Properties and Characteristics of Digital Twins: Review of Industrial Definitions

Lukas Schweiger[1,4] · Linard Barth[2,3]

Abstract

As digital twin configurations depend on their use case, there is a need for research on how companies can select the capabilities and appropriate level of sophistication to deploy digital twins in practice successfully. This study investigated the properties and characteristics of digital twins described in academic literature. It summarized them in a taxonomy, which was subsequently used to code and examine 90 definitions of companies. For the analysis, both supervised and unsupervised methods were applied. The results show that researchers focus more on technological requirements when defining digital twins, while companies use more value-based properties that are not included or not precisely delineated in academic reviews. Therefore, an application-oriented definition is proposed to bridge this gap and complement the taxonomy. This study thus contributes to the discussion and forming of an application-oriented and shared understanding of the digital twin concept in research and practice.

Keywords Digital twin · Definition · Taxonomy · Industry · Systematic review

Introduction

The development and progress in information and communication technologies will transform more traditional products into smart connected products, enabling novel smart services and ultimately changing whole industries [1–4]. Here, the digital twin (DT) concept is regarded as a critical technology for creating value with smart services [5] and for realizing smart manufacturing and industrial digital transformation [6]. With DTs, companies seek to create value in both the internal dimension (internal processes regarding product lifecycle) as well as the external dimension (during the usage phase in the market) [7, 8].

Origin The origin of the DT concept is attributed to Michael Grieves and John Vickers of NASA, with Grieves [9] presenting the concept in a lecture on product life cycle management (PLM) in 2003. In 2012, the concept of DT was revisited by NASA, which defined the DT as a "multiphysics, multiscale, probabilistic, ultra fidelity simulation that reflects, promptly, the state of a corresponding twin based on the historical data, real-time sensor data, and physical model" [10]. Since then, the understanding of the term DT has continuously changed with its development [11–13].

Status quo The definition of DT has evolved in the last decade, alongside its growing popularity and adoption into different industries and use cases, which seldom meet the stringent demands of a real one-to-one copy described in NASA's definition [14, 15]. An overview of selected reviews in the research field of DT considered for this study is shown in Table 1. This demonstrates that DT is both a recent and an active research topic [16]. Many publications adopted

L. Schweiger, L. Barth contributed equally to this work.

✉ Lukas Schweiger
scge@zhaw.ch

Linard Barth
bath@zhaw.ch

1   School of Engineering, Zurich University of Applied Sciences, Technikumstrasse 81, Winterthur 8401, Zurich, Switzerland

2   Faculty of Business and Economics, Mendel University, Zemědělská 1665/1, Brno 613 00, Czech Republic

3   School of Management and Law, Zurich University of Applied Sciences, Theaterstrasse 17, Winterthur 8401, Zurich, Switzerland

4   Department of Industrial Engineering, University of Florence, Viale Morgagni 40, Firenze 50139, Tuscany, Italy

Table 1 Overview of digital twin reviews

Publication, year	Type of sources	Number of sources	Search method	Definition comparison method	Review focus	Main outcomes
[24], 2016	sl	38	sys	nr	Manufacturing	Summary of concepts, comparison of applications
[25], 2017	sl	26	sys	nr	Industry 4.0	Summary of definitions, own definition, comparison of applications and technologies
[18], 2018	sl	43	sys	nr	Manufacturing, level of integration	Framework
[20], 2019	sl	75	sys	sys	Definitions, applications	Comparison of definitions, summary of characteristics and applications
[26], 2019	sl	644	n/a	nr	Application	Comparison of applications
[27], 2019	sl	52	sys	nr	Application in manufacturing	Comparison of applications
[28], 2019	sl	152	sys	nr	Application	Own definition, classification scheme
[29], 2019	sl	19	n/a	nr	Definition, Industry 4.0	Own definition, framework
[30], 2019	sl (50), p (8)	58	sys	nr	Components, development, applications	Summary of technologies and applications
[7], 2020	sl	41	sys	nr	Dimensions, specifications	Framework
[31], 2020	sl	167	sys	nr	Application in maintenance	Comparison of case studies
[32], 2020	sl	26	n/a	nr	Smart cities, manufacturing and healthcare	Comparison of technologies and challenges
[19], 2020	sl	92	sys	sys	Characteristics, knowledge gaps	Summary of characteristics, framework
[33], 2020	sl	n/a	sys	nc	Manufacturing systems and processes	Summary of applications, standards and communication protocols, framework
[34], 2020	sl	n/a	n/a	nr	Internet of things	Summary of technologies and characterizing properties, own consolidated definition, application scenarios, architecture framework
[17], 2020	sl (38), is (2)	40	sys	sys	Definitions	Summary of definitions and models, comparison of purposes of applications
[21], 2020	sl	233	sys	sys	Definitions	Taxonomy, comparison of definitions
[35], 2021	sl	496	sys	nc	Author contribution, topics, trends	Summary of research trends, comparison of research clusters
[22], 2021	sl	134	sys	sys	Civil engineering	Comparison of DT, BIM and CPS definitions, summary of research clusters
[16], 2021	sl	24	sys	sys	Manufacturing, logistics	Own definition, comparison of characteristics and properties
[36], 2021	sl	115	sys	nr	Industrial application	Comparison of definitions
[37], 2021	sl	240	sys	nr	Industrial application	Comparison of applications and technologies
[38], 2021	sl	41	sys	nc	Industrial application	summary of applications
[39], 2021	sl	31	sys	nr	Construction industry	Comparison of applications and technologies
[23], 2021	sl	150	sys	sys	Industry 4.0, manufacturing	Own definition
[40], 2021	sl	n/a	n/a	nc	Definitions	Comparison of definitions, own definition

Table 1 (continued)

Publication, year	Type of sources	Number of sources	Search method	Definition comparison method	Review focus	Main outcomes
[41], 2021	sl	46	n/a	nr	Definitions	Own definition, framework
[42], 2022	sl	234	sys	nc	Operations, supply chain management	Comparison of characteristics and properties
[43], 2022	sl	370	sys	nr	Industrial applications, key implementation technologies	Summary of technologies
[44], 2022	sl	182	sys	nc	Construction industry	Comparison of technologies and applications
[45], 2022	sl	94	n/a	nr	Augmented reality	Comparison of technologies and applications
[46], 2022	sl	n/a	n/a	nr	Application, Components and properties	Comparison of components
[47], 2022	sl	n/a	n/a	nc	Industrial application	Comparison of applications
[48], 2022	sl	91	sys	nr	Product life cycle	Comparison of applications and technologies, framework
[49], 2022	sl	284	sys	nc	Modeling, tools and technology	Framework
[13], 2022	sl	54	sys	sys	Definitions	Comparison of definitions
[6], 2023	sl	117	sys	nr	Application	Summary of applications, comparison of DT and CPS, framework
[50], 2023	sl	148	sys	nc	Robotics	Own definition, framework
[51], 2023	sl	37	sys	nc	Energy consumption management	Summary of applications
This paper	is	90	sys	sys	Definitions, properties, characteristics	Summary of properties and characteristics, comparison of company definitions, own definition

Scientific literature = sl, patents = p, industrial sources = is, systematic = sys, narrative = nr, not available or unclear = n/a, no comparison = nc

the DT concept with some deviations from the original definition, albeit avoiding defining the DT concept explicitly themselves [17]. Instead, they implicitly assume a particular set of abilities and properties, thus hindering the formation of an accurate definition. Owing to this proliferation of similar but different definitions, an increasing number of attempts have been made since 2020 to clarify what constitutes a DT (e.g., [7, 18, 19]). In the following, we briefly summarize the reviews that took a systematic approach to both the selection of sources and their analysis (cf. Table 1).

Barricelli et al. [20] present findings related to examining state-of-the-art DT definitions, assessing essential characteristics a DT should have, and investigating domains where DT applications are currently being created. In their research, Jones et al. [19] elucidated the conceptual status, key terminology, and related processes to define characteristics, a framework of DT and its processes of operation. Sjarov et al. [17] systematically examined DT definitions and assorted related concepts such as product avatar and digital shadow. In addition, depictions of DT models from the prominent literature and derived DT purposes are presented. Van der Valk et al. [21] conducted an extensive literature review on the scope and meaning of the term DT to create a taxonomy and compare some of the most widespread definitions. Jiang et al. [22] compare the DT concept to building information modeling (BIM) and the concept of cyber-physical systems (CPS). Based on this comparison, they propose their own definition of DT. Furthermore, they cluster the research in the field of civil engineering. Kuehner et al. [16] compare existing reviews of DT in a meta-review to detect prevalent as well as contrasting views to clarify commonalities in terminology, conceivable benefits, and remaining research issues. The review of Semeraro et al. [23] aims to answer the question of what a DT is, when it should be developed, why it should be used, how to design and implement it, and what the main challenges of implementation are. These questions are answered by analyzing the concept, the life cycle, and the primary functions of DTs at different stages. Tomczyk et al. [13] analyzed the journey of the DT definition and described a paradigm shift from the

classic three-dimensional definition—physical and virtual space with a bidirectional connection—to an expanded five-dimensional definition—with data and services as additional dimensions.

These researches examine how DT has been defined to date but do not offer a forward-looking, application-oriented definition. Either they only compare and summarize existing concepts and do not provide their own definitions [17, 19–21], or if they propose new definitions, they are too narrow regarding implementation in practice, e.g., by only including physical counterparts or demanding real-time connectivity [16, 22, 23].

In summary, a large number of reviews and theoretical studies exist. However, there is still a gap when it comes to the needs and requirements of practitioners, especially as their approaches and definitions are overlooked in current research. This results in a research gap that hinders the implementation of DT in practice.

Research gap In practice, industrial systems and usage scenarios are complex and diverse, therefore, their requirements for DTs will be similarly diverse [12]. As a result, DTs depend heavily on their individual use case, leading to many different configurations in practice. For this reason, it is difficult to identify the characteristics and nature of the DT needed to realize the benefits each industrial context requires [19]. There are already efforts to close this gap by some authors. For example, van der Valk et al. [52] propose a set of archetypes of DTs for individual use cases, and Haße et al. [53] developed design principles for shared DTs in distributed systems. Other researchers focused on the architecture of DTs and how the various DT components interact in specific domains (e.g., [54, 55]). However, Agrawal et al. [56, 57] found that there is still a lack of consensus on the DT concept in practice and insufficient research on how practitioners can select the capabilities and appropriate level of sophistication to deploy a DT in practice. This research gap hinders the application of DTs in practice, because the potential costs and challenges of the infrastructure and workflow changes needed to implement DTs effectively depend heavily on the sophistication level of the needed DT. Together with a lack of tangible understanding of the scale and nature of value creation with DTs, this forms the substantial knowledge gap in practice [19]. Although the focus of research on DTs is turning more to the services they enable and the value they create [7, 8], there is still a need for an application-oriented definition and understanding of DTs. This definition should be generic enough to be used in various application areas and simultaneously extend the differing current partial views of the DT concept [58]. Moreover, such a definition should be aligned with the existing understanding of DTs in practice. Even though some research was performed on the differences in understanding the DT concept in research and practice [5] there is no systematic analysis of the similarities and differences between the DT definition of companies. Only Sjarov et al. [17] included two industrial sources in their review of definitions, and Tao et al. [30] considered eight patents (cf. Table 1). Apart from this, extensive and systematic reviews solely focus on scientific literature, and only a few use systematic methods for comparing definitions. Therefore, this paper aims to contribute to the discussion and forming of an application-oriented understanding and definition of the DT concept by answering the following main research question:

> *Which properties and characteristics are used by companies to define Digital Twins?*

Relevance for practice DT hit the peak of inflated expectations in the 2018 emerging technology hype cycle by Gartner [59] and was projected to need another 5–10 years to reach the plateau of productivity. 78 percent of companies regard DTs as a competitive advantage [60], which is why established software vendors and large industrial conglomerates devised DT concepts and started initiatives to develop them for their product and service business. At the same time, there is still a long way to go, as shown by a study in Switzerland, where only 22 percent of respondent businesses said their products were smart and connected, and just 13 percent said they were capable of establishing a DT of their products [61]. A more recent study by Barth et al. [8] found that almost a quarter (23 out of 103) of the Swiss companies surveyed currently apply DTs to create value. Another quarter (25 out of 103) plans to apply DTs to create value along the product life cycle in the future. These companies recognize the potential of DTs to create value in the beginning of life (BoL) phase and somewhat weaker in the middle of life (MoL) phase. On the other hand, the use in the end of life (EoL) phase has a subordinate significance. The companies in this survey seek to create value with DTs by offering qualitatively better products and services in a shorter time or with higher availability. Cost savings seem secondary and also overestimated by companies before applying DTs.

