machines

Article

The Anatomy of the Internet of Digital Twins: A Symbiosis of Agent and Digital Twin Paradigms Enhancing Resilience (Not Only) in Manufacturing Environments

Joel Lehmann [1,*], Andreas Lober [2], Tim Häußermann [1], Alessa Rache [1], Lisa Ollinger [3], Hartwig Baumgärtel [2] and Julian Reichwald [1]

[1] Center for Mass Spectrometry and Optical Spectroscopy, Mannheim University of Applied Sciences, Paul-Wittsack-Straße 10, 68163 Mannheim, Germany
[2] Institute for Industrial Engineering and Supply Chain Management, Ulm Technical University of Applied Sciences, Prittwitzstraße 10, 89075 Ulm, Germany
[3] Institute for Manufacturing Technology and Material Science, Ulm Technical University of Applied Sciences, Prittwitzstraße 10, 89075 Ulm, Germany
* Correspondence: j.lehmann@hs-mannheim.de

Abstract: Due to the growing environmental and geopolitical challenges nowadays, which are causing supply chain complications, industry and society are facing significant new objections. As a complement and extension to the technology-driven premises of Industry 4.0, the value-driven Industry 5.0 focuses on society and the environment. Human centricity, sustainability, and resilience should become a more integral part of both industrial and societal revolutions. One of the enabler technologies for both is the Digital Twin (DT). In order to make DTs intelligent, they must become active, online, goal-seeking, and anticipatory. To meet these requirements, the characteristics of Multi-Agent Systems (MASs) can be employed. This paper contributes to the bilateral emergence of the two industrial paradigms and establishes an approach for the provision of Intelligent Digital Twins (IDTs) within the Internet of Digital Twins (IoDT). Initially, a DT reference model aligned with already established Industry 4.0 reference models enriched with the goals of Industry 5.0 is developed, followed by an outline of how IDTs can be realized with the characteristics of MAS. The work is substantiated by an architectural design for IDTs choreographing marketplace-oriented production processes with a subsequent prototypical implementation, followed by a proof of concept.

Keywords: Intelligent Digital Twins; Multi-Agent Systems; Internet of Digital Twins; IoDT; Industry 5.0; cyber-physical systems; Digital Twin reference model; RAMI 4.0; IIRA

Citation: Lehmann, J.; Lober, A.; Häußermann, T.; Rache, A.; Ollinger, L.; Baumgärtel, H.; Reichwald, J. The Anatomy of the Internet of Digital Twins: A Symbiosis of Agent and Digital Twin Paradigms Enhancing Resilience (Not Only) in Manufacturing Environments. *Machines* **2023**, *11*, 504. https://doi.org/10.3390/machines11050504

Academic Editors: Michael Grieves and John Vickers

Received: 8 March 2023
Revised: 24 March 2023
Accepted: 6 April 2023
Published: 22 April 2023

1. Introduction

Over the last decade, problems such as climate change, global pandemics, or geopolitical crises have evolved, causing supply chain complications and shortages; for example, in the semiconductor or food industries [1–3]. These challenges nowadays have provoked a rethinking process in society and industry. There is a remarkable trend in research and industry emerging to other paradigms, in addition to the well-known and published Industry 4.0 or—more generally—the digital transformation of industry and society. As a basis for discussion, the European Union (EU), in conjunction with academia and industry, has developed a white paper [4] which introduces Industry 5.0. It is intended to be human-centric, sustainable, and resilient, achieving a balance between economy, ecology, and society. A future manufacturing plant or industrial site must be designed to meet human needs through a value-based and ethical design of technology that creates symbiotic collaboration between humans and machines [5]. Despite the high degree of autonomy, future DT systems must still rely on computers and humans [6]. Whilst Industry 4.0 is supposed to be a technological revolution, Industry 5.0 is driven by social values. It increases productivity and utilization, reduces labor and accidents, and consequently, acts environmentally and is humanly friendly [7]. Technological advances from Industry 4.0 can be used to achieve the value-oriented goals of Industry 5.0. However, this should not be seen as a successor or replacement for Industry 4.0, but rather, as a complementary, adjunctive revolution. As a result, industry and society effectively work hand in hand. The coincidence of technology and values as drivers is causing a techno-social revolution [8].

According to [9], the enabler technologies for both revolutionary paradigms include: (1) Human-centric solutions and human–machine-interaction; (2) Bio-inspired technologies and smart materials; (3) Real-time-based Digital Twins (DTs) and simulation; (4) Cyber-safe data transmission, storage, and analysis technologies; (5) Artificial Intelligence; and (6) Technologies for energy efficiency and trustworthy autonomy. A closer look at these enablers quickly reveals that Industry 5.0 requires the progress and underlying technologies of Industry 4.0.

Particular importance can be ascribed to the concept of DTs. For both revolutions, these offer the pivotal point for further intelligent handling and processing of accruing data [4,8,9]. For example, process uncertainties can be simulated in advance, waste can be reduced, and the process flow can be optimized [7]. After DTs emerged as a key to sustainability in the literature, it has not yet been adequately considered in the field of manufacturing in terms of intelligent properties [10]. Making DTs intelligent enables a higher level of autonomy and proactivity of cyber-physical systems (CPSs). Therefore, superordinate levels of cognition (integrated simulations and collaborative decision-making) and configuration (self-adaptation and optimization, resilience) according to [11] can be achieved. In order to accomplish this, Intelligent Digital Twins (IDTs) must become active, online, goal-seeking, and anticipatory [12]. Multi-Agent Systems (MASs) offer suitable characteristics, which can facilitate the deployment of such IDTs [13].

This paper contributes to the bilateral emergence of both industrial paradigms and establishes an approach for the provision of IDTs within the Internet of Digital Twins (IoDT). This brings added value to contemporary technological realizations of DTs, as well as human-centered resilient approaches tailored to the increasing requirements of future manufacturing. Based on the elaborated related work, two hypotheses facing the growing challenges on DTs and future industry demands will be examined. Based on both hypotheses, a new reference model is first developed (*H1*), and then the necessity of MAS structures is shown as a possible implementation modality (*H2*).

The first hypothesis, *(H1) Digital Twins must be distinguished according to their levels of application correlated to the proximity to the production process*, deals with the wide variety of definitions for DTs that increasingly distort its initially defined purpose [14]. Regardless of the use case and structure as described in most definitional approaches, the existing reference models are not sufficiently described in terms of the intelligence and autonomous interactions of DTs. For a distinguished provision of DTs, a reorientation of a reference model is mandatory. It is derived from the Reference Architectural Model Industry 4.0 (RAMI 4.0) and the Industrial Internet Reference Architecture (IIRA), and is enriched with important characteristics of Industry 5.0.

The second hypothesis, *(H2) Implementing IDTs within such a reference model requires the symbiosis with Multi-Agent System (MAS) paradigms*, focuses on the promising characteristics and features of MAS which can be suitably fitted to DT requirements [13,15]. The previously realigned reference model is intended to be realized in substantial parts through the use of these technology paradigms. To implement an Industry 4.0-compliant DT, MAS approaches can be used to improve the digitization process, establish distributed intelligence, and create collaborative functionalities for holonic system organization [16]. A correlation of the concepts of IDTs and MASs forms the basis for the partial implementation of the presented architecture. The prototypical implementation shown is substantiated by a marketplace-like choreography of IDTs by means of a production process as a use case.

The paper is structured as follows. Starting with Section 2, the state of the art and the related work of DTs and MAS will be presented. The subsections particularly describe the interrelationships of complementary DT implementations through MAS. Section 3 elaborates an entirely new reference model for DTs based on existing industrial reference models, and concludes by correlating IDT characteristics with MAS characteristics. The following section, Section 4, describes the architectural approach designed based on the characteristics previously derived. After a representative use case from the orchestration of production processes has been illustrated in Section 5, a prototypical implementation with a proof of concept follows in Section 6. Subsequently, a discussion and evaluation of how the implemented architecture contributes to the propagation of industrial paradigms is given in Section 7, concluding the work with a perspective on future challenges in Section 8.

2. Related Work

To create a common understanding of the context of DTs and MAS, the following three subsections provide a basis for further proceedings. First, the origins and different types of DTs and their evolution are discussed. Then, MAS are introduced and subsequently analyzed in more detail in terms of intersections with DTs. This facilitates the basis to further derive a Digital Twin Reference Model (DTRM) and the realization methods of IDTs through MAS paradigms.

2.1. Digital Twins

The DT paradigm was introduced by Michael Grieves in 2002, and was first named by John Vickers. It was intended to be a comprehensive ideal for Product Lifecycle Management (PLM) [17–19]. Despite the long existence of about two decades, the concept turnd out to be a cutting-edge enabler technology of Industry 4.0 and 5.0, but it is still in its infancy with respect to intelligence and autonomy [8,20]. Especially, the applicability in broadband use case domains allows us to keep a focus on research despite relevant standardization activities [21–23].

With regard to Figure 1, which represents Grieves' approach as the smallest denominator in the understanding of DTs [24], the basic concept distinguishes between the physical and digital space. The physical space depicted on the left shelters tangible components, i.e., machines/assets, production equipment, products, production demands/orders, or also physical processes, the so-called Physical Twins (PTs). On the right side, its digital counterpart, the DT, is represented in digital space. All raw data aggregated by the PT are mirrored in its respective DT. The arbitrarily designed data processing of the digital space converts them into valuable information and uses them to influence the real world through the PTs. The bilateral stream between the two spaces of data, and information is denoted as the digital thread. To cover the entire product lifecycle, three different types of DTs are subdivided: the Digital Twin Prototype, which includes the prototype of an entity not yet produced with regard to variants, the Digital Twin Instance, which is the product itself in an as-built state, and finally, the Digital Twin Aggregate, which is intended to aggregate all data from Digital Twin Instances [12,18,25]. The aim is to relocate working activities from the physical space to the digital space so that efficiency and resources are sustainably and economically preserved [12].

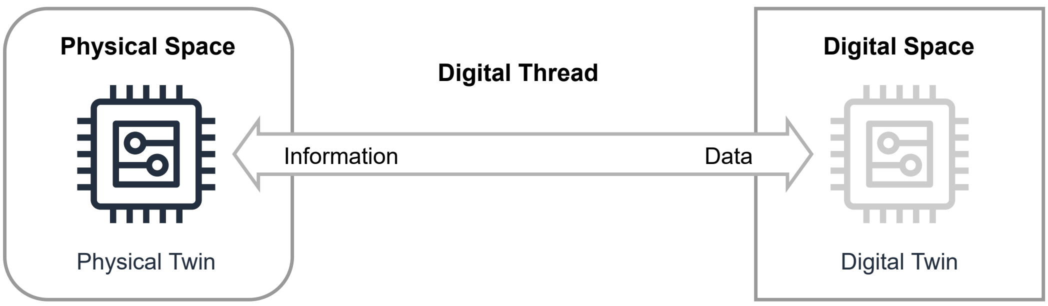

Figure 1. Conceptual Structure of a Digital Twin According to Grieves.

Most DT implementations are limited to representing passive properties. It is mandatory to grant a higher degree of intelligence [12,26]. Therefore, DTs must develop active, online, goal-seeking, and anticipatory characteristics to achieve their full potential [12,27]. How these individual characteristics can be modeled is described at the end of Section 3. According to Jazadi et al. [28], IDTs must offer semantic descriptions of the models, intelligent algorithms for higher-level information generation for quality assessments, and services to provide these functionalities. Through IDTs, highly reconfigurable design can be established that significantly reduces software and hardware engineering efforts [26].

