Information Space

The information space connects and enables a bi-directional communication between the physical and virtual space. This space supports the digital twin's internal and external communication in real time. It supports the system's network and internet connection. Connectivity helps in the development and evolution of digital twins [60]. It supports the creation of a digital thread which generates data traceability keeping operators informed. Digital twin uses a digital thread to send and receive data from the physical and virtual space. Moreover, a digital thread allows the digital twin to analyze a system's lifecycle and integrate the system's components. Finally, it connects the digital twin with external systems that belong to the manufacturing supply chain [52,74].

Information space supports a continuous exchange of data between the physical and virtual space. This interaction enables all digital twin functionalities. This space helps to model objects, processes, systems, and end-products. After analyzing data, the virtual space uses the information space to give feedback to the physical space. Dataflow is important to predict failures and improve the system. Database abstracts physical space data and shares it with the different system components. It collects, compiles, preprocess, and stores data from both spaces. Digital twin uses historical data to create better solutions such as predictions, prescriptions, and diagnosis of the physical system [4,57,69]. This makes digital twin a smart technology able to learn.

The proposed framework includes the middleware layer in the information space. The middleware layer is an intermediary between the physical space and virtual space [75,76]. It manages communication between the two spaces. It uses a wireless or wired connection for this purpose. It stores real-time data collected in the physical space. It processes data and sends it to the virtual space for analysis. This layer feeds data to the application layer in the virtual space. It has two main functions: networking and data management. The networking function aims to exchange information along the distributed network of applications and objects. It allows the communication between the physical and model layer. The data management function stores data and supports the middleware to perform its processes [60,77,78].

The middleware layer components are network, cybersecurity, processes, and databases. Digital twin network connects objects from the physical space to the application layer. This network enables digital twin functionalities. A DT network focuses on communication technology and wireless communication. It allows continuous communication and transfers data between objects. It integrates several types of components with different communication protocols and technologies [4,79]. Cybersecurity looks to protect the virtual and physical space from threats such as malware, eavesdropping attacks, man-in-the-middle (MitM) attacks, denial-of-service (DOS) attacks, and so on. It defines policies, best practices, tools, guidelines, and technologies to assess risk, mitigate potential damage, and counterattack cybercriminals. It maintains the confidentiality and availability of information and data. An example of cybersecurity for digital twins are the authentication and authorization security processes. The authentication process verifies the user's identity. It determines if they are true or valid users. Authorization checks the user's access rights. It authorizes or denies their access. Digital twins could be a tool to enhance the cybersecurity of a factory. It could help to recognize attacks in real-time. Moreover, it could simulate potential

threats or damages to the system. DT can help to build a better security system [80,81]. Middleware processes connect different software, physical components, and data to bring a single centralized service to users. It links new applications such as digital twin to the manufacturing system. It manages different devices in the physical layer, allowing communication between them. It manages applications, provides internet connection, and allows the sending and receiving of data between layers [75,76]. The last concept in the middleware layer is the database management system (DMS). A DMS is a software that allows users to create and manage databases. It has several databases with different types of data. Databases store data from physical objects, processes, products, and so on. DMS can give access to several apps and users at the same time. It also brings security to data due to its centralized storage capability. Digital twins process a great amount of data to analyze, perform functionalities, and make decisions [82,83].

5. Discussion

This article follows a methodological structure to build a DT implementation framework. Frameworks provide a guideline that makes the development of digital twins easier. This framework intends to be easy to learn and precise in terms of concepts. They collect many concepts under a defined structure. It explains how to collect relevant concepts, and classify them in classes, spaces, or layers. It analyzes these concepts and their relationships, properties, and functions. It creates a structure based on their taxonomy. This framework could help small factories to build a digital twin of their products, processes, and systems. They are an effective tool to learn about digital twins and how to implement it in a manufacturing environment.

Before building the digital twin implementation framework, this article studies the complexities of digital twin in the literature review. The digital twin implementation framework uses data found in the literature review and digital twin subsystem model. This framework organizes the concepts in a general digital twin configuration with three spaces: physical, information, and virtual. Physical space presents two layers: physical and perception. Information space has a single layer: middleware. Virtual space presents two layers: application and model. In addition to these layers, users could use other concepts for the development and operation of a digital twin, such as ontology and enabling technologies. These concepts could help increase the maturity of a digital twin, but they are dispensable in its implementation.