Structure of paper The paper is structured as follows: first, approaches for the systematic representation of properties and characteristics of DTs from the existing scientific literature are explained. Then, our own taxonomy to systematize the DT defining properties and characteristics is presented, which is used in this research to analyze the definitions of companies we collected from their web pages. In the subsequent chapter, the detailed research questions, the research procedure, the procedure for data collection, and the methods for supervised and unsupervised analyses of the definitions are explained. After this, we present the results of the supervised and the unsupervised analyses, followed by

a discussion of the research questions based on them. The discussion chapter concludes by proposing an application-oriented definition of the DT concept to promote its use in practice. In the final chapter, we summarize the main findings, highlight the implications, elaborate the contributions to research and practice, concede limitations, and suggest avenues for further research.

Properties and Characteristics of Digital Twins

Academic researchers have long debated a unified definition of DTs in academic literature without arriving at a generally accepted understanding [13], as the understanding of the term varies between applications [16]. As a consequence, the discussion has been placed on increasingly granular components of DTs in recent years (e.g., [34]). Thus, the current approach is to arrive at a generally accepted overall definition by discussing and defining DTs' individual properties and characteristics.

In this chapter, we first present existing approaches to systematize the defining properties and characteristics of DTs. Afterward, we present our proposal for a taxonomy that systematizes the DT-defining properties and characteristics that are particularly relevant for practice. This taxonomy was subsequently used for the systematic analysis of the properties and characteristics used by companies to define DTs in practice. The method section will explain the detailed research procedure (cf. Chapter 3).

Existing Approaches

This section provides a brief overview of existing approaches to systematizing DT-defining properties and characteristics in the order of their publication year.

As the result of the conceptualization of 87 application-based papers, Enders and Hoßbach [28] identified the following six dimensions for DTs: industrial sector, purpose, physical reference object, completeness, creation time, and connection. In their classification scheme, ten industries are distinguished in the industry dimension and between two and four characteristics in the other dimensions.

Stark and Damerau [29] propose a DT 8-dimension model as a structured approach for planning the scope and type of DTs. Each of the eight dimensions has three or four levels, some of which represent different levels of maturity, while others simply represent different realization spaces. The four dimensions, integration breadth, connection mode, update frequency, and product life cycle represent the DT environment and context. The other four dimensions, CPS intelligence, simulation capabilities, digital model richness, and human interaction, represent the DT behavior and capability richness.

After a systematic literature review and a thematic analysis of 92 DT publications Jones et al. [19] described 13 characteristics of DTs: physical entity/twin, virtual entity/twin, physical environment, virtual environment, state, realization, metrology, twinning, twinning Rate, physical-to-virtual connection/twinning, virtual-to-physical connection/twinning, physical processes, and virtual processes.

Van der Valk et al. [21] presented a taxonomy that distinguishes 8 dimensions with 2–3 characteristics derived from a literature review of 233 academic publications. Their taxonomy categorized the various definitions and concepts of DTs that have appeared in the literature over the years.

With their extensive publication, Minerva et al. [34] aim to consolidate a common DT definition by identifying a set of fundamental properties that can hold different contexts and situations and maintain generality, thus outlining the essential characteristics of a DT. They discuss the following 14 properties: identity, virtualization, representativeness and contextualization, reflection, replication, entanglement (including connectivity, promptness, and association), persistency, memorization, composability, accountability/manageability, augmentation, ownership, servitization, and predictability.

Kuehner et al. [16] conducted a meta-review of 24 reviews concerning DTs to detect prevalent and contrasting views on key issues. They found that even though only a third of reviews provide an explicit definition of a DT, most of them describe a selection of defining elements. Subsequently, they distinguish the following four definition elements: virtual representation, bidirectional connection, simulation, and connection across life cycle phases.

The review of Sharma et al. [46] focuses on the key DT features, current approaches in different domains, and successful DT implementations to infer the key DT components and properties. Their conceptualization includes three elementary components (physical asset, digital asset, and information flow), five imperative components (internet of things (IoT) devices, data, machine learning, security, and evaluation), and four properties (self-evolution, domain dependence, autonomy, synchronization). The characteristics of these elements and properties are exemplified by Sharma et al. [46] based on three use cases.

Despite these studies, no conclusive categorization of the possible characteristics of the individual elements and properties has yet been established. This may be due to the fact that many of the studies mainly focused on summarizing and systematizing previous studies [16, 19, 21, 28]. Other approaches are too theoretical, and the relevance of the proposed characteristics and properties with respect to their use in practice is uncertain [34]. Others are application-oriented, but they either focus on existing applications to present the

current state of DT and are thus not enough future-oriented [46] or they have weaknesses in terms of derivation from theory and validation against the practice [29].

Therefore, to promote the adaptation of DTs in practice, a forward-looking, application-oriented taxonomy is still needed that allows companies to assemble a DT concept with the appropriate level of sophistication for their use case in the sense of a morphological box. We propose such a taxonomy based on existing reviews and publications, our research and experience from projects, and definitions we encountered in the industry during this research. This taxonomy was subsequently used to code and compare the definitions from the industry.

New Taxonomy of Digital Twin Defining Properties and Characteristics

We propose a taxonomy for the systematic representation of DT-defining properties and characteristics, which includes ten properties with two to six possible characteristics each. The taxonomy is shown in Fig. 1 and contains, next to the properties and characteristics, an additional column indicating whether the characteristics are exclusive or not.

The following sections explain and substantiate each property and its characteristics.

Counterpart

Most authors agree that the DT *counterpart* should at least cover physical products and components [17]. Even the researchers who focus heavily on physical products mention intangible *counterparts*. For example, Stark and Damerau [29] mention that a DT has a physical twin throughout its life cycle, but they seem to consider directly related services as well. And although Jones et al. [19] do not explicitly include *non-physical* entities to be twinned by DTs, they acknowledge that more abstract entities, such as supply chains are also twinned. On a more generic level, this means systems, subsystems, and systems of systems, which also include the corresponding processes, hence *non-physical* entities. Minerva et al. [34] state that one of the entities connected by a DT is relevant in the real world and, thus, usually physical. However, they explicitly mention that software and intangible entities such as processes and activities can also be represented with DTs. Specific approaches also propose DT concepts for human workers [62] or business services [63]. Malakuti et al. [54] state that the DT definition has been enriched over time to be an evolving digital profile of the historical and current behavior and all properties of an asset—where an "asset can be anything of value for an organization" such as a physical device, subsystem, plant or software entity. This view is agreed by Boss et al. [12], who argue that a DT can represent anything in the real world of interest to an application as long as this real-world *counterpart* can be defined as an item with a recognizably distinct existence. Therefore, we use the term real-world *counterpart* rather than the physical *counterpart*, as in practice, any real-world entities with a recognizably distinct existence and relevance for creating value can be digitally represented by a DT. In conclusion, we propose to classify DT definitions according to whether DTs are used to represent (i) *only physical* objects, (ii) *non-physical* objects (in the sense of processes and services), or (iii) *any distinct entity*. These characteristics are mutually exclusive, as they are extensions of each other—for example if *any distinct entity* is mentioned, then both *physical* and *non-physical counterparts* are included.

Data Sources

According to Minerva et al. [34], an important feature of the DT is the ability to collect, store and represent all relevant present and past data of its real-world *counterpart*. The data are collected from various sources, such as onboard product systems, internal enterprise systems, and third-party sources. Consolidating this information in a DT lays the foundation for any data-based innovation of services and processes. Even in the case of *non-physical* entities, there are usually

Properties	Characteristics					Exclusivity
Counterpart	Only physical		Also non-physical		Any distinct entity	Mutual
Data sources	Sensors		Internal systems		External systems	Not
Data link	Bi-directional			Uni-directional		Mutual
Interface	Human-to-machine			Machine-to-machine		Not
Fidelity	One-to-one			Sufficient		Mutual
Synchronization	Real-time		Near-real-time		Periodic	Not
Capabilities	Simulation	Optimization	Prediction	Detection	Prevention	Automation	Not
Purpose	Performance		Availability		Quality	Not
Life cycle	Beginning of life		Middle of life		End of life	Not
Creation	Independent creation of DT			Types / instances distinguished		Not

Fig. 1 Taxonomy with properties and characteristics of digital twins, used for coding the definitions found in industry

smart connected products with *sensors* and embedded systems, allowing them to send relevant data from the real world to the DT, thus providing conclusions about the *non-physical* entities, such as processes. Also called IoT devices, these *data sources* enable the collection of sensor data from various subcomponents of physical assets and edge devices [46]. Utilizing the DT as a single source of truth for instance-related data minimizes redundant data and potentially conflicting information from heterogeneous, non-embedded systems [64]. These non-embedded systems can be divided into internal and external information systems connected to the DT. Firstly, DTs utilize, integrate, and recombine contents from several internal information systems such as authoring, product data management (PDM), enterprise resource planning (ERP), and customer relationship management (CRM) systems, and others such as computer-aided design (CAD) for specific objectives [24]. Secondly, a DT may utilize, integrate, and recombine content directly from other companies' interconnected systems and third-party data providers offering valuable data via application programming interfaces (API) or IoT platforms. This group of sources is referred to as *external systems*. We, therefore, propose to classify DT definitions according to whether they use data from (i) *sensors*, (ii) *internal systems*, and (iii) *external systems*. These characteristics are not exclusive, as they can be used in any combination.

Data Link

According to the definitions of Grieves [9], and Tao et al. [65], a *bi-directional* connection between the physical and virtual object is a mandatory part of a DT. Kuehner et al. [16] also noted that a *bi-directional* connection is one of only four prevailing definition elements in DT literature. Van der Valk et al. [52] additionally state that a DT must, by definition, have a *bi-directional data link*. They even go one step further and mention that in the context of DTs in networks, one should speak of multi-directional *data links* but do not pursue the concept any further.

However, if analyses are considered which examined DT applications or whose concepts are oriented towards application in practice, a more differentiated picture emerges. Enders and Hoßbach [28] identified three manifestations of connections in self-proclaimed DT applications (amount in parenthesis): no connection (23), one-directional connection (39), and *bi-directional* connection (25). Another influential argument regarding *data links* in the sense of the level of integration between physical and digital objects comes from Kritzinger et al. [18]. Depending on whether the connection between the objects is manual or automatic, they propose to distinguish three subcategories of DTs: digital model (both manual), digital shadow (automatic from physical to digital), and DT (both automatic). However, they found in their review that the term DT is often used synonymously with the other two, as in only 8 of the analyzed 43 publications, DTs are described that explicitly show automated connections in both directions.

Therefore, it seems that from a practice-oriented viewpoint, it is also appropriate to speak of a DT if not both connections are strictly digital and automated. Research consequently started to elaborate a more differentiated view which is required for a practice-oriented approach. For example, the model of Stark and Damerau [29] has a dimension that specifies *uni-directional* and *bi-directional* connectivity. In the dimension, they call connectivity mode, they additionally distinguish a third level of automatic, context-aware self-directed communication capability. Such a view is also supported by the review of Jones et al. [19], who have stated that the virtual-to-physical connection is not always included in the descriptions of DTs. They further mention that conceptually it is possible to generate a DT with just a one-way physical-to-virtual connection and that the role of human-in-the-loop is not frequently discussed in the literature. To illustrate this, they explain a brief example in which a human technician is sent out by the DT to perform a maintenance task decided on the basis of a predictive model. The elaborations of Minerva et al. [34] on the entanglement of DTs with the real world can also be interpreted in this direction. They state that there can be a unidirectional or bidirectional connection between the DT and the real-world *counterpart*. Furthermore, this connection can be direct or indirect, with the two communicating objects relying on a third party to send and receive information in the indirect case.

It can be summarized that a DT must have a connection to its real-world *counterpart* in any case. Depending on the view, it may be sufficient in reality for this connection to be digital and automated in only one direction (from the real-world *counterpart* to the DT) and for the processed data (in the form of information or actions) to be fed back not directly via the DT but manually via human actors. Therefore, we propose to classify DT definitions according to whether the DT has (i) *uni-directional* or (ii) *bi-directional* automated *data links*. These characteristics are mutually exclusive since a *bi-directional data link* extends a *uni-directional data link*.