In order to classify DTs in terms of their intelligent and autonomous capabilities, Van der Falk et al. [29] have identified five archetypes based on literature reviews and industry surveys. The first two stages of *Basic Digital Twins* and *Enriched Digital Twins* offer only passive representational capabilities. In the *Autonomous Control Twin, Enhanced Autonomous Control Twin, and Exhaustive Twin* that build on this, increasingly proactive and intelligent capabilities are attributed, up to and including a fully fledged autonomous system. However, the first three archetypes can already be found in industrial applications, whereas the more sophisticated approaches are limited to research. In the further archetypes, DTs also could provide the best possible foundation for big data and artificial intelligence approaches [30].

One approach to make DTs even more human-centric would be to use human–data interaction principles [31]. Certain mechanisms are used to place the human at the center of the data flow so that individual groups can interact with the overall system [32]. This can further expand the focus of human needs and make access to complex data correlations tangible [33].

Research fields that are not adequately addressed are the operational control and optimization of the dispatch process through collaborative DTs [34]. Therefore, Lehmann et al. [20] use a production scenario to show which prerequisites are needed for proactive collaboration between DTs. They argue that a knowledge representation within the system participants is indispensable for an intelligent interaction. Vogel-Heuser et al. [35] emphasize that a knowledge representation is essential for production systems, especially to obtain DTs from there. Baumgärtel [15] also considers semantic web technologies to be an intrinsic driver of DT systems. Knowledge graphs can improve IDTs. Only by reasoning over knowledge graphs can self-adaptation and self-adjustment be achieved [36,37].

A highly discussed way of implementing DTs is the Asset Administration Shell (AAS). The concept was originated by the German Industry 4.0 initiative as an alliance of industrial companies, associations, and research. Building on RAMI 4.0, the AAS provides the framework for vendor-independent interoperability across the entire product lifecycle [38]. In most cases, only passive features are used in the realization of AAS. Advanced autonomous functionalities, such as intelligent decision-making or collaboration, of reactive and further proactive AAS behavior are still determined as a research gap [13]. Consequently, there is also no adequate framework for DT provision through AAS [39].

After the relevance of DTs for current problems has been identified, the following section will be devoted to MAS. The two concepts are then examined for commonalities.

2.2. Multi-Agent Systems

An agent is a computer system situated in an environment, and it is able to autonomously perform actions in that environment to achieve its goals. An MAS consists of a minimum of two agents interacting through communication and acting in an environment. Each agent of the MAS has different so-called spheres of influence and is connected by other organizational relationships. Agents act on behalf of users, with different goals and motivations. To interact successfully, they must be able to cooperate, coordinate, and negotiate with each other, much like humans do. According to Wooldridge, the most essential characteristic of an agent is that they are autonomous and therefore can act independently [40].

Concerning Industry 4.0, a technical agent is a definable unit with a determined goal that strives to fulfill this goal by acting independently and interacting with the environment and other agents. An agent performs one or more functions, and assumes one or more roles in an MAS for this purpose. In this respect, agent encapsulation satisfies the secrecy principle, so that not all internal states and properties of a resource are visible to the outside world and know the properties, functions, and states (e.g., load) of the asset that they represent [41].

More specifically, Wooldridge describes the intelligent agent, to which he assigns three types of behavior: reactive, proactive, and social. The reactivity of agents is understood as the behavior of a system that constantly interacts with its environment and reacts to changes in it. The proactivity of agents is defined as goal-directed behavior that develops and that attempts to achieve goals. This goal-seeking behavior ensures that intelligent agents take the initiative and act proactively [40]. Proactivity enables the resource to initiate activities through the agent independently [41]. Social capability is understood as the interaction in an MAS. An MAS can only achieve some goals through interaction with other agents. Social capability in agents is thus the ability to interact with other agents (and possibly humans) through cooperation, coordination, and negotiation [40]. Agent interaction in Industry 4.0 scenarios enables communication at the semantic and pragmatic level via a vocabulary and ontologies to which these agents have access. The real-time capabilities of agents are to be preserved via restrictions on their communication flows [41].

There are several methods and approaches for the application of MAS. Perez-Pons et al. [42] have compiled a survey that focuses on the classification of MAS and the methods that are widely used in research. It highlights that despite its broad applicability, MAS still faces many challenges. One example is coordination between agents and security, and task distribution among them. In general, this requires agents to be able to interact, negotiate, and coordinate to achieve common goals. One of the most commonly studied and extensible deliberative architectures in this regard is the Belief, Desire, and Intentions (BDI) model utilized in this paper. It is considered as the most widely used because it is based on a philosophical model associated with human thought. The work distinguishes eight different approaches among the multi-agent methods:

- **Gaia:** provides a set of models used in the analysis and design phases of MAS development, and is undoubtedly considered one of the main MAS development methods [43,44].
- **Ingenias:** is based on the well-known and established software development process, the *Unified Process*, and the definition of metamodels [45].
- **Mase:** uses a set of graphical models to describe system goals, behaviors, types of agents, and agent communication interfaces [46].
- **Tropos:** is an agent-oriented software development methodology based on two main features [47,48].
- **Prometheus:** is specifically designed to build intelligent agents [49].
- **Passi:** is a step-by-step methodology for the design and development of multi-agent partnerships, with the integration of design models and concepts from software engineering approaches using the UML notation [50].
- **Decaf:** is a flexible MAS, i.e., a set of software tools for the rapid design, development, and execution of intelligent agents for complex software systems [51].
- **Retsina:** is a form of a community of peers committed to peer-to-peer relationships of system agents [52,53].

In addition to these methods, various types of communication patterns between agents have evolved over the years in the form of Speech Acts, Messages, and Blackboards [42].

Leitão et al. [54] discuss that in addition to these approaches and methods, agents can support issues such as the digitization and implementation of self-properties; in particular, self-organization, self-adaptation, self-optimization, and self-healing, which are central to the philosophy of Industry 4.0. Karnouskos et al. [55] identified the change in the agent definitions, developments, and solutions over the years, as well as areas that current efforts could focus on. It is highlighted by Karnouskos et al. that industrial agents have the potential to contribute significantly to several challenges that have already been identified for CPSs. Industrial agents and CPSs are intertwined in the context of Industry 4.0. For example, with regard to Cyber-Physical Production Systems (CPPSs), design patterns, metrics, interfaces, and distributed intelligence are explicitly explored. It is rehashed that MASs have properties in common with CPSs and can endow them with various capabilities to achieve complexity management, decentralization, intelligence, modularity, flexibility, robustness, adaptation, and responsiveness, and to perform system functions collaboratively [54]. However, with the emergence of the CPS and its application in various domains (e.g., energy, manufacturing, logistics), a new opportunity for agent-based approaches has emerged [55]. MAS will play an essential role in this perspective by providing a new and alternative approach to developing intelligent and adaptive systems based on the decentralization of control functions. In particular, aspects such as modularity and autonomous functions, smooth migration, human-in-the-loop, and simulation can benefit from the use of agent technology [54].

In academia, various design patterns for MAS are provided for the integration of MAS, e.g., in manufacturing. The work of Cruz Salazar et al. [56] classifies and compares these design patterns. The main contribution is a CPPS architecture that meets the requirements of the smart factory era and RAMI 4.0. To meet the CPPS and RAMI 4.0 requirements, a general structure for the AAS based on the MAS can be developed. For this purpose, the proposed Industry 4.0 components for CPPS represent an abstract form that defines real objects. This paper assumes that a physical asset type is controlled by an open control architecture (e.g., Programmable Logic Controller (PLC)) that implements lower-level programming codes (e.g., IEC 61131-3). From the analysis of the design patterns, a CPPS architecture was identified for the manufacturing control of Industry 4.0 components, considering RAMI 4.0, and based on four subagents. These are the *Resource Agent*, the *Process Agent*, the *Agent Management System*, and the *Communication Agent*. According to the proposed design pattern, these subagents should be considered as mandatory for the agent-based CPPS architecture, since each of them performs basic functions.

Verbeet and Baumgärtel [57] show relationships with other IT systems for the design of an MAS. For this purpose, BDI agents are classified in RAMI 4.0, and guiding questions are defined that must be answered in order to make a classification so that the BDI agent has the necessary capabilities to fulfill its tasks. The authors highlight that BDI agents are not a one-size-fits-all solution to all software development problems. They need to solve the problems of digital representation and networking through their mere existence. However, they are an approach to design architecture and applications for Industry 4.0 systems, and thus they enable the integration of technical and methodological solutions [57].

The literature around agents shows that MAS can be considered as a partial aspect of the realization of Industry 4.0. However, these approaches cannot be understood as the sole realizer, but they can be a part of the DT. The characteristics and properties of agents and MASs must be integrated into them. Some work in recent years has focused on this direction, which will be examined in more detail in the following section.

2.3. Digital Twin Realizations through Multi-Agent Systems

After a brief introduction to the literature on MAS, this subsection focuses on implementing DTs using MAS for realization. Melo et al. [58] consider DTs as a way to drive the digitization of production assets and to cover the levels defined by RAMI 4.0. This paper shows a possible approach to combining the DTs of several assets with a MAS. For this purpose, the DT is analyzed within the paper according to the three dimensions of RAMI 4.0. It focuses on the inherent capabilities of MAS to support the development of distributed and proactive ecosystems of the DTs, covering lifecycle management and meeting the requirements of Industry 4.0-compliant solutions. In this context, the agents focus on providing collaboration and adaptation capabilities. In particular, the use of MAS technology as the infrastructure for the distributed DT system transforms the typical passive and reactive approaches taken by AAS and DTs into a proactive approach by ensuring:

- The sharing of data collection and intelligence capabilities across multiple DTs,

- The execution of collaboration models,
- The evolution and reconfiguration of the system based on emergent and self-organizing processes that use a plug-and-play strategy on the fly.

Implementing this distributed DT architecture requires an individual interaction of the DT according to a collaboration scheme. MAS is suitable for supporting the challenge of implementing collaboration models between these DTs, and to realize functional parts of the collaboration. MAS fits well with this distributed nature of DTs, and with its inherent properties of autonomy, cooperation, and intelligence, it helps to build a collaborative ecosystem for DTs.

Another work of these researchers discusses how MAS technology can be used to realize AAS by embedding its features into AAS functions and extending them by introducing intelligence and data analysis functions [59]. This paper discusses how MAS technology can realize AAS by mapping its features to AAS functions and extending them by introducing intelligence and data analysis functions. In addition, other aspects are also considered, such as the fundamental concepts supporting the development of an agent-based AAS, namely reference architectures that are compliant with Industry 4.0 solutions such as RAMI 4.0, as well as holonic principles and DT federations to build organizational structures [59].

A further contribution of them shows the use of MAS to implement AAS functions, taking advantage of their inherent characteristics such as autonomy, intelligence, decentralization, and reconfigurability [13]. In this context, the mapping between AAS functions and MAS characteristics, and the challenges for this implementation are presented. The applicability is illustrated by digitizing an inspection cell consisting of a robot and several console products, using MAS technology.

Lopéz et al. [60] present a four-layer architecture for implementing Industry 4.0 components based on industrial agents; two of them are integrated with the agent-based AAS, and the other two with the physical asset. The four-layered architecture presented by the authors proposes to use industrial agents to integrate AAS and physical assets. Layering allows for the abstraction of different aspects of the Industry 4.0 component, dividing the effort required to address the implementation requirements. The first two layers implement the AAS of the Industry 4.0 component, which has been extended to include communication and distributed decision-making functions. The upper layers are implemented in an agent. The other two layers are focused on managing the system. These lower layers are deployed at the edge level to meet real-time requirements. This paper also proposes an integration methodology to describe the steps for applying the architecture. This methodology is illustrated by a case study in which a robotic arm is integrated with an agent-based AAS.