Through the modelling of a digital twin subsystem, it is noticeable that DT is a complex system. It was necessary to use a MBSE approach to design it. MBSE provides consistency, traceability, and precision to the digital twin subsystem design. The systems modelling helped to build the digital twin implementation framework. It presents CONOPS, stakeholders, requirements, use cases, logical architecture, and physical architecture. It considers the implementation of digital twins in manufacturing as an operational need. It explains with use cases how to implement digital twins. It describes missions, requirements, activities, functions, objects, relationships between objects, and integration of spaces. It explores several devices and concepts for the physical and virtual spaces. It presents a logical and physical architecture that defines the abstract and technical components necessary for DT implementation.

This study presents a concise description of the DT implementation framework to help researchers and practitioners understand it. The digital twin implementation starts in the physical space. The physical space uses the perception layer components to get data from the physical layer. The physical layer components are the factory processes, machines, layout, tools, and every object that is physically there. The data travels through the information space to the virtual space. The information space saves all data in a database and digitizes it. It also enables the entire system to connect with other systems. The digitized data feeds the model and application layer in the virtual space. The application layer is the interface between the operator and the virtual world. It shows the digital twin software, capabilities, and functionalities. Digital twin software is the computer program

designed for end-users. Digital twin capabilities are simulation, monitoring, and diagnosis. The digital twin system has many functionalities in a factory. The systems modelling presents some high-level functionalities. Finally, the model layer represents the physical objects in the virtual space with all its characteristics, behaviors, structure, geometry, level of fidelity, and rules.

Finally, this study proposes a new definition for digital twins. “Digital twins are virtual objects that mirror physical objects in the virtual world. Digital twins’ characteristics, behaviors, functionalities, and connectivity vary according to their level of maturity.” This definition uses the verb mirror which means to show a reflection of a physical object. This reflection shows the characteristics and data of the original object. This new mirrored object is part of the virtual world. A channel or thread connects and allows communication between the physical and virtual object. Digital twin can be as complex as the system wants it to be. The system scope and desired functionalities decide the level of maturity of the digital twins.

6. Conclusions and Future Research

This article proposed a small-scale digital twin implementation framework for manufacturing systems. The authors used several concepts from the literature review and a digital twin model for manufacturing systems to build this framework. They developed a digital twin model using a MBSE approach. This model helped to define the DT concepts used later in the digital twin implementation framework. This framework uses a digital twin configuration with three spaces: physical, virtual, and information. These spaces have a continuous interaction to enable digital twin functionalities. These spaces present some layers with different concepts. This structure helps researchers and practitioners to learn about digital twins and apply it on their domains.

The development of a digital twin implementation framework highlights some digital twin characteristics. Digital twin is a modern technology that enables smart manufacturing, along with artificial intelligence, cloud computing, and so on. Digital twin looks to support different operations in the factory. It presents some functionalities such as collecting data, processing data, performing simulations, solving problems, and allowing communication between spaces. Digital twin provides feedback from predictions, prescriptions, and descriptions of current and potential situations to the physical space.

Moreover, this framework provides some conclusions for DT operations. DT must enable connection between different devices and applications. It must manage different devices and standardize them. It should help users to make data-driven decisions. It must define the traceability of data from the physical space, through the information space, and to the virtual space. It should be flexible to implement new functionalities and connect new devices and applications to the DT domain.

Finally, this article proposes future research for digital twin development. Future research needs to validate the proposed digital twin implementation framework. It must use the framework to create digital twin models of a factory. Manufacturing case studies are needed to apply the framework. This study could confirm that the proposed framework helps to create a small-scale digital twin for a manufacturing system. Future investigation could also refine the framework and include more concepts relevant to the DT development. This research could lead other researchers to work on DT implementation for manufacturing systems. For instance, researchers can use the proposed framework and compare it to other frameworks in the literature.