Interfaces

The *interface* property defines which gateways can be used to access the data and information provided by the DT. Following on from the explanations in the previous section, we distinguish two types of *interfaces* as relevant for data output. In principle, there are two possibilities: the DT communicates with human users via human-to-machine (*H2M*) interfaces or directly with the real-world *counterpart* or other DTs via machine-to-machine (*M2M*) interfaces [52].

H2M interfaces are essential for DTs since many data and action-triggering decisions still require manual interpretation. Jones et al. [19] mention that humans might be in the loop to carry out actions decided by the DT or to train the DT before engaging with the real-world *counterpart*. Minerva et al. [34] state that humans should be able to interact with the DT via API to transform and change the real-world *counterpart*. Stark and Damerau [29] dedicated an entire dimension of their model to human interaction with DTs, distinguishing three different levels, including smart devices and augmented and virtual reality. In practice, there are many different ways to realize such a *H2M interface*, from simple ones such as light or sound signals to sophisticated dashboards and even augmented reality [66–68]. Ala-Laurinaho et al. [69] propose a single *interface* for accessing all the physical product data, allowing human users to search for the system that contains the desired information or implements the needed function. Brockhoff et al. [70] have used a highly adaptable generator framework to generate large parts of a DT cockpit [71] that aims to integrate human users.

DTs also require a series of *M2M interfaces* to communicate with non-human users. In addition to the mandatory *interface* to the real-world *counterpart*, DTs should have *interfaces* that enable communication with other DTs as users of their data and services [34]. Only in this way they can communicate with all the necessary entities in a CPS. This is the primary enabler for autonomously operating [52] and self-adaptive DTs [70]. The exact design of the *M2M interfaces* can be manifold [72, 73]. However, it is recommended to ensure the desired reliability and performance of the API, as they are critical to the overall success of the DT development model [74]. Therefore, we propose to classify DT definitions according to whether they mention (i) *H2M interfaces* or (ii) *M2M interfaces*. These characteristics are not exclusive, as both or only *H2M interfaces* may be defined since data input via *M2M interfaces* is mandatory for DT.

Fidelity

DTs may take many forms, but they all capture and utilize data representing the real world with a certain authenticity. To determine the authenticity of a DT, two properties have to be distinguished: The *fidelity* to reality and the intervals or the speed of *synchronization*. This section is focused on *fidelity*, while *synchronization* is explained as an independent property in the following section.

The definition by Glaessgen and Stargel [10] describe an "ultra fidelity simulation", while Grieves and Vickers [75] define the DT as accurate "from a micro-atomic level to the macro-geometrical level". This extremely high requirement for the *fidelity* of a DT is adopted accordingly by most academic authors. Jones et al. [19] confirmed this

in their comprehensive work, where they stated that apart from a small number of occasions where an appropriate, use-case-specific *fidelity* is called for, the *fidelity* of the virtual model is described as a highly accurate replication of the physical entity. This is also in line with the findings of van der Valk et al. [21], who demonstrated that the vast majority of authors and the three most widely used definitions describe an identical digital copy. This definition may be correct and productive for academics and as a target for the future development of DTs. However, such high demands are exaggerated for many practical projects, especially the first steps toward developing a highly authentic digital representation of reality. Because the representation of the real-world *counterpart* in all its facets and implications is often difficult and sometimes worthless [34]. In some cases, accuracy "from the micro-atomic level to the macro-geometrical level" is even virtually impossible. For example, it may not be possible to directly measure the process for chemical and biological reactions or extreme conditions. In other cases, instrumenting the physical objects may not be cost-effective or practical. As a result, organizations need to resort to proxies (e.g., relying on the instrumentation and *sensors* within a vehicle rather than putting *sensors* into tires) or things that are possible to detect (e.g., heat or light coming from chemical or biological reactions) [76]. Researching real-world cases that claim to use DTs reveals that most use cases require only a modest number of strategically placed *sensors* to detect critical inputs, outputs, and stages within the process to create a DT [76]. Jones et al. [19] also stated that the DTs of the most researched use cases in their comprehensive literature review was "medium fidelity" on a scale from the abstract (low) to precise (high). While of relatively low *fidelity*, these DTs still provide sufficient authenticity to create value for customers, so that providers can create positive returns on investment for all stakeholders by keeping the costs for development, implementation, and use correspondingly low. Accordingly, the Industrial Internet Consortium (IIC) provides a comparably simple and understandable definition of DT, namely a "digital representation, sufficient to meet the requirements of a set of use cases" [12]. Similarly, Minerva et al. [34] argue when they mention that a DT should be designed and implemented with a set of goals and purposes regarding the use case in which to operate, requiring only those properties and characteristics that are necessary to be a sufficient representation of the real-world *counterpart*. Therefore, we propose to classify DT definitions according to whether they mention (i) *one-to-one fidelity* to reality or (ii) *sufficient fidelity* for the use case. These characteristics are mutually exclusive since a *sufficient fidelity* excludes a *one-to-one* representation.

Synchronization

The second property that must be distinguished concerning the required degree of authenticity of the DT with the *counterpart* is *synchronization*. As seen from the taxonomy study by van der Valk et al. [21], the existence of *synchronization* between DT and its real-world *counterpart* is a relatively undisputed characteristic. In previous definitions, a *real-time* connection was almost exclusively required (e.g., [10]). Even though more recent definitions are less strict about this characteristic, *real-time synchronization* is still often used as a descriptive characteristic of a DT (e.g., [18, 19, 77–79]). However, when considering authenticity for DT use cases, a differentiated view is required to determine what *synchronization* intervals are necessary to realize a DT that is *sufficient* to create the intended value. Some authors have noted that data synchronization in practice could be continuously or at certain time intervals [46]. For example, Stark and Damerau [29] distinguish the four different levels of weekly, daily, hourly, and *real-time synchronization* in their update frequency dimension. Minerva et al. [34] mention that the real-world *counterpart* needs to be timely represented by the DT and, therefore, sometimes real-time processing, communication, and storage capabilities could be required, while for others daily updates or even longer periods may be acceptable. Long intervals are *sufficient* for many specific applications, especially if additional *synchronizations* can be triggered when certain measured values are exceeded. This is important for practice because fewer *synchronizations* make DTs more economical since lower costs are incurred for data storage, processing, and connectivity solution, especially concerning battery runtimes. Therefore, we propose to classify DT definitions according to whether they mention *synchronization* in (i) *real-time*, (ii) *near-real-time*, or (iii) *periodic*. These characteristics are not exclusive since not all subsystems of a DT must have the same *synchronization* intervals.

Capabilities

DTs can exhibit a whole range of *capabilities*, described in the literature in different classifications and generic terminology. Stark and Damerau [29] defined a dimension related to *capabilities* wherein they distinguish four levels of maturity from static to look-ahead prescriptive. Jones et al. [19] mention the *capabilities* of simulation, modeling, optimization, health monitoring, diagnostics, and prediction. Sharma et al. [46] describe DTs as a powerful methodology with *capabilities* combining real-time modeling, simulation, autonomy, agent-based modeling, machine learning, prototyping, optimization, big data, and forecasting, enabling domain-specific services. Enders and Hoßbach [28] described a property they called "purpose" with simulation,

monitoring, and control as possible characteristics. Van der Valk et al. [52] also use a category of "purpose" in their study of archetypes of DTs and then call the characteristics within "tasks". The DT applications they analyzed fulfilled the following tasks in descending frequency: simulation, condition monitoring, and analysis, forecast and prediction, optimization, representation, data transfer and storage, controlling, machine learning, decision-making, and cost reduction. Kuehner et al. [16] provide a rough categorization of "potential benefits", including state monitoring and tracking, system prediction, system analysis, system prescription, and data management. Landahl et al. [80] note that the common goal of DTs is to support a realistic model of system behavior that can often support already established services such as performance prediction and optimization. Hence, they even use the term "services" for what we see as *capabilities*.

As can be seen from this compilation, there is a proliferation of different classifications and generic terminology in this subfield of DT research. As further elaborated in the following section, we argue that DTs should be described as having several *capabilities* that enable domain-specific services that serve defined purposes. Therefore, we propose to distinguish the property *capabilities* with the following characteristics (i) *simulation*, (ii) *optimization*, (iii) *prediction*, (iv) *detection*, (v) *prevention*, and (vi) *automation*. These characteristics are not exclusive, as they can occur alone or in any combination.

Purpose

The integration between DTs and services is promising, as not only can new services be enabled, but also existing services can be enhanced by the additional data supplied by DTs [30, 80], in particular by relieving the pains and increasing the gains of the actors in new ways [81]. We decided against a property for services, because they are defined and named very differently as they are configured industry and even use case specific. Additionally, the research on how the various components of DTs are encapsulated to services and used is not well established [82]. Last, we see services as merely the connecting element between *capabilities*, through which they are enabled, and the purposes they serve. Therefore, our approach does not consider services as a property but distinguishes a property *purpose*, where its characteristics are not *capabilities* but fundamental *purposes*. Different classifications and generic terminology are used in DT research to describe the *purpose* of DT applications in the sense of intended benefits. Jones et al. [19] provide a rather unstructured list of what they call perceived benefits. These include reducing costs, risk, design time, complexity, and reconfiguration time. Improving after-sales, service, efficiency, maintenance decision-making, security, safety, reliability, manufacturing management, processes and tools,

flexibility, the competitiveness of the manufacturing system, and innovation. For classification with a more moderate number of characteristics with established designations, we have defined the characteristics of the property *purposes* in line with the widely used and accepted overall equipment effectiveness (OEE) concept that evolved into Total Productive Manufacturing (TPM). According to Nakajima [83], OEE measurement is an effective way of analyzing an integrated system. OEE is a function of *performance*, *availability*, and *quality* and is calculated as the product of its three contributing factors [84]. Especially in the context of smart manufacturing and industrial IoT, this encapsulation of DT-based services seems promising. The OEE concept is widespread and clearly defined by several norms, for example, by the VDMA association [85]. To achieve a holistic perspective on DT-based services, the definition of OEE is supplemented with a concept applicable to describe data-based services. Bange [86] elaborates the main contributing factors for services: the *performance* or cost contribution factor describes the relationship between input and output from the view of the customer. Hence, it is the price to pay for the service concerning the value received. *Availability* or time of the service is represented by the time needed between the occurrence at the customer triggering the need and the delivery of the service satisfying the need. The *quality* is affected by the discrepancy between the offered and delivered services. Therefore, we propose to classify DT definitions according to whether they mention *purposes* regarding (i) *performance*, (ii) *availability*, or (iii) *quality*. These characteristics are not exclusive, as they can occur alone or in any combination.

Life Cycle

The commonly used three-phase model of the product *life cycle* [87, 88] includes three distinct phases: design, manufacturing, and distribution are processes associated with the BoL phase; the use phase of a product is considered part of the MoL phase, while recycling, energy recovery, and disposal are located in the EoL phase [89]. There are many studies on the application of DTs along the product *life cycle* (e.g., [82, 90, 91]), and the integration of the DT into PLM (e.g., [92]). Research is focused on the potential of DTs in the BoL and MoL phase, for example, to optimize product design, engineering, shop floor design, supply chain management, customer demand analysis, and service and value proposition design (e.g., [68, 82, 91, 93–95]). Stark and Damerau [29], who also use the split of the life cycle into the three main phases of BoL, MoL, and EoL in their model even state that a DT starts in the BoL phase and is then gradually supplemented with activities in MoL and then EoL. Accordingly, in a recent study from Switzerland, only 15% of the companies surveyed reported using DTs in the

EoL phase, as the focus is mainly on the BoL (79%) and MoL (60%) phases [8]. However, there is also a plethora of possibilities for DT applications in any other PLM discipline, which are not yet fully explored (e.g., [25, 80, 81]). It is agreed that DT-based services can, at least in theory, add value at every stage of the life cycle [19, 34], as DT concepts inherently embrace the whole product life cycle [24]. Therefore, we propose to classify DT definitions according to whether the DT defined is applied in (i) *BoL*, (ii) *MoL*, or (iii) *EoL*. These characteristics are not exclusive, as they can occur alone or in any combination.