Vogel-Heuser et al. [61] present a different approach for using DTs, especially the AAS, to realize MAS in a production context. For this purpose, a parser automatically extracts relevant information from DTs and initializes the individual agents using the MAS framework PADE. The work highlights that MASs provide a suitable architecture for creating decentralized and highly adaptable autonomous production systems, while DTs have the potential to put this vision into practice by providing standardized models, Application Programming Interfaces (APIs) access, and data exchange. Intelligent agents in an MAS within a production context require various types of information to make autonomous decisions. The authors have assumed that the products are described in terms of the required production processes. To remove this assumption, an automatic match of product characteristics with production processes would be required. Only when the knowledge base is up to date will the agent be able to make appropriate decisions, possibly using a more complex objective function that reflects criteria such as financial cost and quality, rather than just time. To achieve this, the DT must be coupled to the asset, and the two must be constantly synchronized. Since agents have up-to-date environment models, these could also be used to feed updates back into their DTs.

Ambra and Macharis [62] demonstrate an initial proof of concept for DT solutions for long-distance transportation by connecting real-time data feeds from the physical system to a virtual geographic information system environment using *SYMBIT*, which can be used for real-time synchromodal deliveries. *SYMBIT* is a computer model that generates or reproduces data through agents requiring a schedule, behavior rules, and a certain level of knowledge. The paper explores the DT concept and its potential role in synchromodal traffic. From a methodological point of view, agent-based modeling can simulate the availability or exchange of information that is associated with the reactive behaviors of the agents triggered by it. This capability has been tested in the *SYMBIT* model in previous work by exposing static and dynamic solutions to disruptions, and new incoming orders where individual agents reconfigured themselves based on their positions in space and time [63,64].

Dittler et al. [65] propose a concept for continuous model adaptation in the DT of a modular production system during the operational phase. The approach shows that agents are well suited to realize automatic model adaptation as a service of an IDT. For this purpose, a centralized MAS is used to provide DTs with intelligent capabilities. The background of this application takes place for problem-solving at the shop-floor level. As a positive side effect, the approach supports the plant operator in understanding the reason for the reality gap.

All approaches shown here only deal with applying the DTs in the manufacturing context. DTs and MAS have not yet been considered in other domains. Furthermore, most statements and approaches are limited to internal company processes or use cases. Although AAS is also seen as an implementation option for DTs between companies in some areas, this would require entire industries to agree on a standard language to be used, which is doubtful.

3. The Digital Twin Reference Model

After the related work concerning DTs, MAS, and their overlaps has been illustrated, the first hypothesis can be examined. There is a general lack of a consolidated and consistent view of how the concept of the DT is emerging to satisfy as many use cases as possible. Therefore, many characterizations about DTs have evolved over time, leading to discrepancies with its initial goals [14]. The range of use cases and purposes that can be implemented by DTs is extensive. Sjarov et al. [24] differentiate between the exploratory and decision-making purposes of DTs. Thereby, all life-cycle-dependent tasks and demands, such as the improvement of the overall understanding of the system, the mirroring of assets' live states, future predictions, the derivation of KPIs, the improvement of traceability, troubleshooting, and reductions in cost can be covered. However, all of the purposes shown are within the company's boundaries and are considered as being close to the production environment. Higher-level functionalities, such as intelligent orchestration within a marketplace scenario, as exemplarily shown by Lehmann et al. [20], go beyond these companies' standard applications. Similarly, such assumptions do not consider who has sovereignty over the physical resource nor how the superimposed interaction is regulated. The associated security risks must also be taken into account. Different vulnerabilities arise at each level of such architectures, since a wide variety of compromisable interfaces and services are needed to provide valuable DTs [66].

Hypothesis 1. *Digital Twins must be distinguished between their levels of application, correlated to the proximity to the production process.*

A methodology for this desired distinction during the operational phase is constituted by reference models because not all types or purposes of DTs are clearly assignable. In the literature and industry standardization, there have already been attempts to define and to classify DTs through reference models.

ISO 23247 [67,68] defines a reference frame for how DTs can be developed, especially in the manufacturing domain. Particular emphasis is placed on basic characteristics of DTs and the delineation of various tasks, such as the communication link, the form of representation (operation and management, application and service, and resource access and interchange) to connect all system entities to the upper user applications. All properties according to the definition are taken into account and applied for the mapping of a machine representation (as observable manufacturing elements) within a manufacturing company. Lu et al. [69] present a reference model to help with the standardization of research and development. They differentiate into three parts: an information model to abstract the specifications of the physical object, a bidirectional communication mechanism, and a data processing module to extract information and to map the live representation of the physical object. The structure is strongly reminiscent of Grieve's conceptual structure, which is shown in Figure 1. Aheleroff et al. [70] are utilizing RAMI 4.0 to introduce a reference model for mapping DT layers, the level of integration, and the value life-cycle. While RAMI 4.0 differentiates between the properties of a machine on the level dimension, and reflects the information model of automation on the hierarchy level dimension and the product creation process on the life-cycle value stream dimension, the reference model of Aheleroff et al. deviates significantly from the original intention. The assignment of these DT characteristics is not suitable for the dimensions addressed by RAMI 4.0. Moreover, the already existing AAS as a practical implementation of DTs within RAMI 4.0 is not even considered. Bevilacqua et al. [71] attempt to establish a reference model for risk prediction and prevention. They focus on the following four elements: physical space in the processing industry, a communication system, the DT itself, and user space as an interface to humans. Alam et al. [72] proposed a reference model for cloud-based cyber-physical DT systems. They analytically describe how the interplay can be realized via peers, and an intelligent service layer on top of the physical and cyber things layers. Melo et al. [58] have already performed a mapping of the different RAMI 4.0 layers to the AAS [58]. However, the approach does not observe a cross-organizational approach to the orchestration of DTs.

All identified reference models focus on basic or other problem spaces, which are needed in the further development of the DT concept. None, however, directly address the principles of the operational phase of Industry 4.0 and their division into higher-level orchestration mechanisms that can simultaneously stimulate Industry 5.0 premises. To do so, the common concepts of digital transformation will be correlated and mapped to the DTRM so that a distinction can be drawn between the provided DTs. The most prominent and internationally recognized general reference models are RAMI 4.0, originally developed by the German Plattform Industrie 4.0 association, and IIRA, which was developed by the Industrial Internet Consortium (IIC). Figure 2 illustrates their correlation to establish the DTRM.

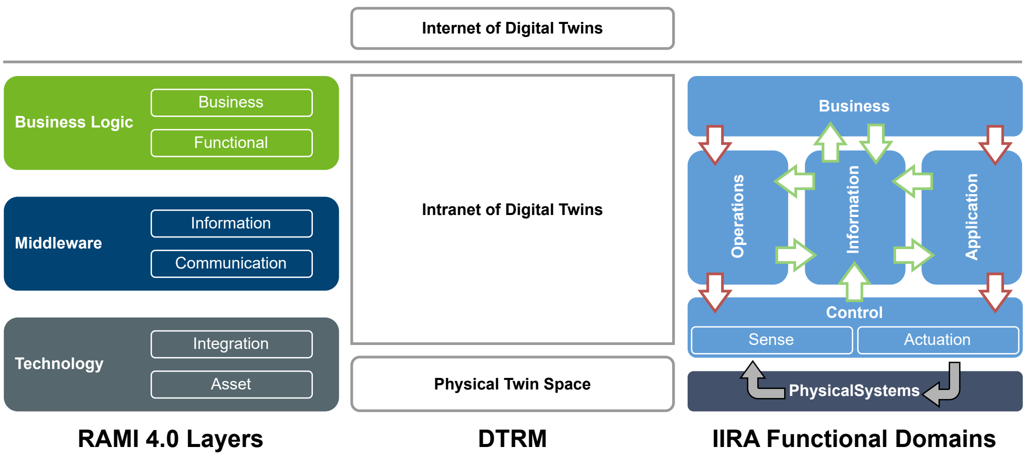

Figure 2. Comparison of RAMI 4.0, IIRA, and DTRM During the Operational Phase.

The model comparison as shown does not attempt to give a fixed methodology for a technical implementation as attempted by the other models, but to sharpen the understanding of the context in which varieties of DTs can be operated and deployed. Likewise, the model does not address a lifecycle horizon, but rather the operational phase in which active data are generated for processing and adaptation within the physical process. However, depending on the application scenario and realization methodology, this would be equivalently transferable to other life cycle phases.

On the left side of Figure 2 there is an excerpt from RAMI 4.0. Actually, RAMI 4.0 is the attempt to amalgamate the most important aspects of Industry 4.0 into a three-dimensional model (architecture layers to establish business goals, product lifecycle, and the technical hierarchy of automation systems) and simultaneously establish a common perspective and understanding [73,74]. On the contrary, IIRA, partially depicted on the right side of Figure 2, focuses on different stakeholder perspectives across industries through four viewpoint layers (*Business Viewpoint* addressing business objectives, *Usage Viewpoint* achieving expected benefits for users, *Functional Viewpoint* caring about the structure and interaction of all participating entities, and *Implementation Viewpoint* realizing the technical aspects) [75]. Due to the limitation of the operational phase and the fact that DTs are the key to vertical integration within automation systems, only the architecture layer dimension of RAMI 4.0 and the *Functional Domains* of IIRA's *Functional Viewpoint* are considered for deducing the DTRM.

The bottom layer of all models is dedicated to the physical space in which the *Physical Systems* or *Assets* are located. An *Asset* includes physical and nonphysical objects such as production machines, engineering documents, or services. While the overlaying *Integration* layer in RAMI 4.0 provides physical assets digitally to the superposed layers, the *Control* layer in IIRA is subdivided and is described in higher granularity. It is there for the fulfillment of common functionalities in the control of *Physical Systems*, such as trigger sensing and actuation, offering communication functionalities to the other layers, and entity abstraction by enriching data with contextual information via semantics. Therefore, similar to RAMI 4.0, it covers the closed-loop control of the *Integration* layer and parts of the *Communication*. The *Communication* layer in RAMI 4.0, which facilitates standard communication interfaces to further process data, is an omnipresent key feature in IIRA that was initially realized at its *Control* layer and fully exploited at the *Information* layer. While all green arrows shown represent the data flow, starting from the *Physical System* and its *Control*, the red arrows represent the downward command flows. The *Information* layer is almost equivalent in both models. RAMI 4.0 describes the services and data of an *Asset* semantically, whereas IIRA transforms, persists, and models or analyzes data for acquiring higher-level intelligence. RAMI 4.0's *Functional* level is the environment for applications to perform the tasks required to achieve the business process objectives. For this purpose, the services and features of the *Assets* are offered here. At IIRA, these tasks are performed by the *Operations* (management and operation, prognostics, monitoring, and optimization for the seamless operation of the *Physical Systems*, especially to achieve a higher-level goal through groups of plants) and *Application* layers (use case specific logic to achieve a system-wide operational optimum including logic, rules, APIs, and user interfaces). The *Business* levels are again equivalent to each other, and describe business-relevant functionalities, achieving the overall business objectives and linking different business processes under legal and regulatory constraints [73–76].

In the DTRM, the *Physical Twin Space* is used to accommodate physical devices that are analogous to the *Asset* and *Physical Systems* layer, which, for example, could be physical production machines such as drilling or milling machines, or, in a more open view, physical Internet of Things (IoT) devices such as a coffee machine. In order to make these physical assets available to the upper reference model layers, they need to be connected digitally through the *Integration* or *Control* layers. This includes a closed loop control with suitable interfaces, for example, at most industrial applications, PLCs and their often integrated Open Platform Communications Unified Architecture (OPC UA) servers; for IoT applications, these functionalities could be realized by microcontrollers with a Message Queuing Telemetry Transport (MQTT) interface. To provide an initial added value in further data use, the connected interfaces must be made available with semantic information, or directly via an information model to attain machine-understandable interoperability. All these operations take place exclusively in the *Physical Twin Space* since the physical assets and their necessary *Integration/Control* are to be seen as a completed unit.