Author Contributions: Conceptualization, J.H.L., R.J.C. and K.L.; methodology, J.H.L., R.J.C. and K.L.; validation, J.H.L. and R.J.C.; formal analysis, J.H.L.; investigation, J.H.L.; data curation, J.H.L. and R.J.C.; writing—original draft preparation, J.H.L.; writing—review and editing, J.H.L., R.J.C. and K.L.; visualization, J.H.L.; supervision, R.J.C. and K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Deloitte Development LLC. *Industry 4.0 and the Digital Twin Technology*; Deloitte Development LLC: Dallas, TX, USA, 2017.
2. Frank, A.G.; Dalenogare, L.S.; Ayala, N.F. Industry 4.0 technologies: Implementation patterns in manufacturing companies. *Int. J. Prod. Econ.* **2019**, *210*, 15–26. [[CrossRef](#)]
3. Tao, F.; Qi, Q.; Wang, L.; Nee, A.Y.C. Digital Twins and Cyber-Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. *Engineering* **2019**, *5*, 653–661. [[CrossRef](#)]
4. Lu, Y.; Liu, C.; Wang, K.I.-K.; Huang, H.; Xu, X. Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robot. Comput. Integr. Manuf.* **2020**, *61*, 101837. [[CrossRef](#)]
5. Li, Y.; Tao, Z.; Wang, L.; Du, B.; Guo, J.; Pang, S. Digital twin-based job shop anomaly detection and dynamic scheduling. *Robot. Comput. Integr. Manuf.* **2023**, *79*, 102443. [[CrossRef](#)]
6. Tao, F.; Cheng, J.; Qi, Q.; Zhang, M.; Zhang, H.; Sui, F. Digital twin-driven product design, manufacturing and service with big data. *Int. J. Adv. Manuf. Technol.* **2018**, *94*, 3563–3576. [[CrossRef](#)]
7. Kuehn, W. Digital twins for decision making in complex production and logistic enterprises. *Int. J. Des. Nat. Ecodynamics* **2018**, *13*, 260–271. [[CrossRef](#)]
8. Židek, K.; Pitel', J.; Adámek, M.; Lazorík, P.; Hošovský, A. Digital twin of experimental smart manufacturing assembly system for industry 4.0 concept. *Sustainability* **2020**, *12*, 3658. [[CrossRef](#)]
9. Qi, Q.; Tao, F. Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison. *IEEE Access* **2018**, *6*, 3585–3593. [[CrossRef](#)]
10. He, B.; Bai, K.J. Digital twin-based sustainable intelligent manufacturing: A review. *Adv. Manuf.* **2021**, *9*, 1–21. [[CrossRef](#)]
11. Saad, A.; Faddel, S.; Mohammed, O. IoT-Based Digital Twin for Energy Cyber-Physical Systems: Design and Implementation. *Energies* **2020**, *13*, 4762. [[CrossRef](#)]
12. Yang, W.; Yoshida, K.; Takakuwa, S. Digital Twin-Driven Simulation for a Cyber-Physical System in Industry 4.0 Era. In *Daaam International Scientific Book*; DAAAM International Vienna: Vienna, Austria, 2017; pp. 227–234.
13. Yildiz, E.; Møller, C.; Bilberg, A. Virtual Factory: Digital Twin Based Integrated Factory Simulations. *Procedia CIRP* **2020**, *93*, 216–221. [[CrossRef](#)]
14. Grieves, M. *Digital Twin: Manufacturing Excellence through Virtual Factory Replication*; Florida Institute of Technology: Melbourne, FL, USA, 2014.
15. Kritzing, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* **2018**, *51*, 1016–1022. [[CrossRef](#)]
16. Trauer, J.; Schweigert-Recksiek, S.; Engel, C.; Spreitzer, K.; Zimmermann, M. What is a digital twin?—Definitions and insights from an industrial case study in technical product development. In *Proceedings of the Design Society: DESIGN Conference*; Cambridge University Press (CUP): Cambridge, UK, 2020; Volume 1, pp. 757–766.
17. Martínez-gutiérrez, A.; Díez-gonzález, J.; Ferrero-guillén, R.; Verde, P.; Álvarez, R.; Perez, H. Digital Twin for Automatic Transportation in Industry 4.0. *Sensors* **2021**, *21*, 3344. [[CrossRef](#)] [[PubMed](#)]
18. Identity Management Institute Digital Twin Technology Benefits and Challenges. Available online: <https://identitymanagementinstitute.org/digital-twin-technology-benefits-and-challenges/> (accessed on 27 May 2021).
19. Tao, F.; Zhang, H.; Liu, A.; Nee, A.Y.C. Digital Twin in Industry: State-of-the-Art. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2405–2415. [[CrossRef](#)]
20. Bordeleau, F.; Combemale, B.; Eramo, R.; van den Brand, M.; Wimmer, M. Towards Model-Driven Digital Twin Engineering: Current Opportunities and Future Challenges. *Commun. Comput. Inf. Sci.* **2020**, *1262*, 43–54. [[CrossRef](#)]
21. Jabareen, Y. Building a Conceptual Framework: Philosophy, Definitions, and Procedure. *Int. J. Qual. Methods* **2009**, *8*, 49–62. [[CrossRef](#)]
22. Guo, J.; Zhao, N.; Sun, L.; Zhang, S. Modular based flexible digital twin for factory design. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1189–1200. [[CrossRef](#)]
23. Zhang, Z.; Wang, X.; Wang, X.; Cui, F.; Cheng, H. A simulation-based approach for plant layout design and production planning. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 1217–1230. [[CrossRef](#)]
24. Marmolejo-Saucedo, J.A. Digital Twin Framework for Large-Scale Optimization Problems in Supply Chains: A Case of Packing Problem. *Mob. Netw. Appl.* **2022**, *27*, 2198–2214. [[CrossRef](#)]
25. Kumbhar, M.; Ng, A.H.C.; Bandaru, S. A digital twin based framework for detection, diagnosis, and improvement of throughput bottlenecks. *J. Manuf. Syst.* **2023**, *66*, 92–106. [[CrossRef](#)]
26. Friederich, J.; Francis, D.P.; Lazarova-Molnar, S.; Mohamed, N. A framework for data-driven digital twins of smart manufacturing systems. *Comput. Ind.* **2022**, *136*, 103586. [[CrossRef](#)]