Creation

To answer the question of the timing of DT *creation* in the product *life cycle*, Grieves and Vickers [75] expanded the concept in a later work by introducing the DT prototype, DT instance, DT aggregate, and DT environment. Jones et al. [19] mapped these elements of DT defined by Grieves to the product *life cycle* according to Stark [88] to explain the transitions and relationships between them and the real-world *counterpart*. According to their mapping of the DT *life cycle*, a DT begins its life as a prototype in the design phase, which can be understood as a blueprint for creating the instances. These DT instances are created during the realization phase for each real-world *counterpart* and, in their entirety, form the DT aggregate. The DT prototype's *creation* happens before the real-world *counterpart*, while the DT instances are created parallel to the real-world *counterpart's* realization. Both the instances and the aggregate exist within the DT environment - the virtual representation of the environment in which the physical product exists. Similarly, Minerva et al. [34] describe a process of replication of DT and exemplify two different replication patterns whereas one is using a master replica. Also, Stark and Damerau [29] mention that the DT consists of a unique instance that is purpose-specifically derived from a DT master or prototype. Therefore, we propose to distinguish two characteristics regarding the *creation* of the DT in the product *life cycle* in DT definitions, (i) are DT types and DT instances distinguished, and (ii) is the time of the creation of the DT independent of the realization of the real-world *counterpart*. These characteristics are not exclusive, as they can occur alone or in any combination.

Methods

The main objective of this research is to investigate the definitions of DT from the point of view of companies. To achieve this, we defined the main research question, which in turn is divided into seven sub-questions (cf. Chapter 3.1),

and developed the appropriate research approach to answer them (cf. Chapter 3.2)

Research question

The main research question guiding our research was: "Which properties and characteristics are used by companies to define digital twins?" To answer this question, we have divided it into sub-questions that examine sub-aspects. In addition, other interesting questions arose during the data collection and analysis, which we also included in our research. The following enumeration lists the research questions worked on in the order in which they are answered in the Discussion (cf. Chapter 5.1).

1. How common are definitions of digital twins on websites of companies?
2. Which properties and characteristics are used by companies to define digital twins?
3. Which properties and characteristics are used in combination by companies to define digital twins?
4. How similar are the definitions of digital twins of companies?
5. Are there any relevant clusters concerning the definitions of digital twins of companies?
6. Which differences between definitions in research and industry can be identified?
7. Which properties and characteristics should be included in application-oriented digital twin definitions?

Research Procedure

The research procedure used in this paper follows a systematic approach. To investigate different definitions of DTs from company websites, we started with the PRISMA 2020 systematic review approach for databases, registers, and other sources [96]. With this approach, we collected the data for the review from more than 1300 potential data sources. After a multi-stage screening process, 90 definitions were obtained for further analysis (cf. Fig. 2). Based on the resulting 90 definitions, we performed two different analyses. First, we started with a classical approach by coding the different definitions with the developed taxonomy as the coding key (cf. Fig. 1) and subsequent statistical analysis of the codes given to the different definitions. This analysis was performed on the English and German definitions. For the second approach, we chose a natural language processing method to understand the relationship between the different definitions better. Due to technical restrictions, this analysis was only possible on the English definitions. After presenting the data with several plots we then discussed the results by revisiting the research questions from Chapter 3.1. Finally, we summarize the main findings, highlight
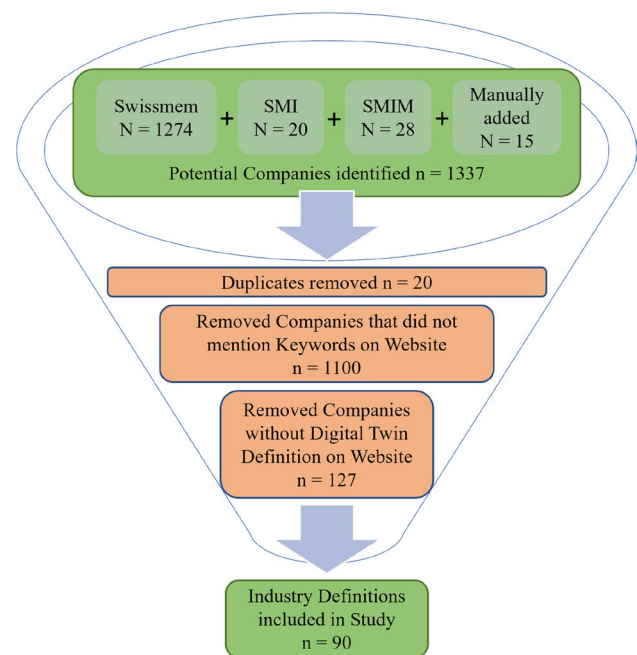

Fig. 2 Identification and screening of industry definitions

the implications, elaborate on the contributions to research and practice, concede limitations, and suggest avenues for further research.

Data Collection

For the first phase of the research procedure consisting of data collection we followed the first step of the PRISMA method [96]. To compile potential sources for definitions of DTs, we created a list of companies from the Swiss Market Index (SMI), the SMI MID (SMIM), and the companies that are members of Swissmem (https://www.swissmem.ch). In addition to those three sources, we added internationally renowned companies based on expert recommendations. This collection resulted in 1337 records that we subsequently screened for definitions (cf. Fig. 2). Using a web crawler, the websites of the companies were searched for the following keywords:

1. *Digital Twin*
2. *Twin*
3. *Digitaler Zwilling*
4. *Zwilling*

The 207 sites that responded positively were manually searched for definitions of DTs. On 80 sites, we could find at least one definition of a DT. The same manual search was done for the 15 sites recommended by experts. We could find ten additional definitions on those websites, which were

then included in the data set. This resulted in a data set of 90 company definitions of DTs for the two analyses.

Supervised Analysis

For the supervised analysis, a coding key based on the developed taxonomy (cf. Fig. 1) was applied to the 90 company definitions. The coding key hence consists of two hierarchy levels, the first level are the properties and the second level are their characteristics as described in Chapter 2. For each definition, the matching properties and characteristics were simultaneously assigned by both authors of this paper. If only properties were named (for example *counterpart*) without naming specific characteristics, then we only set a tag for the property. If specific characteristics were mentioned (e.g., *physical* asset), then we set a tag for the characteristic and additionally for the property (in this case *counterpart*), even if this was not mentioned literally. Therefore, it happened that a tag is set for a property without any tag for a corresponding characteristic. However, if tags have been set for characteristics, they will always be accompanied by the property tag to which they belong. This coding procedure resulted in Table 2, showing the frequency of the properties and characteristics for all 90 DT definitions of the companies. The link to the detailed data set is available at the end of the paper in the declarations. We calculated the proportional occurrence of each property and characteristic based on these results. Furthermore, we calculated the distance between the companies' definitions by comparing the selected properties and characteristics. Finally, using the resulting distance matrix, we performed a multidimensional scaling (MDS) using scikit-learn [97] to reduce dimensions to two and visualized the differences between the different definitions of the companies in a plot.

To identify potential clusters, we assigned at least one industry sector to each company. The following industry sectors have been assigned: 1. Health 2. Industrial manufacturer 3. Software 4. Consulting 5. Building 6. Energy 7. Automation 8. Association 9. Consumer electronics 10. Aviation 11. Transportation.

Unsupervised Analysis

To get a more in-depth understanding of the definitions, we used an unsupervised text-mining approach to compare the similarity between the definitions. In addition, this text-mining approach enables the validation of the supervised analysis, which inevitably has biases due to the human coding of data. However, for this approach, we could only use the definitions in English due to technical limitations, as translations from German to English would alter the data too much. We further prepared the English data set consisting of 60 definitions by removing the unnecessary stop words

Table 2 Frequency of properties and characteristics as a result of the coding process

Properties	Count	Characteristic	Count
Counterpart	85	Only physical	36
		Also non-physical	29
		Any distinct entity	19
Capabilities	61	Simulation	38
		Optimization	26
		Prediction	21
		Detection	11
		Prevention	5
		Automation	2
Purpose	47	Performance	33
		Quality	18
		Availability	12
Data Sources	35	Sensors	28
		Internal System	15
		External System	3
Lifecycle	32	BoL	22
		MoL	20
		EoL	4
Synchronization	31	Real-Time	23
		Near-Real-Time	3
		Periodic	1
Fidelity	30	1:1 representation	19
		Sufficient representation	8
Interfaces	18	H2M	14
		M2M	4
Creation	11	Independent creation of DT	8
		Types / Instance distinguished	4
Data Link	8	Bi-directional	6
		Uni-directional	1

and then tokenizing the definitions by dividing the string into sub-strings. Next, using the word2vec neural network algorithm [98], we analyzed the resulting similarity (distance) of the definitions. As with the supervised data, we used scikit-learns MDS [97] to reduce the dimensions to two and visualized the differences between the different definitions of the companies in a plot.

Results

As outlined in Chapter 3.2, the review process included two phases consisting of a supervised analysis (cf. Chapter 3.2.2) and an unsupervised analysis (cf. Chapter 3.2.3). A total of 90 definitions from companies that had a definition on their website (cf. Chapter 3.2.1) were included in the supervised analysis. The results of the unsupervised analysis are only based on the 60 definitions that were available in English.

Supervised Analysis

In the first step, we analyzed the frequency of codes assigned to the definitions of DTs of companies. As seen in Table 2, each property and characteristic was assigned at least once, but there are considerable differences in frequency. When looking at the properties in isolation, it is noticeable that the *counterpart* was used most frequently in the definition. We could also identify a tendency in the companies' definitions to mention *capabilities* and *purpose*. In addition, we also noticed a tendency for technical aspects, such as *data links* or *interfaces* to be mentioned rather rarely.

In Table 2, the characteristics were grouped by the respective properties to understand their affiliation better. Looking at the three most frequently assigned characteristics, *simulation*, *only physical*, and *performance*, it can be seen that they belong to the top three assigned properties. This is even though for these characteristics, in particular, there are relatively large differences between the number of the most frequently named characteristic and the number of mentions of the corresponding property. For example, the most assigned property *counterpart* is mentioned a total of 85 times, but *only physical* as the most frequently mentioned characteristic within this property is mentioned only 36 times, which corresponds to 42%. If we compare this, for example, with the property *data source*, which is assigned a total of 35 times, *sensors* is mentioned in 80% of the definitions as a concrete source. In addition, we saw some characteristics that were seldom used, like the *data link*'s characteristic *uni-directional* or the *synchronization*'s characteristic *periodic*, both of which we just assigned once.

Following the frequency analysis, we examined the appearance of the different properties in combination. As the property *counterpart* occurs in 94% of the definitions, only a few other properties do not occur in combination with it. Figure 3 shows the relative appearance of the properties, where the relevance of *counterpart* as a property in the definitions gets even more evident.

As seen in Fig. 3, the properties *creation* and *data link* are the only properties that never occur together. All other properties were used at least once together with the others. To better understand the distances between the different properties, we conducted an MDS (cf. Chapter 3.2.2) that projects the differences into two artificially created axes as seen in Fig. 4.

In Fig. 4, it is noticeable that three pairs are relatively close to each other. *Synchronization* and *counterpart*, *data link* and *fidelity*, as well as *interfaces* and *data sources*.

We proceeded in the same way for the coding of the characteristics. As with the properties, we calculated the relative appearance of the characteristics as seen in Fig. 5. Similar to the results regarding the properties, certain combinations are more common than others. To illustrate

Fig. 3 Relative frequency of the properties in dependence on each other
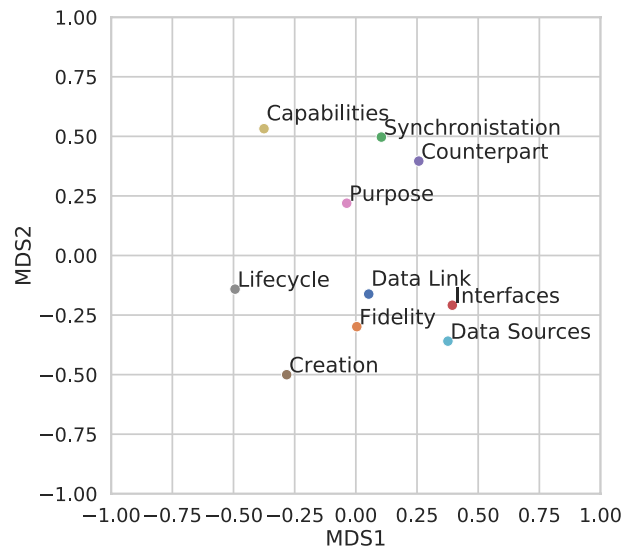
Fig. 4 Distance between properties based on MDS

the relative distance between the characteristics, we again used MDS (cf. Chapter 3.2.2) resulting in Fig. 6.