The interconnection to the *Intranet of Digital Twins* enables the fully fledged and dedicated representation of the physical asset in the digital space. It contains not only all the comprehensive data required for value-bringing DT architectures, but also all the upper goals and layer premises of both established reference models. However, not only the representation as such is in the center of the *Intranet of Digital Twins*, but also all the realization technologies required for it. Similar to the *Communication* and *Information* level, the DTRM requires a kind of middleware to interconnect heterogeneous devices harmonizing their data on the one hand, and for the actual structure and status representation of the physical asset on the other. DT-representing middleware concepts (i.e., Eclipse Ditto, AWS IoT, Azure IoT, or self-developed frameworks) could also be used across domains, not only for automated production systems, but for completely independent use cases towards IIRA, for implementation. The *Functional, Operations, Application*, and the overlying *Business* layers are founded on these DT data and process them to achieve overall goals. These value-adding tasks facilitated by DTs, such as predictive maintenance, data analytics, forecasting, or simulations, are located here within the boundaries of a company. For instance, existing approaches, such as the AAS in the manufacturing domain, could be used to map the *Intranet of Digital Twins*, since all levels of RAMI 4.0 can already be realized in this, depending on its characteristics [77]. In this context, Melo et al. [58] show how RAMI 4.0 and the AAS correlate to each other, enabling a fully mapped DT representation in Industry 4.0. With regard to IIRA, this implementation approach can now also be transferred to all other use cases outside of manufacturing by applying the DTRM.

For confidentiality and security reasons, these information and DT processes may not be published across company boundaries. While both established reference models already provide space for analysis across plant associations, they are fixed in their direct interaction and autonomy. This imposes limits on the higher-level cross-enterprise orchestration of DTs. Therefore, a higher-level space must be created for overarching orchestration tasks during the operational phase of such systems. IDTs could be brought to life in this space and empowered to collaborate. This requires the formation of a representative subset of the DTs located in the *Intranet of Digital Twins* to provide intelligent entities within the *Internet of Digital Twins*—***IoDT***. For example, an interdisciplinary marketplace scenario of negotiating IDTs could be built across company boundaries under rigidly defined regulatory conditions. The basis for negotiation and the goals of a marketplace of IDTs could contribute to sustainability, resilience, and human orientation from an Industry 5.0 perspective. For example, objective functions and outcomes could be related to carbon footprint, resilience, and optimized supply chains. These correlations lead to the second hypothesis that a possible orchestration logic for the IDTs of such a collaborative network can only come about by means of MAS paradigms [13,15,16,57].

Hypothesis 2. *Implementing IDTs within such a reference model requires symbiosis with Multi-Agent System (MAS) paradigms.*

To leverage IDTs through MAS paradigms, the correlation in Table 1 can be considered. This compares the characteristics of an IDT established by Grieves [12], with the corresponding characteristics of MAS outlined by Sakurada and Leitão [59] and Wooldridge [40]. In comparison, it is apparent that both approaches do not act far from each other and can be easily amalgamated. As mentioned in related studies, there are already some attempts to raise the AAS to an active level by implementing discrete MAS methodologies. However, a more open approach will be adopted in the subsequent part of the paper to realize an Industry 4.0- and 5.0-compliant orchestration between IDTs within the ***IoDT***.

Nevertheless, the challenges identified by Melo et al. [58] to develop a distributed, agent-based DT system that can be summarized as six key aspects: (1) design and engineering, (2) compatibility and interoperability, (3) decentralization, (4) emergence and self-organization, (5) simulation and data analytics, and (6) infrastructure and platforms, also apply to this approach.

Table 1. Correlation of Intelligent Digital Twin Characteristics According to Grieves [12], and Matching Multi-Agent Systems Characteristics according to Sakurada and Leitão [59], and Wooldridge [40].

Characteristics	Description	Matching MAS Characteristics
Active	IDTs should actively provide information that is currently needed. This enhances collaboration with humans and other machines.	Through a permanent perception of the environment and continuous exchange with other agents, processed information can be actively incorporated into collaboration processes.
Online	In order to ensure active interaction, the IDT must be online and have a continuous connection to the PT's respective environment perception.	Agents interconnect the physical space and digital space for perception and interaction purposes.
Goal-seeking	The goal-seeking, which has always been present, is not to be carried out by human intervention as before, but with the support of the IDT.	Agents can interact autonomously with humans as well as machines or other agents through their social abilities to achieve an overall goal.
Anticipatory	The IDT anticipatorily adapts its actions and goals to its self-predicted future based on all its accumulated information and experience.	Agents have the ability to learn, share their knowledge, and adapt their behavior to their future goals.

4. Architecture

After the correlation of IDT characteristics and matching MAS characteristics have been outlined, a conceptional architecture based on the elaborated DTRM will be introduced.

According to the presented DTRM, the architecture shown in Figure 3 is divided into three layers: Physical Twin Space, the Intranet of Digital Twins, and IoDT. These layers are structured according to organizational areas, as well as within a company environment, as within a superimposed cross-company environment. While the Physical Twin Space and the Intranet of Digital Twins refer to one organizational area, the IoDT provides the aggregation of a subset of the underlying organizational areas. On the lowest level of the architecture, the so-called Physical Twin Space, the physical devices are located, which are generally referred to as PT1 - PTn. According to Grieves' twinning paradigm, these are uniquely associated with their digital equivalents, which are contained in the Intranet of Digital Twins and are connected through one-to-one relations with their PTs. Based on the occurrence of asynchronous use cases, the bivalent data pipeline linking the twins is self-driven to ensure data synchronicity.

The DTs in the Intranet of Digital Twins are located close to production and are mainly passive in nature, as mere representations of their PTs. Their functionalities lie in the aggregation of all data, as well as static and structural information from the Physical Twin Space, and the provision via standard communication interfaces for the upper hierarchical level in a harmonized manner. In addition to DTs as passive state representations of their PTs, IDTs could be utilized in this layer as well. Smart applications facilitate the intelligent value-bringing use of the provided DT data, i.e., for the objectives of predictive maintenance, simulation, what-if scenarios, or the digitization of business models.

Further, a knowledge graph representing the entire semantic context of the facilities within a company is deployed at the Intranet of Digital Twins, enabling interoperability between the magnitude of different DTs to be stored and queried independently. However, the ability to collaborate for DTs in this layer is limited due to the boundaries and limitations within the enterprise. This limitation can be overcome with the help of the overlaid DTRM layer.

The IoDT is located above the Intranet of Digital Twins and includes the Marketplace of Intelligent Digital Twins. In this architectural part, a dedicated marketplace knowledge graph, the IDTs, which are generically referred to as IDT 1-IDT n and the Order Demand IDT 1-IDT n, are established. The Marketplace of Intelligent Digital Twins provides the setting and environment for collaboration and takes over a framework with extensive orchestration tasks to bring DTs to life. Therefore, it interfaces with the Intranet of Digital Twins so that cross-company IDTs can gather in the marketplace for intelligent collaboration.

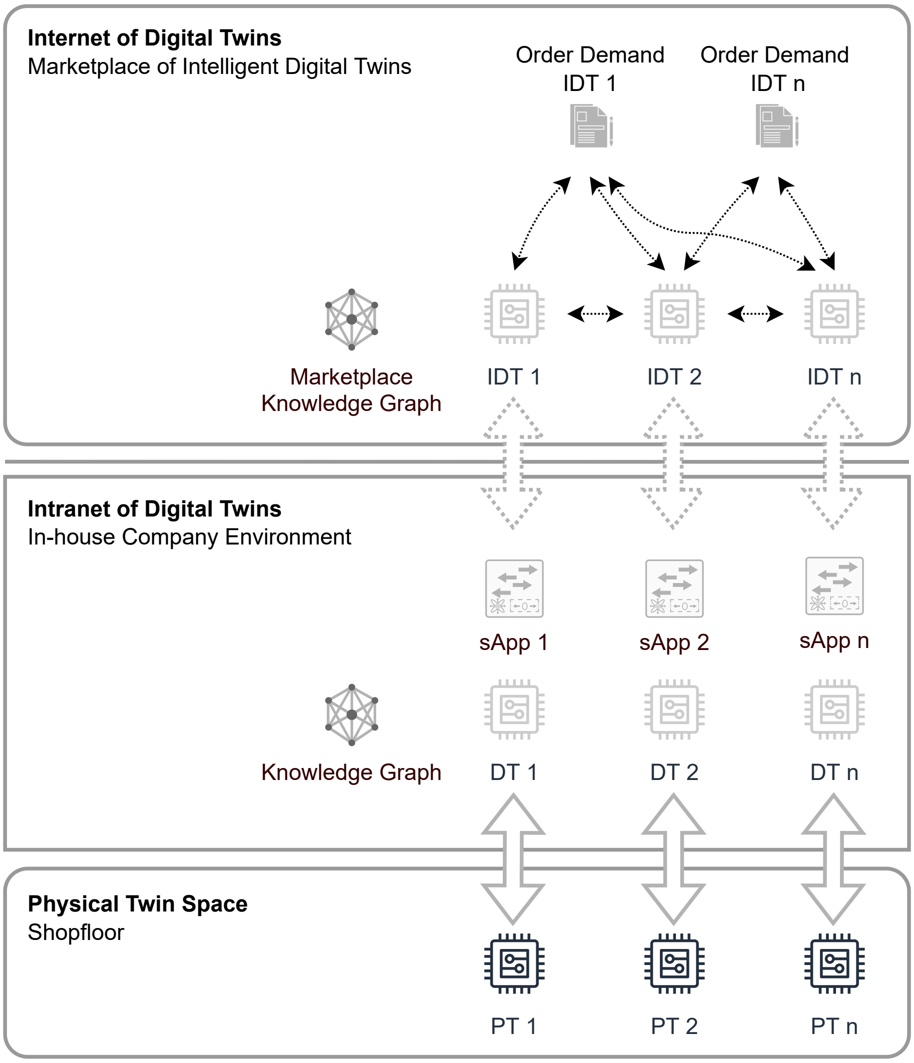

Figure 3. Conceptual Architecture.

In order to participate in the marketplace, DTs from the layer below must be registered. This is achieved through the usage of smart applications. These take over the registration and deregistration of DTs in the marketplace. In order to register a DT, they extract the properties of the DTs that are necessary for collaboration and prepare them in an adequate manner for the marketplace. With this subset of specifications, the orchestration in the Marketplace of Intelligent Digital Twins provides an IDT. This IDT has a subset of the capabilities of its associated DT and takes on the role of a proxy twin. By constraining the capabilities and information provided by the DT, only the data needed for collaboration are disclosed, and the security of the inner company's internal, sensitive data is maintained. Nevertheless, the IDT is connected to its DT and thus has an indirect connection to its PT. With the provision of the IDT, it actively offers its semantic information to the marketplace knowledge graph in order for them to be used in subsequent queries for collaboration purposes.

The marketplace knowledge graph provides an environment for the storage of cross-company knowledge using an ontology. It serves the holistic organization of the semantic representations of the DTs and their underlying information model. This knowledge-based approach enables the derivation and inference of the complex interrelationships of the overall system, and thus it can facilitate collaboration among DTs. Therefore, IDTs will be enriched with intelligence through the marketplace knowledge graph.