27. Shao, G.; Helu, M. Framework for a digital twin in manufacturing: Scope and requirements. *Manuf. Lett.* **2020**, *24*, 105–107. [[CrossRef](#)] [[PubMed](#)]
28. STEP-NC AP238 ISO 23247, Digital Twin Framework for Manufacturing. Available online: <https://www.ap238.org/iso23247/> (accessed on 21 December 2022).
29. Loaiza, J.H.; Cloutier, R.J. Analyzing the Implementation of a Digital Twin Manufacturing System: Using a Systems Thinking Approach. *Systems* **2022**, *10*, 22. [[CrossRef](#)]
30. Zhao, X.; Zhao, L.; Hu, Z.X.; Zhou, C.L. Modeling and analysis of the discrete manufacturing product quality with information formation process for traceability. *Appl. Mech. Mater.* **2013**, *345*, 477–481. [[CrossRef](#)]
31. U.S. Department of Transportation. *Systems Engineering for Intelligent Transportation Systems*; FHWA-HOP-0; Federal Highway Administration: Washington, DC, USA, 2007.
32. ISO/IEC/IEEE 15288; International Standard ISO/IEC/IEEE 15288 Systems and Software Engineering—System Life Cycle Processes. IEEE: New York, NY, USA, 2015; Volume 17.
33. AIAA & AIA. *Digital Twin: Definition & Value*; AIAA & AIA: Reston, VA, USA, 2020.
34. DI SPRING Digital Twin. Available online: <https://dispring.com/Products/Digital-Twin> (accessed on 27 May 2021).
35. Barricelli, B.R.; Casiraghi, E.; Fogli, D. A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *IEEE Access* **2019**, *7*, 167653–167671. [[CrossRef](#)]
36. Shea, G. *NASA Systems Engineering Handbook Revision 2*; National Aeronautics and Space Administration: Washington, DC, USA, 2017.
37. Forsberg, K.; Mooz, H.; Cotterman, H. *Visualizing Project Management: Models and Frameworks for Mastering Complex Systems*; J. Wiley: Hoboken, NJ, USA, 2005; ISBN 978-0-471-64848-2.
38. Firesmith, D. Using V Models for Testing. Available online: <https://insights.sei.cmu.edu/blog/using-v-models-for-testing/> (accessed on 28 December 2022).
39. Khan, N.U.; Mclucas, A. *A Case Study in Application of Vee Model of Systems Engineering to System Dynamics Modelling of Dryland Salinity in Australia*; UNSW Canberra at the Australian Defence Force Academy: Canberra, Australia, 2008.
40. SysML.org. *Systems Modeling Language (SysML) Specification*. 2005. Available online: <https://sysml.org/.res/docs/specs/SysML-v0.9-PDF-050110R1.pdf> (accessed on 6 December 2022).
41. Selvy, B.M.; Claver, C.; Angeli, G. Using SysML for verification and validation planning on the Large Synoptic Survey Telescope (LSST). In *Proceedings of the Modeling, Systems Engineering, and Project Management for Astronomy VI*; SPIE: Bellingham, WA, USA, 2014; Volume 9150, p. 91500N.