Compared to the properties, only a few apparent combinations could be found in the characteristics, but this could also be due to the larger number of parameters. However, the closeness of some of the characteristics is striking. For example, *non-physical* and *H2M interfaces* are relatively close together. Furthermore, the proximity

Fig. 5 Relative frequency of the characteristics in dependence on each other

	Uni-directional	Bi-directional	1:1 representation	Sufficient representation	Real-Time	Near-Real-Time	Periodic	H2M	M2M	Only physical	Also non physical	Any distinct entity	Types / Instance distinguished	Independent creation of DT	Performance	Availability	Quality	BoL	MoL	EoL	Prediction	Automation	Detection	Prevention	Optimization	Simulation	Thing	Internal System	External System
Uni-directional	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Bi-directional	0	1	0	.17	.5	0	0	.5	.17	.83	.17	0	0	0	.17	.17	.33	.17	.33	0	0	.17	.17	0	.17	.33	.83	.17	0
1:1 representation	0	0	1	0	.26	.05	0	.16	.11	.42	.42	.16	.05	.21	.53	.26	.37	.42	.26	0	0	.16	.05	.42	.26	.63	.37	.42	.11
Sufficient representation	0	.12	0	1	.12	.12	0	.25	.12	.38	.38	.12	0	.12	.25	0	0	0	0	.12	.12	0	0	.12	.38	.62	.25	0	0
Real-Time	0	.13	.22	.04	1	.13	0	.17	.09	.22	.39	.35	.09	.09	.43	.22	.26	.39	.35	.04	.04	.13	.09	.43	.48	.57	.35	.26	.04
Near-Real-Time	0	0	.33	.33	1	1	0	.67	0	.33	.33	.33	0	0	1	.33	.67	.33	.33	0	0	0	0	.67	.33	.33	.67	.67	.33
Periodic	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1	0	0	1	0
H2M	0	.21	.21	.14	.29	.14	0	1	.07	.43	.43	.07	0	.07	.36	.21	.29	.07	.14	0	0	.14	.07	.14	.29	.29	.64	.36	.07
M2M	0	.25	.5	.25	.5	0	0	.25	1	.5	.5	0	.25	.25	0	0	.25	.25	0	0	0	0	0	0	.5	.5	.5	.5	0
Only physical	0	.14	.22	.08	.14	0	0	.17	.06	1	0	0	.03	.06	.44	.14	.28	.22	.22	0	.03	.19	.06	.19	.22	.28	.42	.11	0
Also non physical	.03	.03	.28	.1	.31	.03	0	.21	.07	0	1	0	.07	.14	.34	.1	.21	.24	.21	.07	0	.03	.03	.41	.21	.55	.24	.21	.1
Any distinct entity	0	0	.16	.05	.42	.05	0	.05	0	0	0	1	.05	.05	.32	.16	.05	.32	.26	.11	.05	.11	.11	.32	.32	.53	.21	.21	0
Types / Instance distinguished	0	0	.25	0	.5	0	0	0	.25	.25	.25	.25	1	.25	.25	0	.25	.25	.5	0	0	.25	0	.25	0	.5	.25	.5	0
Independent creation of DT	0	0	.5	.12	.25	0	0	.12	.12	.25	.5	.12	.12	1	.25	.12	.5	.62	.5	.12	0	0	0	.38	.12	.75	.38	.75	.12
Performance	0	.03	.3	.06	.3	.09	.03	.15	0	.48	.3	.18	.03	.06	1	.24	.36	.42	.36	.12	0	.15	.06	.52	.39	.55	.39	.27	.06
Availability	0	.08	.42	0	.42	.08	.08	.25	0	.42	.25	.25	0	.08	.67	1	.5	.33	.33	.08	0	.33	.25	.5	.75	.5	.5	.42	.08
Quality	0	.11	.39	0	.33	.11	0	.22	.06	.56	.33	.06	.06	.22	.67	.33	1	.44	.33	.17	0	.17	.06	.39	.39	.61	.44	.28	.06
BoL	0	.05	.36	0	.41	.05	0	.05	.05	.36	.32	.27	.05	.23	.64	.18	.36	1	.68	.18	0	.23	.09	.55	.27	.64	.32	.32	0
MoL	0	.1	.25	0	.4	.05	0	.1	0	.4	.3	.25	.1	.2	.6	.2	.3	.75	1	.2	0	.3	.1	.4	.3	.6	.45	.3	0
EoL	0	0	0	0	.25	0	0	0	0	0	.5	.5	0	.25	1	.25	.75	1	1	1	0	.25	.25	.75	.5	1	.5	.25	0
Prediction	0	0	.5	.5	0	0	0	0	.5	0	.5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Automation	0	.09	.27	.09	.27	0	0	.18	0	.64	.09	.18	.09	0	.45	.36	.27	.45	.55	0	0	1	.27	.36	.36	.45	.73	.27	0
Detection	0	.2	.2	0	.4	0	0	.2	0	.4	.2	.4	0	0	.4	.6	.2	.4	.4	.2	0	.6	1	.6	.8	.4	.6	.2	0
Prevention	0	0	.31	0	.38	.08	.04	.08	.04	.27	.46	.23	0	.12	.65	.23	.27	.46	.31	.12	0	.15	.12	1	.42	.5	.27	.27	.12
Optimization	0	.05	.24	.05	.52	.05	.05	.19	0	.38	.29	.29	0	.05	.62	.43	.33	.29	.29	.1	0	.19	.19	.52	1	.57	.48	.24	.1
Simulation	.03	.05	.32	.08	.34	.03	0	.11	.05	.26	.42	.26	.05	.16	.47	.16	.29	.37	.32	.11	0	.13	.05	.34	.32	1	.37	.26	.08
Thing	0	.18	.25	.18	.29	.07	0	.32	.07	.54	.25	.14	.04	.11	.46	.21	.29	.25	.32	.07	0	.29	.11	.25	.36	.5	1	.32	.07
Internal System	0	.07	.53	.13	.4	.13	.07	.33	.13	.27	.4	.27	.13	.4	.6	.33	.33	.47	.4	.07	0	.2	.07	.47	.33	.67	.6	1	.13
External System	0	0	.67	0	.33	.33	0	.33	0	0	1	0	0	.33	.67	.33	.33	0	0	0	0	0	0	1	.67	1	.67	.67	1

of *uni-directional* and *types/instances distinguished* is striking.

To further examine the results, we focused on the distance between the individual companies. In the first step, the distance between all coding vectors of the individual companies was calculated. This allowed us to carry out MDS, resulting in the plot shown in Fig. 7.

To gain further insights and validate the results of the supervised analysis based on manual coding, we compared the English definitions with a natural language processing method, representing an unsupervised analysis method.

Unsupervised Analysis

We produced a distance matrix for all companies with English definitions by using a natural language processing approach to calculate the distances between the definitions. One of the main reasons to analyze the definitions with the help of word2vec was, as described in Chapter 3.2.3, to define a baseline to reduce or classify the bias of the results in Chapter 4.1 caused by the manual coding in the supervised approach. We also carried out an MDS to compare these results as can be seen in Fig. 8. Even though the results in Figs. 7 and 8 cannot be compared one-to-one, as the position after the dimension reduction is not comparable, we assume that significant differences are better visible this way. It is essential to mention that the axes of the plots in Fig. 7 and 8 cannot be compared. The dimensionality reduction only tries to keep the distance between the companies with as little loss of information as possible. Therefore, the absolute position in the plot is not relevant; only the relative distance to the other companies is relevant.

After comparing the plots of the unsupervised analysis with the supervised analysis, it can be seen that there are similar distances between most companies. For example, if we look at WinGD and Siemens Logistics, we see a similar relative distance in both analyses. The same applies when comparing Wärtsilä Services and Siemens Healthcare, where a relatively large distance is evident in both analyses. However, there are also pairs of companies where this is not the case. For example, when comparing ABB and Huawei 2, there is a noticeable difference in the distance from the supervised to the unsupervised method.

Fig. 6 Distance between characteristics (grouped by properties) based on MDS

Discussion

The discussion of the results of this study has two parts. First, we reexamine the research questions from Chapter 3.1 and answer them. Second, we propose an application-oriented definition for DTs based on the findings of this study.

Research Questions

In this section, we return to the research questions from Chapter 4 and answer them based on the results from the DT definitions of companies examined.

How common are definitions of digital twins on websites of companies?

Rather surprisingly, we could only find a few publicly available definitions for DT of companies on the 1317 websites we searched for the study (cf. Figure 2). The 90 definitions we could find represent only about 7.5% of the total searched. This is especially noteworthy, as in previous study 23% of Swiss companies stated that they are already using DTs to create value, as already mentioned in Chapter 1.

These 23% and some of another 25% that stated in the study that they are working on an implementation would be expected to communicate their DT activities on their websites. We have some guesses how this difference between communication and application occurs. One possibility is that the DT applications in many of the companies analyzed are not yet developed enough to offer them to their customers and partners. We also suspect that some companies use DT only to optimize internal processes and therefore do not communicate these applications publicly. The study by Barth et al. [8] supports this assumption, which found that DT is only used for marketing purposes in fifth out of six places. Our study also suggests that the DT concept is not a marketing buzzword for companies surveyed.

Which properties and characteristics are used by companies to define digital twins?

All properties defined in our taxonomy appeared at least once in the evaluation. Nevertheless, as expected, some were used much more often than others. The most used property is *counterpart* (cf. Table 2), for which a high usage was expected since it has been one of the central elements of

Fig. 7 Distance between companies (grouped by industry) based on the MDS of properties and characteristics (supervised)
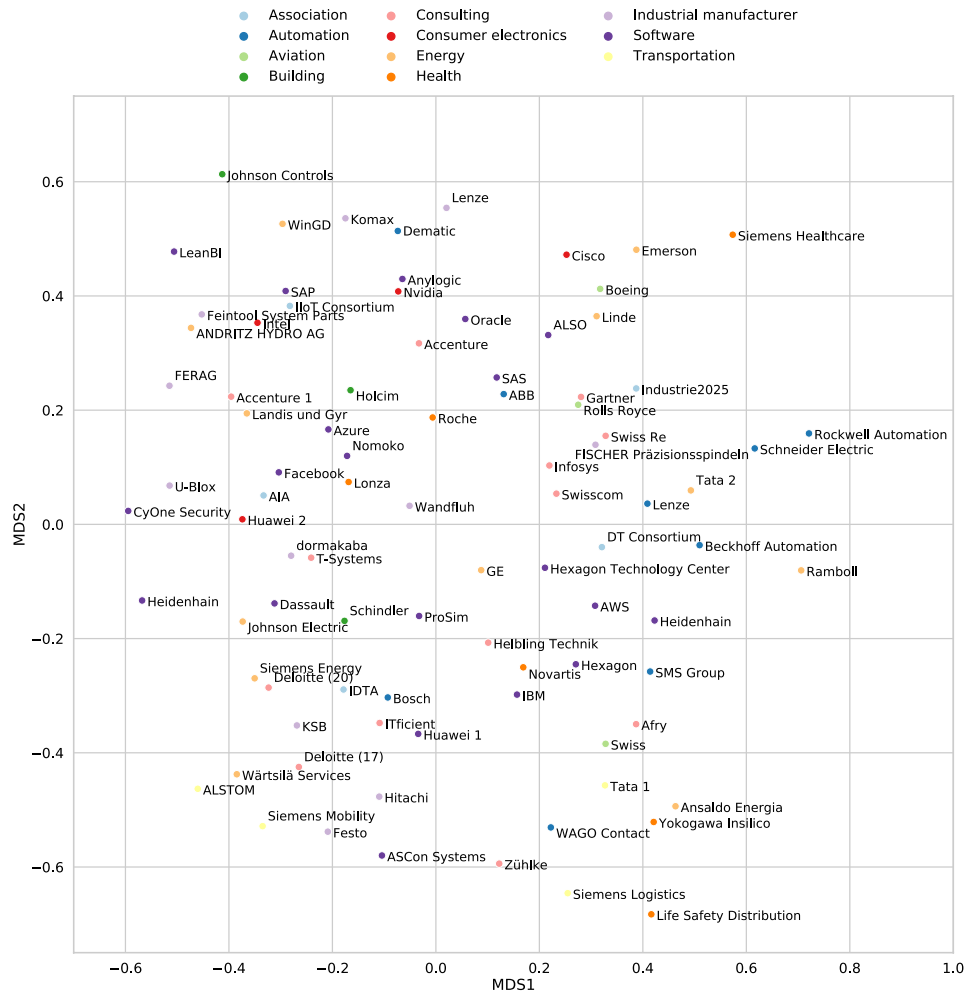
the DT concept since its origin. What was striking was the number of mentions of the characteristic *non-physical* or *any distinct entity*. Based on scientific literature reviews, we expected a stronger focus on *physical* objects. Another difference to the literature is the properties *interfaces* and *data link*. These two were underrepresented in the companies' definitions compared to their presence in the literature. On the other hand, the properties *capabilities* and *purpose* were rather overrepresented. We might assume that these differences have resulted from the exclusive use of text from the companies websites. However, the answer to the previous question shows that companies tend not to use the term as a marketing buzzword. Anyhow, even if this approach leads to a marketing bias, our results show what seems essential to companies in communicating their DT to the customer—*capabilities*, and *purpose*. A somewhat expected result is observed when looking at the individual characteristics of the *capabilities*. *Simulation*, *optimization*, and *prediction* are used the most, which seems to align with scientific literature. However, comparing the mentioned characteristics regarding the property *purpose*, it is noticeable that besides

performance, which was listed most frequently, *quality* is used more often than *availability*. We would have expected this in reverse order, primarily due to the strong focus on companies from the industrial sector. We suspect that this reflects the Swiss industry's strong focus on quality.