Once the IDT is fully instantiated, it starts searching for its target to offer its service. To achieve its goal, it needs cooperation partners that necessitate its offered service. These are the Order Demand IDTs, referred to as Order Demand IDT 1 - Order Demand IDT n. Order demand IDTs are created via an interface to the Marketplace of Intelligent Digital Twins with a registration message. The framework receives the registration message with the demand and provides an Order Demand IDT. It intermittently queries the marketplace knowledge graph with the set requirements to provide the requester with possible collaborators. Based on this response, the negotiation between the IDTs and the Order Demand IDTs can take place. The collaboration of both parties is completely autonomous and ends with the awarding of the order by the Order Demand IDT. Orders are placed on the basis of requirements underlying the Order Demand IDT and its goal. After placing the order, the IDT reports it to its DT, which then forwards the order to its PT while the Order Demand IDT waits for its order to be manufactured. In addition, an evaluation of the IDT's actions by the DT is utilized to optimize future negotiation patterns between the IDTs.

It must be particularly emphasized that the writing permission for the managed resource, the PT, lies in the authority of the DT within the Intranet of Digital Twins. The DT is aware of all the actual conditions and information about the resource, and takes over the entire management of it. As described before, the IDT is a subset of the DT and acts as a proxy. The intelligence of the DTs is not totally dependent on the overarching processes of the IoDT, but it can also reside within the Intranet of Digital Twins as described. Nevertheless, the focus of this architecture is on external orchestration across company boundaries toward the realization of novel industrial paradigms.

5. Use Case Description—Choreography of Production Processes

Based on the architecture for the realization of the DTRM and IDTs, a use case for the choreography of production processes and their relevant components will be introduced subsequently. First, the physical structure of a demonstrator is described, followed by how production processes can be established.

The paper uses parts of the Industry 4.0 fischertechnik models previously presented by Lober et al. [78] as a physical use case. Specifically, a cycle line with two processing stations controlled by an IoT-capable Wago PLC was utilized. To achieve the highest possible degree of freedom, orchestrability, and reconfigurability, the program logic is based on the principles of skill-based engineering.

Figure 4 shows the topologies of the different modules. Four conveyor belts, *C1–C4*, connect the product flow with the other components. The position sensors *S11, S12, S2–S4* are directly attached to the belts to detect the position of the workpiece. At each corner of the U-lines, there is a pushing device, *P1, P2* for the change of direction. Sensors *PS11, PS12*, and *PS21, PS22* are also installed to detect the piston position of the pusher. To simulate the work steps, production machines *M1, M2* imitate a drilling or milling process. The workpiece moving in the direction of the arrows thus has various possible product configurations that it can assume.

For better comprehensibility, a production process of a complex product in this use case can be subdivided into different stages of manufacturing. It can be extended to further stages as required. The variable applicability of the production machines allows each production step to be located at any point in the product's production process. In the use case, the product is specified via its knowledge graph by defining at which production step the technical process is required. The respective production resources then report to the particular production step that they could fulfill. Figure 5 shows an example three-step production process. In this case, the product has the following specification for the individual steps A–C.

- Step A: Thrilling process,

- Step B: Milling process,
- Step C: Drilling process.

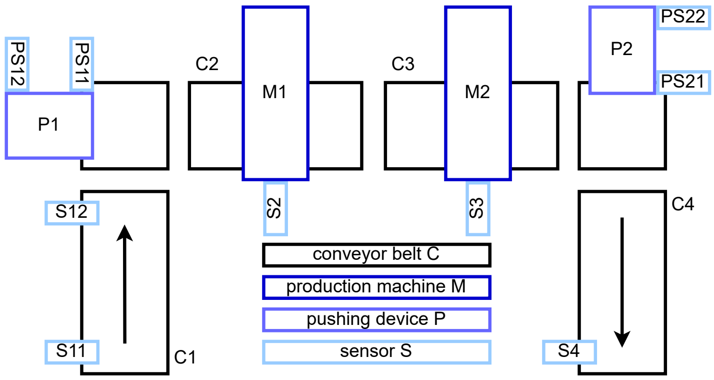

Figure 4. Use Case Setup According to Lober et al. [78].

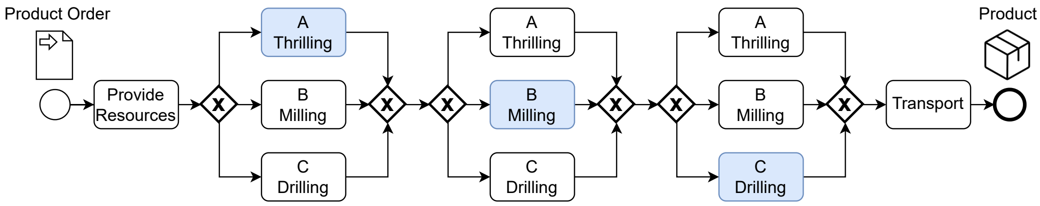

Figure 5. Production Flow of a Skill-based Manufacturing Scenario.

On the basis of this specification, the corresponding available resources that can perform this process step are selected. This results in the production and logistic processes for the product to be manufactured.

6. Implementation and Proof of Concept

After the architecture and the use case have been presented, the implementation follows subsequently. Therefore, the previously described Industry 4.0 demonstration production line is utilized to verify the architecture's functionality in a brief proof of concept. The two machines of the production line simulate production processes and are integrated into the Physical Twin Space of the architecture. While M1 simulates a drilling service, M2 is dedicated to a milling simulation. The PLC provides an IoT communication interface that enables bidirectional linkage as DTs.

As the main component of the Intranet of Digital Twins, the open-source project Eclipse Ditto [79] is utilized as a middleware layer. The microservice-based project offers the functionality to provide DTs of physical devices through multiple interfaces with integrated management. The DTs are represented at a high abstraction level through JavaScript Object Notation (JSON). The system can influence the physical space through backend commands, and offers the greatest possible freedom, with individual scalability, space-saving deployment, and the segmentation of the various functions. In addition to Eclipse Ditto as the middle layer, an AAS-based system, i.e., Eclipse Basyx, could also be used. In this context, the selection is not important, as only an abstraction layer is needed to accommodate the DTs. In order to provide DTs within Eclipse Ditto, the production line PLC must send a provisioning message with the given information model. For the sake of simplicity, the production line provides one DT as equivalent to an accumulative DT, consisting of one DT for M1 and one for M2. These are referred to as DT M1 and DT M2.

To further continue with the interconnection to the Marketplace of Digital Twins within the superordinated IoDT layer, a smart application within the intranet takes over the registration of the production line's DT as a proxy. This smart application is (in this case here) is automatically triggered by the DT's provision, extracts the information necessary for collaboration from the DT's information model, and transmits it to the Marketplace of Intelligent Digital Twins via Hypertext Transfer Protocol (HTTP). The smart application's trigger can be set to manual as well, e.g., in the case of a sudden underutilization of a specific production machine within a company, not covered by the availability mechanisms, the lack of automation in the company, or jurisdictional demands. Figure 6 illustrates the sequences within the marketplace. This sequence diagram is limited to the main processes for reasons of simplicity and clarity. A full illustration would be too extensive and unclear. However, all implemented processes are explained below.

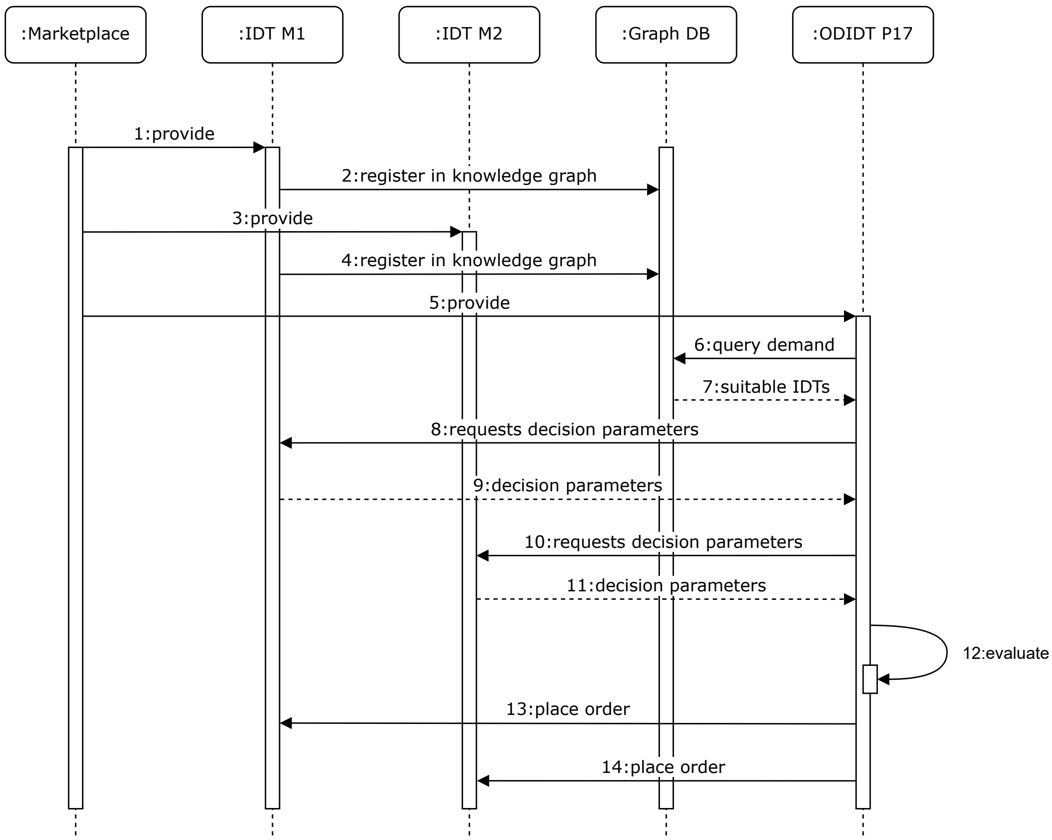

Figure 6. Sequence Chart of Selected Relevant Parts of an IDT Collaboration in an Agent-like Manner, beginning at the IDT Provision and Ending at the Placement of an Order.

The Marketplace of Intelligent Digital Twins is a microservice-based and dynamically event-driven framework. This is implemented as Docker containers, which are based on Python. Individual microservices take over various tasks such as the registration or provision of IDTs, as well as orchestrating the IDTs. The marketplace receives the registration from the smart application for DT M1 and DT M2. Subsequently, it provides an IDT for each machine in the production line. These are designated as IDT M1 and IDT M2. The provision of the IDTs proceeds in independent threads within the main container. These threads operate completely autonomously from each other. The link between the threads is the orchestration part of the framework. It takes on the role of a mediator that forwards messages between the threads, from the threads to the DTs, or from the DTs to the threads. The DTs on the other hand route messages from the PTs to the framework, and vice versa. This enables communication for negotiations within the framework, and externally to the intranet DTs thus indirectly to the PTs. In this way, the IDTs have a continuous connection to their PTs and can interact with them. These IDTs are based on Python classes that contain a specific flow method that is executed in the threads. The instantiation of this class requires information for the underlying DT, which is transferred to the marketplace via the smart application and is based on a specific representation in JSON format. These representations contain all the information needed for their flow method and the semantic description of their capabilities. The flow method enables IDTs to actively provide information about themselves and to seek their goal of offering their service. After the IDTs provision, they register themselves actively, autonomously, and automatically with their semantic representation in the Marketplace Knowledge Graph via a SPARQL query. This knowledge graph is implemented by GraphDB. GraphDB is a triplestore that has a faster execution speed of queries compared to alternatives such as Jena Fuseki or 4Store. The GraphDB execution speed is slightly slower, at up to 1,000,000 triples compared to Virutoso, but it exceeds this after that [80]. Furthermore, GraphDB uses SPARQL, a common query language, while other alternatives such as Neo4j or JanusGraph use Cypher or Gremlin, which are less common [81]. Based on the arguments outlined above, GraphDB was implemented as a marketplace knowledge graph due to its better performance and common query language. After registering themselves, the IDTs wait for possible collaboration partners to achieve their goal of offering a set of production services.