42. Delligatti, L. *SysML Distilled: A Brief Guide to the Systems Modeling Language*; Addison-Wesley: Boston, WA, USA, 2014; ISBN 9780133430356.
43. Davis, J.; Edgar, T.; Porter, J.; Bernaden, J.; Sarli, M. Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Comput. Chem. Eng.* **2012**, *47*, 145–156. [[CrossRef](#)]
44. U.S. Department of Commerce Smart Manufacturing Operations Planning and Control Program | NIST. Available online: <https://www.nist.gov/programs-projects/smart-manufacturing-operations-planning-and-control-program> (accessed on 27 May 2021).
45. Miles, M.B.; Huberman, M. *Qualitative Data Analysis: An Expanded Sourcebook*, 2nd ed.; Sage Publications, Inc.: Thousand Oaks, CA, USA, 1994; ISBN 9781506353081.
46. Levering, B. Concept Analysis as Empirical Method. *Int. J. Qual. Methods* **2002**, *1*, 35–48. [[CrossRef](#)]
47. Singh, M.; Fuenmayor, E.; Hinchy, E.P.; Qiao, Y.; Murray, N.; Devine, D. Digital Twin: Origin to Future. *Appl. Syst. Innov.* **2021**, *4*, 36. [[CrossRef](#)]
48. Qin, J.; Liu, Y.; Grosvenor, R. A Categorical Framework of Manufacturing for Industry 4.0 and Beyond. *Procedia CIRP* **2016**, *52*, 173–178. [[CrossRef](#)]
49. Agrawal, A.; Fischer, M.; Singh, V. Digital Twin: From Concept to Practice. *J. Manag. Eng.* **2022**, *38*, 06022001. [[CrossRef](#)]
50. Black, J.T. Manufacturing Systems. In *Encyclopedia of Production and Manufacturing Management*; Springer: New York, NY, USA, 2006; pp. 423–431.
51. Chukhno, O.; Chukhno, N.; Araniti, G.; Campolo, C.; Iera, A.; Molinaro, A. Optimal placement of social digital twins in edge iot networks. *Sensors* **2020**, *20*, 6181. [[CrossRef](#)] [[PubMed](#)]
52. Pang, T.Y.; Pelaez Restrepo, J.D.; Cheng, C.T.; Yasin, A.; Lim, H.; Miletic, M. Developing a Digital Twin and Digital Thread Framework for an 'Industry 4.0' Shipyard. *Appl. Sci.* **2021**, *11*, 1097. [[CrossRef](#)]
53. Weller, S. Manufacturing and labor. In *Handbook of Manufacturing Industries in the World Economy*; Edward Elgar Publishing Ltd.: Northampton, MA, USA, 2015; pp. 30–41, ISBN 9781781003930.
54. Levinson, M. *What Is Manufacturing? Why Does the Definition Matter?* Congressional Research Service: Washington, DC, USA, 2017.
55. Dallery, Y.; Gershwin, S.B. Manufacturing flow line systems: A review of models and analytical results. *Queueing Syst.* **1992**, *12*, 3–94. [[CrossRef](#)]
56. Boukas, E.K.; Haurie, A. Manufacturing Flow Control and Preventive Maintenance: A Stochastic Control Approach. *IEEE Trans. Automat. Contr.* **1990**, *35*, 1024–1031. [[CrossRef](#)]
57. He, R.; Chen, G.; Dong, C.; Sun, S.; Shen, X. Data-driven digital twin technology for optimized control in process systems. *ISA Trans.* **2019**, *95*, 221–234. [[CrossRef](#)]