The two properties, *synchronization*, and *fidelity*, which represent the topic of authenticity, are only used in roughly one-third of the definitions analyzed. It seems like companies are not using technical issues to define and communicate their DT concepts on their website as they focus on properties relevant to their customers. What is surprising regarding these two properties is, however, how often companies use the characteristics of *one-to-one* representation regarding *fidelity* and *real-time synchronization*, even though they pose significant technical challenges.

Regarding the property of product *life cycle*, we can confirm the findings of Barth et al. [8] that *EoL* is much less relevant to companies than *BoL* and *MoL* as it is mentioned very rarely compared to the other two. However, we expect that the ongoing trend of sustainability and circular economy topics will undoubtedly lead to growing interest in this area.

Fig. 8 Distance between companies (grouped by industry) based on the MDS of distance matrix from word2vec (unsupervised)
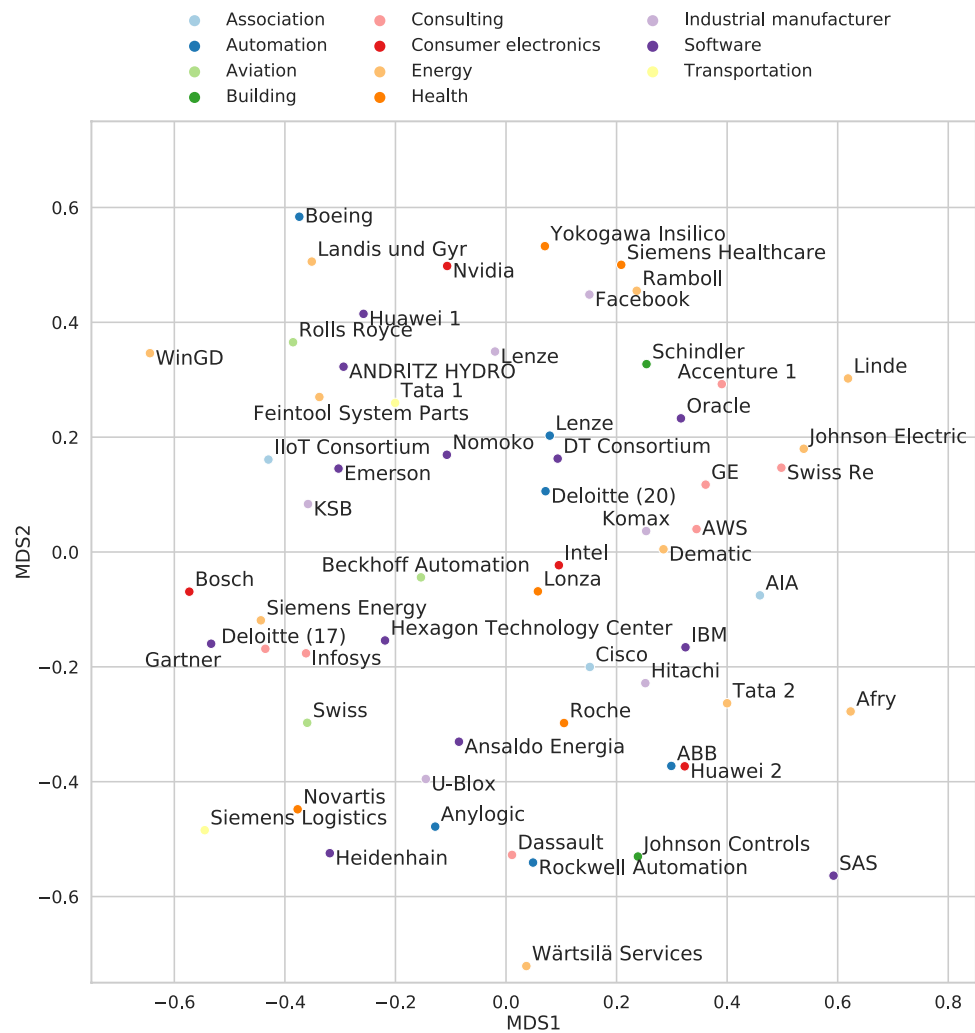
*Another remarkable finding is the rare use of *external data* as a *data source* for DTs. We suspect this underrepresentation is because companies are currently still busy integrating *internal systems* and need more time to incorporate *external data*.

Which properties and characteristics are used in combination by companies to define digital twins?

The combination of different properties and characteristics can be derived from the heatmaps in Figs. 3 and 5 and from the MDS scatterplots in Figs. 4 and 6.

Considering only the properties first, we see different frequencies of occurrence in the definitions (cf. Figures 4 and 3). Therefore, we assume implicit importance of the properties for the companies. Due to the high frequency of the property *counterpart* (occurs in 94% of the definitions), we refrain from an in-depth discussion of the relevance of this property. If we look at the properties *purpose* and *capabilities*, we see that they often occur in combination with the other properties. It can be concluded from this that the two

properties are very relevant for companies in their communication with customers. It can also be seen that *interface* often occurs with *data sources*, which can also be seen from the small distance in Fig. 4, which is not surprising as *data sources* need *interfaces* to be accessed.

Less clear patterns emerge when examining the characteristics that companies use to define DTs (cf. Figures 6 and 5). In the following, we focus on the three most frequently used characteristics. Starting with *simulation*, the characteristic that appeared the most in the companies' definitions, we see that it was used in combination with *performance* in 47% of the cases, *non-physical* in 42% of the cases, and *BoL* as well as *sensors* in 37% of cases. Considering the characteristic *automation*, it is striking that it was used in combination with *only physical* in 64% of the cases. Examining the combinations in which *automation* occurs, we see that in most cases, it is used with *only physical* (64%) and only in a few with *non-physical* (9%) or *any distinct entity* (18%). This seems surprising, as *automation* is primarily associated with processes and not necessarily physical assets. Three

characteristics always occur in combination with *performance*: (*near-real-time*, *periodic*, and *EoL*). This seems surprising at first sight, but since the absolute numbers are low, we assume random accumulation here.

How similar are the definitions of the companies?

To answer this question, we examine the two Figs. 7 and 8. It is evident that although all companies define the same concept, very different texts are used. Thus, our study confirms that there is still no common understanding regarding the definition of the DT concept in practice.

It is particularly interesting in this respect that even individual companies within the same corporation use different definitions. For example, in our analysis, shown in Fig. 7, the Siemens Group is represented by four companies: Energy, Mobility, Logistics, and Healthcare. While Energy, Mobility, and Logistics are relatively close to each other, Siemens defines the DT concept for Healthcare very differently. Siemens is undisputedly one of the pioneers concerning the use of DTs in industry. Therefore, it can be assumed that these different definitions are no coincidence or the result of insufficient coordination. Rather, it seems more effective for companies to interpret and define the DT concept in an industry-specific way to tailor it to use cases in their particular industry. In addition to Siemens, our analysis includes other corporations with several companies and definitions, some of which are relatively close to each other, for example, Hexagon or Deloitte. Whether the use of distinctly different, industry-specific definitions indicates a group's maturity level regarding DT technology and means a competitive advantage can only be assumed based on the present results.

Are there any relevant clusters concerning the definitions of digital twins of companies?

Before the evaluation, we assumed there would be clusters, especially considering different industries. However, we could not find conspicuous clusters in our evaluation of the results. As can be seen clearly in the healthcare industry, for example, the definitions of companies in an industry are divergent. Since we could not detect any obvious clusters, we refrained from a deeper, more elaborate search for them. Because as St. Pierre and Jackson [99] put it colloquially: one should not search for patterns in the data where there are none. It, therefore, appears that no common understanding of the DT concept has yet been established within individual industries. Perhaps companies like Siemens, which use industry-specific definitions, could have an influence, and clusters could form around these definitions in the future.

Which differences between definitions in research and industry can be identified?

Some points stand out when comparing the results of this review with the results of van der Valk et al. [21], who compared 233 definitions from academic sources in their review. Not all properties in our application-oriented taxonomy used for this review are one-to-one comparable with the ones used in the review of van der Valk et al. [21]. Therefore the following properties are excluded from the discussion: *purpose*, *capabilities*, *life cycle*, and *data sources*.

However, the studies have significant differences if one compares them on a property level, for example, regarding the property *data link*. In van der Valk et al. [21], 199 out of 233 sources (85%) use this property, while in our study, only 8 out of 90 (8%) companies mentioned *data link*. A similar but not quite as substantial difference can be observed for the property *fidelity*, which was used in 74% of the cases in the review of van der Valk et al. [21] and in 33% of the definitions in our review. Differences between the scientific publications and the companies' definitions are also evident when comparing the properties *synchronization*, *interfaces*, *counterpart*, and *creation*. There is a clear difference between the results two reviews and, thus, between research and practice regarding the defining properties of DTs.

Interestingly, the differences in the properties are not apparent at the level of characteristics. For example, regarding the property *fidelity* the ratio of the characteristics *one-to-one* representation to *sufficient* representation is similar in both reviews (70% in our review versus 71% in the review from van der Valk et al. [21]). An additional example is the ratio of the characteristics *non-physical* to *only physical* of the property *counterpart*, which is also very similar (40% versus 42%). This pattern can also be found with the other characteristics. Therefore, we conclude that there is a smaller gap between research and industry for characteristics than with properties.

Which properties and characteristics should be included in application-oriented digital twin definitions?

After ensuring both practical relevance and scientific rigor by explaining the origin and evolution of the definitions, analyzing the properties and characteristics, and examining the definitions used in practice, we propose a new, application-oriented definition of DT in the next section to answer this research question. Our research and the discussed example of Siemens show that companies should develop and apply use case-specific definitions. The application-oriented definition and our taxonomy provide a tool for companies to tailor individual definitions.

Application-Oriented Definition

There is a consensus that a DT is a digital representation of a real-world *counterpart*, which can receive and provide data to create value within a use case. However, there is no common understanding both in research and in practice of

how to define DTs regarding specific properties and characteristics. We, therefore, propose the following application-oriented definition for DTs, which contains the necessary DT-defining properties and characteristics, but allows use-case-specific adaptations for others:

A DT is a sufficiently authentic digital representation of a distinct real-world entity that exists as a prototype from which instances to accompany those real-world entities are derived. It has interfaces to communicate with users bidirectionally and receives raw and preprocessed data to provide data, information, and services to create value within a specific use case.

Table 9 shows the definition with explanations of the properties considered and comments on their interpretation.

Conclusion

In this concluding chapter, we summarize the main findings, highlight the implications, elaborate the contributions to research and practice, concede limitations, and suggest avenues for further research.