Besides the class for the IDTs, there is another Python class contained in the framework. This is needed to provide Order Demand IDTs. Their instantiation also requires information about the underlying product and its semantic description. Similarly to the IDT class, the Order Demand IDT class contains a flow method that enables it to seek its goal of being produced. Order Demand IDTs are also executed in separate threads in the main container and feature a continuous connection to their PTs as well. The framework offers an HTTP API for instantiating Order Demand IDTs. This receives the specification of a product in JSON format and provides an Order Demand IDT. In this implementation, Product17 sends its registration to the marketplace API to be provided with Order Demand IDTs. The structure of this JSON is shown in Figure 7.

It is divided into two parts, firstly, the general information, and secondly, the demand itself. The general information includes the *thingId* as a unique identifier, the *thingType*, the *task* to create an Order Demand IDT, and the *negotiationAwardCriterion*. The *negotiationAwardCriterion* forms the basis for the Order Demand IDTs negotiating within the collaboration process, and is set to the price (there could be various other as well as multi-level negotiation objectives instead) for this use case. Other examples are the delivery date or the lowest price with the shortest delivery time. In the demand section, the individual process steps required for manufacturing are specified. The process steps are then divided into the required service, the material to be processed, the geometry, and the dimensions of the geometry. In this implementation, *Product17* requires two process steps for its production that match the skills of the production line. These consist of a drilling and a milling service with the dimensions displayed. After the marketplace receives the registration of an order, it provides an Order Demand IDT, referred to as ODIDT P17. The ODIDT P17 then actively, autonomously, and automatically queries its demand for its individual process steps in GraphDB. The response from GraphDB where IDTs are able to realize the demand (or parts of it), is transferred back to the Order Demand IDT of *Product17*. At this point, ODIDT P17 begins negotiating with the suitable IDTs. In this implementation, IDTs M1 and M2 match the demands. For the negotiation, ODIDT P17 requests its decision parameters from IDTs M1 and M2. The IDTs provide the Order Demand IDT with the requested information to accomplish their goal of offering their service. With the information, the ODIDT P17 starts to evaluate the possible collaboration partners. Based on the selected *negotiationAwardCriterion price*, the Order Demand IDT selects an IDT for each step and places the orders.

The M1 and M2 IDTs receive the orders and transfer them to their DTs in the Intranet of Digital Twins. The DTs check the utilization of their corresponding PTs and report it to the IDTs. This is required to address unexpected conditions such as immediate own demand for the machine, or in the case where an order has been accepted in the meantime, within the context of another negotiation. If the PT is utilized, the associated IDT must renegotiate with its collaboration partner. The IDT autonomously and actively notifies the Order Demand IDT of the new delivery time and awaits confirmation. The Order

Demand IDT evaluates the new terms, and accepts or rejects them. In the case of rejection, the Order Demand starts new negotiations with the IDTs to meet the required demand. If it is accepted, the new conditions are agreed and subordinate claims emerge. The IDT hands over the order to its DT, which schedules it in the production planning. This implemented renegotiation mechanism enables the mapping of dynamic events and subordinate claims.


```
{
    "thingId": "Product17",
    "thingType": "OrderDemandIDT",
    "task": "CreateOrderDemandIDT",
    "negotiationAwardCriterion": "price",
    "Demand": {
        "Step1": {
            "ProductionService": "DrillingService",
            "TypeOfMaterial": "Metal",
            "Geometry": "Circle",
            "Dimensions": {
                "DiameterHoleResource": 15.0,
                "Depth": 30.0,
                "Thickness": 55.0
            }
        },
        "Step2": {
            "ProductionService": "MillingService",
            "TypeOfMaterial": "Metal",
            "Geometry": "Rectangle",
            "Dimensions": {
                "LengthResource": 15.0,
                "WidthResource": 5.0,
                "Depth": 30.0,
                "Thickness": 55.0
            }
        }
    }
```


Figure 7. PDDT Information Model.

In this implementation, the PTs M1 and M2 machines are fully available. Therefore, DTs M1 and M2 confirm the PTs' availability to the IDTs M1 and M2. The IDTs accept the orders from Product17 and transfer the specifications to their DTs in the Digital Twin Intranet. Afterward, the DTs initiate production and the production line begins the simulation based on the placed orders and the underlying parameters. Here, the production parameters are generated by the DTs on the basis of the order's geometrical characteristics and these are passed on to the machines M1 and M2 for simulation. After finishing the simulation, the DTs report to the associated IDTs that the process is finished. The M1 and M2 IDTs pass this message to their collaboration partner, ODIDT P17. When ODIDT P17 has received the confirmation of completion for each production step, it is manufactured and has achieved its goal. Now, ODIDT P17 performs final business logic such as the customer accepting the product or quality assurance, after which it deregisters from the marketplace and the thread is terminated. IDTs M1 and IDT M2 remain on the marketplace for further collaboration. The deregistration process is omitted in the sequence chart in order to maintain clarity and comprehensibility.

The entire production and awarding process runs completely autonomously through DTs and is in a continuous feedback loop with the PTs due to their permanent connection. In this case, there are two production steps, and a simulation is performed on both M1 and M2. However, single-step scenarios can also be mapped with this implementation. In addition, the *negotiationAwardCriterion* can also contain multiple inputs of two or more criteria to use for the negotiation. For several criteria, the Order Demand IDT generates an optimization function according to an underlying weighting for the required criteria, and makes its selection based on this function.

7. Discussion and Evaluation

Based on the two hypotheses defined at the beginning, this paper develops approaches for the realization of IDTs in order to create technological concepts to fulfill the premises of

Industry 4.0 and 5.0. The fine granular differentiation of DTs correlated with proximity to the production process and the application (*H1*) can be achieved by the DTRM. Through the popular reference models of Industry 4.0 and Industrial IoT, the current requirements of the digital transformation, as well as those of the future with Industry 5.0 could be satisfied through the overlay of the IoDT. By aligning the negotiation base with the guiding principles to address future issues such as sustainability through carbon dioxide savings and human-centered machine collaboration, for example, to complete hazardous tasks, or the achievement of maximum utilization through resilient production systems, Industry 5.0 can also be directly targeted. Through awarding mechanisms in the IDT negotiation, these objectives can be freely set. However, to establish a collaboration of IDTs first, it could be proven that symbiosis with MAS approaches (*H2*) is indispensable. The overlaying of the characteristics on IDT and MAS were elaborated and they built up the foundation for a prototypical use case architecture, which will be discussed subsequently.

- *Active:* The IDTs are generated from the DT in the Intranet of Digital Twins, and they are supplied with all data relevant for collaboration. Depending on the workload and the status of the physical machine, the DT updates and provides its IDT as a subset of its available information. These current data reflect the perceived context of the PT, and thus form the basis for negotiating with other IDTs. Once an Order Demand IDT enters the marketplace, the IDTs proactively try to find its production optimum, depending on the desired negotiation strategy.
- *Online:* Due to the available interfaces between the levels of the DTRM, all entities involved are mutually up to date. Since the IDT acts only as a proxy for the DT, it cannot directly affect the PT. This still requires a controlling instance for confirmation, but this is required anyway for functional safety and security reasons.
- *Goal seeking:* Without an overarching goal, intelligence to proactively solve problems would not be needed either. Therefore, the overall goal in the marketplace is to fulfill the optimal production flow of the order demands. Depending on the desired negotiation strategy, the IDT can be provided with various sub-goals regarding its bargaining behavior in a human-like and social manner.
- *Anticipatory:* In order to optimize for future negotiation scenarios, another award mechanism is implemented between DTs and their IDTs. Feedback from the evaluation of the production process provides the basis for self-adaptation to meet future goals more efficiently.

The contribution of the paper shows the importance of interdisciplinary research from different communities. It is through the use of MAS paradigms that DTs can become intelligent. Evidently, the modalities of both worlds can be perfectly synthesized into an overall system. This is independent of the use case and the scenarios. In the present work, a marketplace with negotiating production machines serves as an illustrative object, whereby the attractive aspect of the concept of the DT is its independence from use cases to be applied in every other domain. In particular, the bivalent advancement of two industrial revolutions through DTs as an enabler technology of both brings future added value for resilience in manufacturing. This can be implemented, for example, through the higher utilization of production machines. If specific machines of a company are idle or underutilized, they can register on the marketplace and acquire orders for production within the IoDT. Similarly, companies can find replacement production routes in the event of a breakdown or damage to a specific machine.

As a qualitative evaluation, the comparison to the previous work (Lober et al. [78]) can be considered. Due to the lack of twinning resources and the lack of possibilities regarding industrial scalability, no suitable quantitative evaluation could be performed. In the mentioned preparatory work, the same physical use case setup was employed, but without a proper and separate DT management or reference model. The framework was a NodeRed instance with manually triggered queries on the knowledge graph and the resulting production orders for the physical system. Consequently, no MAS approaches were relevant for it. The product composition was limited to a three-step production process via NodeRed's graphical interface. In contrast, all relevant negotiation and query steps are now performed proactively by instantiating an order demand IDT. An interface can be used to send the demand specification directly to the IoDT, which finally results in the production order after the negotiation procedures. In addition, a feedback loop conducts improvements for future optimization actions. In conclusion, the DTRM-aligned framework behaves entirely autonomously, in contrast to the manually designed preliminary work. Besides saving a lot of time and errors due to no human interference in the ordering and negotiation process, this is also a big step towards later real implementation within a productive environments. Likewise, there is no longer any restriction on the size of the production processes, as before.

Nevertheless, the architectural design offers considerable potential for future improvement, as it was only intended as a proof of concept. From a functional point of view, for example, the other dimensions of RAMI 4.0 and IIRA have been neglected, since the approaches described here relate only to part of the operational phase. To include them, the AAS approaches will be implemented for the Intranet of Digital Twins in the future. A realization via ready-made MAS frameworks, such as JADE/Jadex, must also be investigated. However, it should be noted that the proof of concept demonstrated here does not claim to be industrially feasible, which is why a minimal scope of communication and narrow implementation with as little overhead as applicable is used, so that as few problems as possible of distributed system topologies follow. Furthermore, all aspects regarding security, confidentiality, and functional safety were disregarded. Each individual layer must be adequately secured in accordance with the current state of the art, and protected against unauthorized use. Likewise, such a framework is only viable when there is a sufficiently large number of participating machines and companies. For this reason, validations on an industrial scale are planned for the future. It should also be noted that for the present study, the supply, transport, and logistics processes have been neglected for the sake of simplicity. In order to further focus on human needs and to meet the requirements of Industry 5.0, the approaches of human–data interaction must also be applied more profoundly.

8. Conclusions and Future Work

To adequately address the growing global challenges of the future, appropriate novel technological approaches such as Intelligent IDTs, as described in this paper, must be adopted. It has been shown that the premises of Industry 4.0 and 5.0 complement each other perfectly in their forms of realization through IDTs. As a result, the contribution has been able to promote important aspects in both industrial revolutions—technology-driven value creation and value-driven value creation.