58. Kadlec, P.; Gabrys, B.; Strandt, S. Data-driven Soft Sensors in the process industry. *Comput. Chem. Eng.* **2009**, *33*, 795–814. [CrossRef]
59. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Futur. Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [CrossRef]
60. Grieves, M.W. *Virtually Intelligent Product Systems: Digital and Physical Twins*; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2019; ISBN 9781624105654.
61. Lai, Y.; Wang, Y.; Ireland, R.; Liu, A. Digital twin driven virtual verification. In *Digital Twin Driven Smart Design*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 109–138.
62. Armstrong, M.M. Cheat Sheet: What Is Digital Twin? Available online: <https://www.ibm.com/blogs/internet-of-things/iot-cheat-sheet-digital-twin/> (accessed on 27 May 2021).
63. Xie, X.; Lu, Q.; Rodenas-Herraiz, D.; Parlikad, A.K.; Schooling, J.M. Visualised inspection system for monitoring environmental anomalies during daily operation and maintenance. *Eng. Constr. Archit. Manag.* **2020**, *27*, 1835–1852. [CrossRef]
64. Exor International What Is the Difference between a Simulation and a Digital Twin? Available online: <https://www.exorint.com/en/blog/what-is-the-difference-between-a-simulation-and-a-digital-twin> (accessed on 27 May 2021).
65. Diehl, S. Software visualization. In Proceedings of the International Seminar, Dagstuhl Castle, Wadern, Germany, 20–25 May 2001; p. 403.
66. Madni, A.M.; Nance, M.; Richey, M.; Hubbard, W.; Hanneman, L. Toward an experiential design language: Augmenting Model-Based Systems Engineering with technical storytelling in virtual worlds. *Procedia Comput. Sci.* **2014**, *28*, 848–856. [CrossRef]
67. Söderberg, R.; Wärmeffjord, K.; Carlson, J.S.; Lindkvist, L. Toward a Digital Twin for real-time geometry assurance in individualized production. *CIRP Ann.-Manuf. Technol.* **2017**, *66*, 137–140. [CrossRef]
68. Wu, C.; Zhou, Y.; Pereira Pessôa, M.V.; Peng, Q.; Tan, R. Conceptual digital twin modeling based on an integrated five-dimensional framework and TRIZ function model. *J. Manuf. Syst.* **2020**, *58*, 79–93. [CrossRef]
69. Kapteyn, M.G.; Knezevic, D.J.; Huynh, D.B.P.; Tran, M.; Willcox, K.E. Data-driven physics-based digital twins via a library of component-based reduced-order models. *Int. J. Numer. Methods Eng.* **2020**, *123*, 2986–3003. [CrossRef]
70. Friedenthal, S.; Moore, A.; Steiner, R. Residential Security System Example Using the Object-Oriented Systems Engineering Method. In *A Practical Guide to SysML*; Morgan Kaufmann: Burlington, MA, USA, 2015; pp. 417–504, ISBN 978-0-12-800202-5.
71. Weilkens, T. NextGenSysML Part 6—Structure Requirements—Model Based Systems Engineering. Available online: <https://mbse4u.com/2019/03/07/nextgensysml-part-6-structure-requirements/> (accessed on 21 July 2022).
72. Erickson, C. *Object Oriented Programming*; Atomic Object LLC: Grand Rapids, MI, USA, 2009.
73. Weilkens, T. NextGenSysML Part 8—Behavior Requirements—Model Based Systems Engineering. Available online: <https://mbse4u.com/2019/05/20/nextgensysml-part-8-behavior-requirements/> (accessed on 22 July 2022).
74. Madni, A.; Madni, C.; Lucero, S. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems* **2019**, *7*, 7. [CrossRef]
75. Angrish, A.; Starly, B.; Lee, Y.S.; Cohen, P.H. A flexible data schema and system architecture for the virtualization of manufacturing machines (VMM). *Manuf. Syst.* **2017**, *45*, 236–247. [CrossRef]
76. IBM. Cloud Education What Is Middleware? Available online: <https://www.ibm.com/cloud/learn/middleware> (accessed on 26 July 2022).
77. Alam, K.M.; El Saddik, A. C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems. *IEEE Access* **2017**, *5*, 2050–2062. [CrossRef]
78. Steinmetz, C.; Rettberg, A.; Ribeiro, F.G.C.; Schroeder, G.; Pereira, C.E. Internet of things ontology for digital twin in cyber physical systems. *Braz. Symp. Comput. Syst. Eng. SBESC* **2018**, *2018*, 154–159. [CrossRef]
79. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [CrossRef]
80. Holmes, D.; Papathanasaki, M.; Maglaras, L.; Ferrag, M.A.; Nepal, S.; Janicke, H. Digital Twins and Cyber Security—Solution or challenge? In Proceedings of the 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), Preveza, Greece, 24–26 September 2021. [CrossRef]
81. Craigen, D.; Diakun-Thibault, N.; Purse, R. Defining Cybersecurity. *Technol. Innov. Manag. Rev.* **2014**, *4*, 13–21. [CrossRef]
82. Derclaye, E. What is a Database? *J. World Intellect. Prop.* **2005**, *5*, 981–1011. [CrossRef]
83. Gunjal, B. Database System: Concepts and Design. In Proceedings of the 24th IASLIC-SIG-2003, Dehradun, India, 15–18 December 2003; p. 20.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.