Main Findings

Only a few (7.5%) of the companies we investigated published their definition of a DT on their website. Two conclusions can be drawn from this. Firstly, despite the wide use of 23% based on the data from Barth et al. [8], the DT is not used as a differentiator on the websites of the companies. Therefore, we would argue that it is not solely used as a marketing tool. And secondly, it could also be that the DT has not yet reached the standard portfolio of the companies. Concerning the definitions, it is remarkable how often the characteristics *non-physical* or *any distinct entity* were found. This led us to the conclusion that despite the focus of the research on physical twins, companies are diligently working on twins for *non-physical* entities. Furthermore, it also became clear that the standard from the research literature to achieve a *one-to-one* representation when it comes to *fidelity* is not practical for companies. Therefore, they evade using the term 1:1 representation in their definitions. When we compared the different definitions of the companies, it was rather surprising that no clusters were found. However, we could find clear differences when it comes to definitions inside corporate groups, for example, in the case of Siemens. This shows us that it is important to define the DT to the needs of the ecosystem a company is navigating in and the corresponding use cases. Some points stand out if we compare the industry definitions with those from research. Companies tend to use more value-based properties, such as *capabilities* and *purpose*, that are not included or not precisely delineated in academic reviews. On the other hand, technical aspects like *data link* and *interfaces* are mentioned much less by the companies. These differences are mostly seen at the property level, but if we go into more detail and compare the characteristics between the definitions, we no longer see such big differences.

Implications

The proposed taxonomy and definition are intended more for practitioners than for researchers. In this respect, the paramount implication from our research is that any company

Detailed definition of a Digital Twin

Properties	Definition	Comments / Explanatory Notes
	A digital twin …	
Authenticity	… is a sufficient authentic digital representation…	Regarding fidelity and synchronization
Counterpart	... of a distinct real-world entity…	Entity with i) a recognizably distinct existence and ii) which are relevant for the creation of value
Creation	… that exists as a prototype…	Containing all the information required to create the real world entity, as well as the DT instance
	… from which instances to accompany these real-world entities are derived.	A specific instance of an individual entity that remains linked to it throughout its life
Communication	… has interfaces to communicate with users bidirectionally...	Users include humans via Human-Machine-Interfaces (HMI) as well as other objects and DTs via Machine-to-Machine-Interfaces (M2M)
	… receives raw and preprocessed data...	From own sensors, internal systems, external systems or users
	… to provide data, information and services…	Reactively or proactively offered
Purpose	… to create value…	For defined stakeholders
	… within a specific use case.	Is individually customized for use cases

Fig. 9 Proposal for a application-oriented definition

that wants to use DTs for value creation must consider and clearly define the appropriate DT for their use case. To define a use case-specific DT application, companies can use the taxonomy with the properties and characteristics we defined for our research (cf. Table 1) in the sense of a morphological box to create a tailored definition to implement. As a first step, a company should define in particular the real-world *counterpart* to be represented by the DT and the intended *purposes* along the *life cycle*. As a second step, the company should derive the necessary *capabilities*, *data sources*, *data links*, *interfaces*, and *creation* logic as well as the *sufficient* authenticity in terms of *fidelity* and *synchronization*. The second step is crucial as businesses need a simple, cost-effective DT to implement and apply but with *sufficient* authenticity to deliver meaningful insights. The central challenges of defining an appropriate DT that delivers positive net returns are highlighted in the following paragraphs.

Capabilities and *purposes* are the second and third most used properties found in industry definitions of DT. However, research struggles to provide precise and systematic classifications and terminology about the benefits of DTs in practice. Therefore, we recommend companies to define the intended benefits of their DT application along the following logic: What *purposes* will the DT serve? What services are needed for these *purposes*? What *capabilities* must the DT have to realize these services? This approach ensures that the development of DT applications is always focused on creating relevant value and that no effort is wasted on *capabilities* that are unnecessary and thus negatively impact the return on investment. This step is critical, because it sets the course for many later decisions, for example, regarding the necessary communication capabilities and the *sufficient* level of authenticity of the DT.

In designing the communication capabilities of the DT, the two properties *data link* and *interfaces*, are the focus of interest. With regard to the *data link*, the question arises in particular as to how the link from DT back to the real-world *counterpart* is to be designed: Should decisions and actions be transmitted directly and automatically or is a human needed in the loop? Both approaches have different advantages and disadvantages depending on the application context, especially with regard to costs and risks, which must be weighed against each other. Our research also suggests to design a DT application with a human in the loop at the beginning to train the DT and avoid expensive mistakes in the implementation phase. The considerations and decisions regarding *interfaces* are closely related to the decisions regarding *data links*. If humans are to be in the loop, it must be defined via which *interfaces* they are to interact with the DT. In addition, it must be clarified in any case via which *M2M interfaces* the DT should be able to communicate with its real-world *counterpart* and other DTs in the CPS.

When discussing which authenticity level of DT regarding its *counterpart* in the real world is *sufficient* for the planned use case, *fidelity* and *synchronization* are the two essential properties to discuss. *Fidelity* is important as it governs the processes that can be performed. Namely, the closer the alignment between the virtual and physical twins, as quantified by the *fidelity*, the more potent the *simulation*, modeling, and *optimization capabilities* become [19]. Some implementations of DT may contain many attributes and data, as well as computational, geometrical, visualization, and other models to satisfy the application requirements. Some others may only need a small set of attributes and data to be *sufficient* to support their application. With regard to *synchronization*, the focus is on the question of what information is needed at what intervals. In many cases, the need for information can be significantly reduced by a rule-based approach in which data is only sent when specific threshold values are exceeded. A self-adaptive DT would even be able to adjust these limits to ensure that it always uses as much data as necessary but as little as possible for its services. Fewer *synchronizations* make DTs more economical since lower costs are incurred for data communication, including data storage, processing, and connectivity solutions. Fewer *synchronizations* furthermore increase the battery runtime, which is especially important in remote systems.

Contribution

Our investigation of the understanding of DTs in industry, the taxonomy developed for coding the definitions, and the proposed application-oriented definition contribute to research and practice.

To research

The investigation of the definitions of DT in the industry is a unique approach to contribute to a still young area of research with limited maturity. As can be seen from the literature review, research is increasingly shifting towards applications and DTs in specific industries, even if no final consensus on theoretical considerations and frameworks has yet been found. Therefore, it is time to systematically investigate what understanding of the DT concept exists in companies by analyzing their definitions instead of reviewing and summarizing scientific literature. Our analysis contributes to the research not only because of its content focus and unique approach in terms of the data sources used but also through the application of a systematic, distinctive methodology. In addition to the systematic supervised coding with the help of the created taxonomy, an unsupervised approach with neural network algorithms and multi-dimensional scaling was applied to compare the definitions, which has not been done before in this research field.

Further, we contribute to clarification and standardization of terms for properties and characteristics of DTs.

Especially with regard to the properties *counterpart*, *data link*, *fidelity*, *synchronization*, *capabilities* and *purpose*, where the discussion in research is far from being concluded, we make valuable contributions. With regard to the *counterpart*, based on our explanations and analyses, the conclusion should prevail that any real-world entity with a recognizably distinct existence and relevance for creating value in a specific use case can be represented by a DT. With respect to *data link*, the considerations of human involvement in the *bi-directional* connections between DT and the real world are worth remarking. We consider our findings regarding the development of *sufficiently* authentic DTs with respect to *fidelity* and *synchronization* to be of particular importance as a contribution to the discussion in research. Another valuable contribution is made with respect to the benefits of DTs. We argue that DTs should be described as having several *capabilities* that enable domain-specific services that serve defined *purposes*.

Many researchers have called for a shared definition to fill the research gaps described at the beginning of this paper. The proposed definition answers, among others, the call from Golovatchev [58] for a definition that is

1. generic enough to be used in various application areas and, at the same time,
2. extends the various existing partial views of the DT concept.

Further, our definition is future-oriented and, with its modular structure for customization, geared to practical implementation. Together with the developed taxonomy, it helps to compare approaches, applications, and definitions in different research fields. In summary, we contribute in several ways to the discussion of the research community about the defining properties and characteristics of DTs and thus to the development of a common understanding of this much-debated concept.

To practice

Our explanations, analyses, and results enable companies to gain an overview of an application-oriented, future-oriented understanding of defining properties and characteristics of DTs. This allows them to develop an understanding of how other companies define DTs, which in turn helps them contextualize their own understanding.

A particularly large contribution to practice is the insight that every company needs a definition of DT adapted to its own requirements. In the case of large companies, even multiple definitions should be used for different application areas. Ideally, companies are able to develop a specific definition of the adequate DT application for each use case. Our research makes an important contribution to the development of these capabilities by providing companies with an application-oriented and adaptable definition and the taxonomy as a tool for adaptation.

By discussing the results in the implications section, we give valuable guidance on how the provided tools could be used, respectively, and which dependencies have to be considered regarding decisions determining the adequate characteristics of specific DT applications. The valuable explanations regarding the properties and characteristics enable companies to develop a DT that is *sufficient* for the use case. This understanding and the tools provided enable a systematic derivation of the properties and characteristics necessary for their use cases, thus allowing companies to develop DTs that generate a positive return on investment in practice. As a result, DT applications will become more attractive for many companies, and they will consider investing in corresponding developments. Our research, therefore, makes an important contribution to the widespread use and implementation of DTs in practice.

Limitations

This study is subject to certain limitations listed below and outlined thereafter.

1. Narrative derivation of properties and characteristics
2. Includes only publicly available definitions
3. Includes mainly Swiss companies
4. Human bias in the coding process
5. Incongruence of proposed definition with existing definitions

The taxonomy was deliberately not derived on the basis of a systematic comparison of existing scientific literature on the properties and characteristics of DTs. Because the aim was to create a taxonomy that allows to examine and compare practice-oriented definitions from the industry, which is why a narrative derivation of the properties and characteristics was chosen, this limitation is mitigated by the broad discussion of a large number of relevant and recent publications in the derivation of the taxonomy.

The first limitation considering the review process is that we only included definitions from companies that could be found on their public websites. Therefore, we must note a certain bias toward marketing formulations. Instead, companies could be asked to provide their definition in written form, which would lead to a different but smaller bias since it is an assignment. The second limitation is the focus on the Swiss market and the industry association Swissmem, which leads to a particular bias towards industrial companies and certain mindsets prevalent in Switzerland. The third limitation of the study is the subjectivity of the coding definitions. We structured the coding process accordingly to avoid this as much as possible. In addition, we built another control into

our procedure with the unsupervised definition comparison method.

Finally, the proposed application-oriented definition for a common understanding of DTs is also subject to certain limitations. These derive mainly from the fact that, in contrast to many other publications, a future- and practice-oriented definition was sought. The resulting definition is thus not congruent with many existing definitions with respect to some properties and characteristics, since it considers, for example, a relatively broader definition of DT with respect to the *counterpart* to be represented and its authenticity in terms of *fidelity* and *synchronization*.

Further Research

Following the design-oriented research approach, the utility of the developed application-oriented definition and taxonomy decreases over time. Especially in research areas with high dynamics, as with DTs, further investigations are necessary to adapt and refine such artifacts.

1. The presented application-oriented definition and the use of the taxonomy as a morphological box to develop customized definitions need to be evaluated with companies. This way, the properties, and characteristics can be validated and further developed.
2. The analysis of definitions in practice should also be extended to support further iterations. As our results have shown, unsupervised analyses provide comparable results to supervised analyses. Thus, a large number of definitions can be included without creating a lot of manual work. Therefore, another study that does not focus only on the Swiss industrial sector should be performed.
3. A questionnaire could be used to collect definitions that are not available on websites for further research. The definitions collected in this way might be significantly less marketing- or customer-oriented, which could make a comparison with the results presented here particularly interesting.
4. Furthermore, the same supervised as well as the unsupervised approach could be used to analyze definitions from the scientific literature. This could provide further insight into the differences in the understanding of research clusters and thus contribute to the consolidation of the concept und definition of DT.