On the basis of the related work regarding DTs, MAS, and the interrelations of both, two initial hypotheses could be derived. These hypotheses were the driver and validation of the necessity for practical elaboration. To distinguish the DTs that are correlated to the levels of application and the proximity of the production process (*H1*), the DTRM could be derived for substantiation. The common reference models of the fourth industrial revolution (RAMI 4.0 and IIRA) could be transferred therein, not just addressing manufacturing environments, but also expanding on the concept of DT in manifold other use case domains. The superimposed layer of the Internet of Digital Twins—***IoDT***—provides a green field for the all-encompassing interaction and collaboration of IDTs within the DTRM. Particularly, the guiding principles of Industry 5.0 (human-centricity, sustainability, and resilience) could be embedded here as a basis for negotiation between the IDTs and through the marketplace concept itself. Furthermore, the implementation has proven that MAS paradigms are inevitable for the generation of IDTs (*H2*). Initially, the specific characteristics of the IDTs (active, online, goal-seeking, and anticipatory) were determined and compared with the main requirements of MAS. It is obvious that the modalities of both worlds can be perfectly synthesized into a comprehensive system. For this, both research domains must move closer together and develop universal approaches for the realization of IDTs.

Concisely, this paper contributes: **(1)** a reference model to precisely distinguish DT application areas and their production proximity; **(2)** a comparison of the characteristics on IDTs and MAS; **(3)** to the bilateral emergence of Industry 4.0 and 5.0 premises; and **(4)** a universal architectural approach to establish an IDT system for marketplace-oriented collaboration.

Nonetheless, the implementation presented here has some challenges, especially for future proceedings. To fully exploit the capabilities of the architecture, many machines need to be integrated into the marketplace, which could only be achieved by scaling up to an industrial scale. This entails a high level of integration and connection effort, as well as the need to find suitable project partners. Furthermore, it is planned to build the system via AAS and to use a more common MAS approach in the form of JADE/Jadex, in addition to the marketplace architecture. The issues of security, confidentiality, and functional safety should also be investigated in a similar way. Detailed design patterns for a precise implementation of the explained approaches will be covered in future publications. Conclusively, it should be remarked that both MAS and systems with DTs only are not a panacea for the realization of Industry 4.0 and 5.0 premises. As a small part, however, they make an important methodological and technological step towards its upcoming future feasibility.

Author Contributions: Conceptualization, J.L., A.L., and T.H.; methodology, J.L., A.L., A.R., and T.H.; software, J.L., A.L., and T.H.; validation, J.L., A.L., and T.H.; investigation, J.L., A.L., A.R., and T.H.; writing—original draft preparation, J.L., A.L., A.R., and T.H.; supervision, L.O., H.B., and J.R.; project administration, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: Parts of the work presented in this paper are supported by a grant from the German Ministry of Education and Research (BMBF), grant number 02J21C142 (FLoW-mIT).

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.

Abbreviations

The following abbreviations are used in this manuscript:

AAS	Asset Administration Shell
API	Application Programming Interface
BDI	Belief, Desire, and Intention
CPS	Cyber-Physical System
CPPS	Cyber-Physical Production Systems
DT	Digital Twin
DTRM	Digital Twin Reference Model
EU	European Union
HTTP	Hypertext Transfer Protocol
IDT	Intelligent Digital Twin
IIC	Industrial Internet Consortium
IIRA	Industrial Internet Reference Architecture
IoDT	Internet of Digital Twins
IoT	Internet of Things
JSON	JavaScript Object Notation
MAS	Multi-Agent System
MQTT	Message Queuing Telemetry Transport
OPC UA	Open Platform Communications Unified Architecture
PLC	Programmable Logic Controller
PLM	Product Lifecycle Management
PT	Physical Twin
RAMI 4.0	Reference Architectural Model Industry 4.0

References

1. Abbass , K.; Zeeshan Qasim, M.; Song, H.; Murshed, M.; Mahmood, H.; Younis, I. A review of the global climate change impacts, adaptation, and sustainable mitigation measures. In *Environmental Science and Pollution Research*; Springer Science and Business Media LLC: Berlin/Heidelberg, Germany, 2022; Volume 29, Number 28, pp. 42539–42559. [CrossRef]
2. Wu, X.; Zhang, C.; Du, W. An Analysis on the Crisis of "Chips shortage" in Automobile Industry —Based on the Double Influence of COVID-19 and Trade Friction. *J. Phys. Conf. Ser.* **2021**, *1971*, 012100. [CrossRef]
3. Jagtap, S.; Trollman, H.; Trollman, F.; Garcia-Garcia, G.; Parra-López, C.; Duong, L.; Martindale, W.; Munekata, P.E.S.; Lorenzo, J.M.; Hdaifeh, A.; et al. The Russia-Ukraine Conflict: Its Implications for the Global Food Supply Chains. *Foods* **2022**, *11*, 2098. [CrossRef]
4. Breque, M.; De Nul, L.; Petridis, A. *Industry 5.0: Towards a Sustainable, Human Centric and Resilient European Industry*; Publications Office of the European Union: Brussels, Belgium, 2021. [CrossRef]
5. Longo, F.; Padovano, A.; Umbrello, S. Value-Oriented and Ethical Technology Engineering in Industry 5.0: A Human-Centric Perspective for the Design of the Factory of the Future. *Appl. Sci.* **2020**, *10*, 4182. [CrossRef]
6. Agrawal, A.; Thiel, R.; Jain, P.; Singh, V.; Fischer, M. Digital Twin: Where do humans fit in? *Autom. Constr.* **2023**, *148*, 104749. [CrossRef]
7. Nahavandi, S. Industry 5.0—A Human-Centric Solution. *Sustainability* **2019**, *11*, 4371. [CrossRef]
8. Xu, X.; Lu, Y.; Vogel-Heuser, B.; Wang, L. Industry 4.0 and Industry 5.0—Inception, conception and perception. *J. Manuf. Syst.* **2021**, *61*, 530–535. [CrossRef]
9. Müller, J. *Enabling Technologies for Industry 5.0: Results of a Workshop with Europe's Technology Leaders*; Publications Office of the European Union: Brussels, Belgium, 2020. [CrossRef]
10. Miehe, R.; Waltersmann, L.; Sauer, A.; Bauernhansl, T. Sustainable production and the role of digital twins–Basic reflections and perspectives. *J. Adv. Manuf. Process.* **2021**, *3*, e10078. [CrossRef]
11. Lee, J.; Bagheri, B.; Kao, H.A. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. *Manuf. Lett.* **2015**, *3*, 18–23. [CrossRef]
12. Grieves, M. Intelligent digital twins and the development and management of complex systems [version 1; peer review: 4 approved]. *Digit. Twin* **2022**, *2*, 8. [CrossRef]
13. Sakurada, L.; Leitao, P.; De la Prieta, F. Towards the Digitization using Asset Administration Shells. In Proceedings of the IECON 2021—47th Annual Conference of the IEEE Industrial Electronics Society, Toronto, ON, Canada, 13–16 October 2021; pp. 1–6. [CrossRef]
14. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]
15. Baumgärtel, H. The role of Semantics and Semantic Web Technologies for Digital Twins in Industry 4.0 Systems. In Proceedings of the Workshop Management for Industry 4.0, IEEE/IFIP Network Operations and Management Symposium (NOMS), Budapest, Hungary, 25–29 April 2022. [CrossRef]
16. Sakurada, L.; Leitao, P.; la Prieta, F.D. Agent-Based Asset Administration Shell Approach for Digitizing Industrial Assets. *IFAC-PapersOnLine* **2022**, *55*, 193–198. [CrossRef]
17. Shafto, M.; Conroy, M.; Doyle, R.; Glaessgen, E.; Kemp, C.; LeMoigne, J.; Wang, L. *Modeling, Simulation, Information Technology and Processing Roadmap*; National Aeronautics and Space Administration: Washington, DC, USA, 2010.
18. Grieves, M.; Vickers, J. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In *Transdisciplinary Perspectives on Complex Systems*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 85–113. [CrossRef]
19. Piascik, B.; Vickers, J.; Lowry, D.; Scotti, S.; Stewart, J.; Calomino, A. *Technology Area 12: Materials, Structures, Mechanical Systems, and Manufacturing Road Map*; NASA Headquarters: Washington, DC, USA, 2010.
20. Lehmann., J.; Lober., A.; Rache., A.; Baumgärtel., H.; Reichwald., J. Collaboration of Semantically Enriched Digital Twins based on a Marketplace Approach. In Proceedings of the 19th International Conference on Wireless Networks and Mobile Systems—WINSYS, INSTICC, Lisbon, Portugal, 11–13 July 2022; SciTePress: Setúbal, Portugal; pp. 35–45. [CrossRef]
21. Wang, K.; Wang, Y.; Li, Y.; Fan, X.; Xiao, S.; Hu, L. A review of the technology standards for enabling digital twin. *Digit. Twin* **2022**, *2*, 4. [CrossRef]
22. 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. *Proc. Des. Soc. DESIGN Conf.* **2020**, *1*, 757–766. [CrossRef]
23. Singh, M.; Fuenmayor, E.; Hinchy, E.; Qiao, Y.; Murray, N.; Devine, D. Digital Twin: Origin to Future. *Appl. Syst. Innov.* **2021**, *4*, 36. [CrossRef]
24. Sjarov, M.; Lechler, T.; Fuchs, J.; Brossog, M.; Selmaier, A.; Faltus, F.; Donhauser, T.; Franke, J. The Digital Twin Concept in Industry—A Review and Systematization. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; Volume 1, pp. 1789–1796. [CrossRef]
25. 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]
26. Lehmann, J.; Schorz, S.; Rache, A.; Häußermann, T.; Rädle, M.; Reichwald, J. Establishing Reliable Research Data Management by Integrating Measurement Devices Utilizing Intelligent Digital Twins. *Sensors* **2023**, *23*, 468. [CrossRef] [PubMed]

27. Massonet, A.; Kiesel, R.; Schmitt, R.H. Der Digitale Zwilling über den Produktlebenszyklus. *Z. Für Wirtsch. Fabr.* **2020**, *115*, 97–100. [CrossRef]
28. Jazdi, N.; Talkhestani, B.A.; Maschler, B.; Weyrich, M. Realization of AI-enhanced industrial automation systems using intelligent Digital Twins. *Procedia CIRP* **2021**, *97*, 396–400. [CrossRef]
29. van der Valk, H.; Haße, H.; Möller, F.; Otto, B. Archetypes of Digital Twins. *Bus. Inf. Syst. Eng.* **2021**, *64*, 375–391. [CrossRef]
30. Rathore, M.M.; Shah, S.A.; Shukla, D.; Bentafat, E.; Bakiras, S. The Role of AI, Machine Learning, and Big Data in Digital Twinning: A Systematic Literature Review, Challenges, and Opportunities. *IEEE Access* **2021**, *9*, 32030–32052. [CrossRef]
31. Relekar, A.; Smolira, P.; Petrova, E.; Svidt, K. Enabling Digital Twin with Advanced Visualization and Contextualization of Sensor Data with BIM and Web Technologies. In Proceedings of the Conference CIB W, Luxembourg, 11–15 October 2021; Volume 78, pp. 11–15.
32. Haddadi, H.; Mortier, R.; McAuley, D.; Crowcroft, J. *Human-Data Interaction*; Technical Report UCAM-CL-TR-837; University of Cambridge, Computer Laboratory: Cambridge, UK, 2013. [CrossRef]
33. Stead, M.; Coulton, P.; Pilling, F.; Gradinar, A.; Pilling, M.; Forrester, I. More-than-Human-Data Interaction. In Proceedings of the 25th International Academic Mindtrek Conference, Tampere, Finland, 16–18 November 2022. [CrossRef]
34. 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]
35. Vogel-Heuser, B.; Ocker, F.; Weiß, I.; Mieth, R.; Mann, F. Potential for combining semantics and data analysis in the context of digital twins. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2021**, *379*, 20200368. [CrossRef] [PubMed]
36. Sahlab, N.; Kamm, S.; Muller, T.; Jazdi, N.; Weyrich, M. Knowledge Graphs as Enhancers of Intelligent Digital Twins. In Proceedings of the 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS), Victoria, BC, Canada, 10–12 May 2021; pp. 19–24. [CrossRef]
37. Zhang, C.; Zhou, G.; He, J.; Li, Z.; Cheng, W. A data- and knowledge-driven framework for digital twin manufacturing cell. *Procedia CIRP* **2019**, *83*, 345–350. [CrossRef]
38. P. I. 4.0. *Details of the Asset Administration Shell Part 1*; Plattform Industrie 4.0; Federal Ministry for Economic Affairs and Energy (BMWi): Berlin, Germany, 2020.
39. Ocker, F.; Urban, C.; Vogel-Heuser, B.; Diedrich, C. Leveraging the Asset Administration Shell for Agent-Based Production Systems. *IFAC-PapersOnLine* **2021**, *54*, 837–844. [CrossRef]
40. Wooldridge, M.J. *An Introduction to MultiAgent Systems, Second Edition*; Wiley: Hoboken, NJ, USA, 2009.
41. Fay, A.; Gehlhoff, F.; Seitz, M.; Vogel-Heuser, B.; Baumgaertel, H.; Diedrich, C.; Lüder, A.; Schöler, T.; Sutschet, G.; Verbeet, R. *Agents for the Realisation of Industrie 4.0—VDI Status Report*; VDI Verein Deutscher Ingenieure: Düsseldorf, Germany, 2019. [CrossRef]
42. Pons, M.E.P.; Domínguez, J.P.; Rojas, C.A.; Barroso, R.J.D.; de Miguel Jiménez, I.; de Queiroz, J.M.; Leitao, P. A brief review on multi-agent system approaches and methodologies. In Proceedings of the IV Workshop on Disruptive Information and Communication Technologies for Innovation and Digital Transformation, Online, 18 June 2021; pp. 35–47.
43. Wooldridge, M.; Jennings, N.; Kinny, D. The Gaia Methodology For Agent-Oriented Analysis And Design. *Auton. Agents Multi-Agent Syst.* **2000**, *3*, 285–312. [CrossRef]
44. Zambonelli, F.; Jennings, N.R.; Wooldridge, M. Developing Multiagent Systems: The Gaia Methodology. *ACM Trans. Softw. Eng. Methodol.* **2003**, *12*, 317–370. [CrossRef]
45. Pavón, J.; Gomez-Sanz, J.J.; Fuentes, R. *The Ingenias Methodology and Tools*; IGI Global: Hershey, PA, USA, 2005; pp. 236–276. [CrossRef]
46. Deloach, S.; Wood, M.; Sparkman, C. Multiagent Systems Engineering. *Int. J. Softw. Eng. Knowl. Eng.* **2001**, *11*, 231–258. [CrossRef]
47. Bresciani, P.; Perini, A.; Giorgini, P.; Giunchiglia, F.; Mylopoulos, J. Tropos: An Agent-Oriented Software Development Methodology. *Auton. Agents Multi-Agent Syst.* **2004**, *8*, 203–236. [CrossRef]
48. Giunchiglia, F.; Mylopoulos, J.; Perini, A. The Tropos Software Development Methodology: Processes, Models and Diagrams. In Proceedings of the 3rd International Conference on Agent-Oriented Software Engineering III, Bologna, Italy, 15–19 July 2002; pp. 162–173.
49. Padgham, L.; Winikoff, M. Prometheus: A Methodology for Developing Intelligent Agents. In Proceedings of the Agent-Oriented Software Engineering III, Bologna, Italy, 15–19 July 2003; pp. 174–185.
50. Chella, A.; Cossentino, M.; Sabatucci, L.; Seidita, V. Agile passi: An agile process for designing agents. *Int. J. Comput. Syst. Sci. Eng.* **2006**, *21*, 133–144.
51. Graham, J.; Decker, K.; Mersic, M. DECAF - A flexible multi agent system architecture. *Auton. Agents Multi-Agent Syst.* **2003**, *7*, 7–27. [CrossRef]
52. Sycara, K.; Pannu, A.; Willamson, M.; Zeng, D.D.; Decker, K. Distributed Intelligent Agents. *IEEE Expert* **1997**, *11*, 36–46. [CrossRef]
53. Sycara, K.P.; Paolucci, M.; Velsen, M.V.; Giampapa, J.A. The RETSINA MAS Infrastructure. *Auton. Agents Multi-Agent Syst.* **2003**, *7*, 29–48. [CrossRef]
54. Leitao, P.; Karnouskos, S.; Ribeiro, L.; Lee, J.; Strasser, T.; Colombo, A.W. Smart Agents in Industrial Cyber–Physical Systems. *Proc. IEEE* **2016**, *104*, 1086–1101. [CrossRef]

55. Karnouskos, S.; Leitao, P.; Ribeiro, L.; Colombo, A.W. Industrial Agents as a Key Enabler for Realizing Industrial Cyber-Physical Systems: Multiagent Systems Entering Industry 4.0. *IEEE Ind. Electron. Mag.* **2020**, *14*, 18–32. [CrossRef]
56. Cruz Salazar, L.A.; Ryashentseva, D.; Lüder, A.; Vogel-Heuser, B. Cyber-physical production systems architecture based on multi-agent's design pattern—comparison of selected approaches mapping four agent patterns. *Int. J. Adv. Manuf. Technol.* **2019**, *105*, 4005–4034. [CrossRef]
57. Verbeet, R.; Baumgärtel, H. Implementierung von autonomen I4.0-Systemen mit BDI-Agenten. In *Handbuch Industrie 4.0: Produktion, Automatisierung und Logistik*; ten Hompel, M., Vogel-Heuser, B., Bauernhansl, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2020; pp. 1–36. [CrossRef]
58. Melo, V.; de la Prieta, F.; Leitao, P. Alignment of Digital Twin Systems with the RAMI 4.0 Model Using Multi-agent Systems. In Proceedings of the Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future, Bucharest, Romania, 22–23 September 2022; pp. 23–35.
59. Sakurada, L.; Leitão, P. Multi-Agent Systems to Implement Industry 4.0 Components. In Proceedings of the 2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS), Tampere, Finland, 10–12 June 2020; Volume 1, pp. 21–26. [CrossRef]
60. López, A.; Casquero, O.; Estévez, E.; Leitão, P.; Marcos, M. Towards the generic integration of agent-based AASs and Physical Assets: A four-layered architecture approach. In Proceedings of the 2021 IEEE 19th International Conference on Industrial Informatics (INDIN), Palma de Mallorca, Spain, 21–23 July 2021; pp. 1–6. [CrossRef]
61. Vogel-Heuser, B.; Ocker, F.; Scheuer, T. An approach for leveraging Digital Twins in agent-based production systems. *Automatisierungstechnik* **2021**, *69*, 1026–1039. [CrossRef]
62. Ambra, T.; Macharis, C. Agent-Based Digital Twins (ABM-DT) in Synchromodal Transport and Logistics: The Fusion of Virtual and Pysical Spaces. In Proceedings of the 2020 Winter Simulation Conference (WSC), Orlando, FL, USA, 14–18 December 2020; pp. 159–169.
63. Ambra, T.; Caris, A.; Macharis, C. Should I Stay or Should I Go? Assessing Intermodal and Synchromodal Resilience from a Decentralized Perspective. *Sustainability* **2019**, *11*, 1765. [CrossRef]
64. Ambra, T.; Caris, A.; Macharis, C. Do You See What I See? A Simulation Analysis of Order Bundling within a Transparent User Network in Geographic Space. *J. Bus. Logist.* **2021**, *42*, 167–190. [CrossRef]
65. Dittler, D.; Lierhammer, P.; Braun, D.; Müller, T.; Jazdi, N.; Weyrich, M. An Agent-based Realisation for a continuous Model Adaption Approach in intelligent Digital Twins. *arXiv* **2022**, arXiv:2212.03681.
66. Alcaraz, C.; Lopez, J. Digital Twin: A Comprehensive Survey of Security Threats. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 1475–1503. [CrossRef]
67. *ISO/DIS 23247-1:2020(E)*; Automation Systems and Integration—Digital Twin Framework for Manufacturing—Overview and General Principles. International Organization for Standardization: Geneva, Switzerland, 2020.
68. *ISO/DIS 23247-2:2021(E)*; Automation Systems and Integration—Digital Twin Framework for Manufacturing—Reference Architecture. International Organization for Standardization: Geneva, Switzerland, 2021.
69. 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]
70. Aheleroff, S.; Xu, X.; Zhong, R.Y.; Lu, Y. Digital Twin as a Service (DTaaS) in Industry 4.0: An Architecture Reference Model. *Adv. Eng. Inform.* **2021**, *47*, 101225. [CrossRef]
71. Bevilacqua, M.; Bottani, E.; Ciarapica, F.E.; Costantino, F.; Donato, L.D.; Ferraro, A.; Mazzuto, G.; Monteriù, A.; Nardini, G.; Ortenzi, M.; et al. Digital Twin Reference Model Development to Prevent Operators' Risk in Process Plants. *Sustainability* **2020**, *12*, 1088. [CrossRef]
72. 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]
73. *DIN SPEC 91345*; Reference Architecture Model Industrie 4.0 (RAMI4.0). DIN German Institute for Standardization: Berlin, Germany, 2016.
74. Plattform Industrie 4.0. RAMI4.0—A Reference Framework for Digitalisation. 2018. Available online: https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/rami40-an-introduction.html (accessed on 2 January 2023).
75. Lin, S.W.; Miller, B.; Durand, J.; Bleakley, G.; Chigani, A.; Martin, R.M.; Murphy, B.; Crawford, M. The Industrial Internet of Things Volume G1: Reference Architecture; Version 1.9. 2019. Available online: https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf (accessed on 30 December 2022).
76. Lin, S.W.; Murphy, B.; Clauer, E.; Loewen, U.; Neubert, R.; Bachmann, G.; Pai, M.; Hankel, M. Architecture Alignment and Interoperability: An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper; IIC:WHT:IN3:V1.0:PB:20171205. 2018. Available online: https://www.iiconsortium.org/pdf/JTG2_Whitepaper_final_20171205.pdf (accessed on 30 December 2022).
77. Belyaev, A.; Diedrich, C. Specification, "Demonstrator I4.0-Language" v3.0, Technical Report. IFAT-LIA 07/2019. 2019. Available online: https://www.researchgate.net/publication/334429449_Specification_Demonstrator_I40-Language_v30 (accessed on 10 January 2023).
78. Lober, A.; Lehmann, J.; Häußermann, T.; Reichwald, J.; Baumgärtel, H. Improving the Engineering Process of Control Systems Based on Digital Twin Specifications. In Proceedings of the 2022 4th International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM), Reduit, Mauritius, 22–24 November 2022; pp. 1–6. [CrossRef]

79. Eclipse Foundation. Eclipse Ditto™ Documentation.2021. Available online: https://www.eclipse.org/ditto/intro-overview.html (accessed on 9 November 2021).
80. Luyen, L.N.; Tireau, A.; Venkatesan, A.; Neveu, P.; Larmande, P. Development of a knowledge system for Big Data. In Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics, Nimes, France, 13–15 June 2016. [CrossRef]
81. Vrgoc, D.; Rojas, C.; Angles, R.; Arenas, M.; Arroyuelo, D.; Aranda, C.B.; Hogan, A.; Navarro, G.; Riveros, C.; Romero, J. MillenniumDB: A Persistent, Open-Source, Graph Database. *arXiv* **2021**, arXiv:2111.01540.