Declarations

References

1. Dawid H, Decker R, Hermann T, Jahnke H, Klat W, König R, Stummer C. Management science in the era of smart consumer products: challenges and research perspectives. Central Euro J Oper Res. 2017;25(1):203–30. https://doi.org/10.1007/s10100-016-0436-9.
2. Porter ME, Heppelmann JE. How smart, connected products are transforming competition. Harvard Busin Rev. 2014;92(11):64–88.
3. Porter ME, Heppelmann JE. How smart, connected products are transforming companies. Harvard Busin Rev. 2015;93(10):96–114.
4. Wuenderlich NV, Heinonen K, Ostrom AL, Patricio L, Sousa R, Voss C, Lemmink JGAM. "Futurizing'' smart service: implications for service researchers and managers. J Services Market. 2015;29(6/7):442–7. https://doi.org/10.1108/JSM-01-2015-0040.
5. Barbieri C, West S, Rapaccini M, Meierhofer J Are practitioners and literature aligned about digital twin? In: 26th EurOMA Conference Operations Adding Value to Society, pp. 17–19 (2019)
6. Liu X, Jiang D, Tao B, Xiang F, Jiang G, Sun Y, Kong J, Li G. A systematic review of digital twin about physical entities, virtual models, twin data, and applications. Adv Eng Inform. 2023;55: 101876. https://doi.org/10.1016/J.AEI.2023.101876.
7. Barth L, Ehrat M, Fuchs R, Haarmann J Systematization of Digital Twins: Ontology and Conceptual Framework. In: Proceedings of the 2020 The 3rd International Conference on Information Science and System, pp. 13–23 (2020). https://doi.org/10.1145/3388176.3388209
8. Barth L, Ehrat M, Galeno G, Holler M, Savic N Value Generation in the Product Lifecycle with Digital Twins: Status Quo in Swiss Companies. In: 55th Hawaii International Conference on System Sciences (HICSS), pp. 4528–4537. University of Hawaii at Manoa (2022). https://doi.org/10.24251/HICSS.2022.551
9. Grieves M. Digital twin: manufacturing excellence through virtual factory replication. White paper. 2014;1:1–7.
10. Glaessgen E, Stargel D The digital twin paradigm for future NASA and US Air Force vehicles. In: 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference

20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA, p. 1818 (2012)

11. Eisenträger M, Adler S, Kennel M, Möser S Changeability in Engineering. In: 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC), pp. 1–8 (2018). https://doi.org/10.1109/ICE.2018.8436295

12. Boss B, Malakuti S, Lin SW, Usländer T, Clauer E, Hoffmeister M, Stojanovic L, Flubacher B Digital twin and asset administration shell concepts and application in the industrial internet and industrie 4.0. Plattform Industrie 4, 13–14 (2020)

13. Tomczyk M, van der Valk H Digital Twin Paradigm Shift: The Journey of the Digital Twin Definition. In: Proceedings of the 24th International Conference on Enterprise Information Systems - Volume 2: ICEIS,, pp. 90–97. SciTePress (2022). https://doi.org/10.5220/0010997600003179

14. Bécue A, Fourastier Y, Praça I, Savarit A, Baron C, Gradussofs B, Pouille E, Thomas C CyberFactory#1 —- Securing the industry 4.0 with cyber-ranges and digital twins. In: 2018 14th IEEE International Workshop on Factory Communication Systems (WFCS), pp. 1–4 (2018). https://doi.org/10.1109/WFCS.2018.8402377

15. Baskaran S, Niaki FA, Tomaszewski M, Gill JS, Chen Y, Jia Y, Mears L, Krovi V. Digital Human and Robot Simulation in Automotive Assembly using Siemens Process Simulate: A Feasibility Study. Proc Manuf. 2019;34:986–94. https://doi.org/10.1016/j.promfg.2019.06.097.

16. Kuehner KJ, Scheer R, Strassburger S. Digital twin: finding common ground—a meta-review. Proc CIRP. 2021;104:1227–32. https://doi.org/10.1016/j.procir.2021.11.206.

17. 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: 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), vol. 1, pp. 1789–1796 (2020). https://doi.org/10.1109/ETFA46521.2020.9212089

18. Kritzinger W, Karner M, Traar G, Henjes J, Sihn W. Digital Twin in manufacturing: a categorical literature review and classification. IFAC-PapersOnLine. 2018. https://doi.org/10.1016/j.ifacol.2018.08.474.

19. 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. https://doi.org/10.1016/j.cirpj.2020.02.002.

20. Barricelli BR, Casiraghi E, Fogli D. A survey on digital twin: definitions, characteristics, applications, and design implications. IEEE Access. 2019;7:167653–71. https://doi.org/10.1109/ACCESS.2019.2953499.

21. Van der Valk H, Haße H, Möller F, Arbter M, Henning J-L, Otto B A Taxonomy of Digital Twins. In: AMCIS (2020)

22. Jiang F, Ma L, Broyd T, Chen K. Digital twin and its implementations in the civil engineering sector. Aut Construct. 2021;130: 103838. https://doi.org/10.1016/J.AUTCON.2021.103838.

23. Semeraro C, Lezoche M, Panetto H, Dassisti M. Digital twin paradigm: a systematic literature review. Comput Ind. 2021;130: 103469. https://doi.org/10.1016/J.COMPIND.2021.103469.

24. Holler M, Uebernickel F, Brenner W Digital twin concepts in manufacturing industries-a literature review and avenues for further research. In: Proceedings of the 18th International Conference on Industrial Engineering (IJIE), pp. 1–9. Korean Institute of Industrial Engineers Seoul, Korea (2016)

25. Negri E, Fumagalli L, Macchi M. A Review of the Roles of Digital Twin in CPS-based Production Systems. Proc Manuf. 2017;11:939–48. https://doi.org/10.1016/j.promfg.2017.07.198.

26. Campos-Ferreira AE, Lozoya-Santos JdJ, Vargas-Martínez A, Mendoza R, Morales-Menéndez R Digital twin applications: A review. In: Memorias del Congreso Nacional de Control Automático, pp. 606–611. Asociación de México de Control Automático Puebla, Mexico (2019)

27. Cimino C, Negri E, Fumagalli L. Review of digital twin applications in manufacturing. Comput Ind. 2019;113: 103130. https://doi.org/10.1016/J.COMPIND.2019.103130.

28. Enders MR, Hoßbach N Dimensions of digital twin applications-a literature review. In: Twenty-fifth Americas Conference on Information Systems, pp. 2575–2584. Association for Information Systems (AIS), Cancun (2019)

29. Stark R, Damerau T Digital Twin. In: Chatti, S., Tolio, T. (eds.) CIRP Encyclopedia of Production Engineering, pp. 1–8. Springer, Berlin, Heidelberg (2019). https://doi.org/10.1007/978-3-642-35950-7_16870-1

30. Tao F, Zhang H, Liu A, Nee AYC. Digital Twin in Industry: State-of-the-Art. IEEE Trans Ind Informat. 2019;15(4):2405–15. https://doi.org/10.1109/TII.2018.2873186.

31. Errandonea I, Beltrán S, Arrizabalaga S. Digital Twin for maintenance: A literature review. Comput Ind. 2020;123: 103316. https://doi.org/10.1016/J.COMPIND.2020.103316.

32. Fuller A, Fan Z, Day C, Barlow C. Digital twin: enabling technologies, challenges and open research. IEEE Access. 2020;8:108952–71. https://doi.org/10.1109/ACCESS.2020.2998358.

33. Lu Y, Liu C, Wang KIK, Huang H, Xu X. Digital Twin-driven smart manufacturing: connotation, reference model, applications and research issues. Robot Comput-Integrated Manuf. 2020;61: 101837. https://doi.org/10.1016/J.RCIM.2019.101837.

34. Minerva R, Lee GM, Crespi N. Digital twin in the IoT context: a survey on technical features, scenarios, and architectural models. Proc IEEE. 2020;108(10):1785–824. https://doi.org/10.1109/JPROC.2020.2998530.

35. Ciano MP, Pozzi R, Rossi T, Strozzi F. Digital twin-enabled smart industrial systems: a bibliometric review. Int J Comput Integrated Manuf. 2021;34(7–8):690–708. https://doi.org/10.1080/0951192X.2020.1852600.

36. Lattanzi L, Raffaeli R, Peruzzini M, Pellicciari M. Digital twin for smart manufacturing: a review of concepts towards a practical industrial implementation. Int J Comput Integrated Manuf. 2021;34(6):567–97. https://doi.org/10.1080/0951192X.2021.1911003.

37. Liu M, Fang S, Dong H, Xu C. Review of digital twin about concepts, technologies, and industrial applications. J Manuf Syst. 2021;58:346–61. https://doi.org/10.1016/j.jmsy.2020.06.017.

38. Melesse TY, Di Pasquale V, Riemma S. Digital Twin models in industrial operations: State-of-the-art and future research directions. IET Collaborative Intell Manuf. 2021;3(1):37–47. https://doi.org/10.1049/cim2.12010.

39. Opoku DGJ, Perera S, Osei-Kyei R, Rashidi M. Digital twin application in the construction industry: a literature review. J Build Eng. 2021;40: 102726. https://doi.org/10.1016/J.JOBE.2021.102726.

40. Singh M, Fuenmayor E, Hinchy EP, Qiao Y, Murray N, Devine D Digital twin: Origin to future. Applied System Innovation **4**(2) (2021). https://doi.org/10.3390/asi4020036

41. VanDerHorn E, Mahadevan S. Digital Twin: Generalization, characterization and implementation. Decision Support Syst. 2021;145: 113524. https://doi.org/10.1016/J.DSS.2021.113524.

42. Bhandal R, Meriton R, Kavanagh RE, Brown A. The application of digital twin technology in operations and supply chain management: a bibliometric review. Supp Chain Manag. 2022;27(2):182–206. https://doi.org/10.1108/SCM-01-2021-0053.

43. Fang X, Wang H, Liu G, Tian X, Ding G, Zhang H. Industry application of digital twin: from concept to implementation. Int J Adv Manuf Technol. 2022;121(7):4289–312. https://doi.org/10.1007/s00170-022-09632-z.

44. Hu W, Lim KYH, Cai Y Digital Twin and industry 4.0 enablers in building and construction: a survey. Buildings 12(11), 2004 (2022). https://doi.org/10.3390/buildings12112004

45. Liu YK, Ong SK, Nee AYC. State-of-the-art survey on digital twin implementations. Adv Manuf. 2022;10(1):1–23. https://doi.org/10.1007/s40436-021-00375-w.

46. Sharma A, Kosasih E, Zhang J, Brintrup A, Calinescu A. Digital Twins: State of the art theory and practice, challenges, and open research questions. J Ind Inform Integration. 2022;30: 100383. https://doi.org/10.1016/j.jii.2022.100383.

47. Singh M, Srivastava R, Fuenmayor E, Kuts V, Qiao Y, Murray N, Devine D. Applications of Digital Twin across Industries: A Review. Appl Sci. 2022;12(11):5727. https://doi.org/10.3390/app12115727.

48. Son YH, Kim G-Y, Kim HC, Jun C, Noh SD. Past, present, and future research of digital twin for smart manufacturing. J Comput Design Eng. 2022;9(1):1–23. https://doi.org/10.1093/jcde/qwab067.

49. Tao F, Xiao B, Qi Q, Cheng J, Ji P. Digital twin modeling. J Manuf Syst. 2022;64:372–89. https://doi.org/10.1016/J.JMSY.2022.06.015.

50. Mazumder A, Sahed MF, Tasneem Z, Das P, Badal FR, Ali MF, Ahamed MH, Abhi SH, Sarker SK, Das SK, Hasan MM, Islam MM, Islam MR. Towards next generation digital twin in robotics: Trends, scopes, challenges, and future. Heliyon. 2023;9(2):13359. https://doi.org/10.1016/J.HELIYON.2023.E13359.

51. Perossa D, Santacruz RFB, Rocca R, Fumagalli L Digital Twin Application to Energy Consumption Management in Production: A Literature Review, pp. 96–105 (2023). https://doi.org/10.1007/978-3-031-25182-5_10

52. van der Valk H, Haße H, Möller F, Otto B. Archetypes of Digital Twins. Busin Inform Syst Eng. 2022;64(3):375–91. https://doi.org/10.1007/s12599-021-00727-7.

53. Haße H, van der Valk H, Möller F, Otto B. Design Principles for Shared Digital Twins in Distributed Systems. Busin Inform Syst Eng. 2022. https://doi.org/10.1007/s12599-022-00751-1.

54. Malakuti S, Schmitt J, Platenius-Mohr M, Grüner S, Gitzel R, Bihani P. A four-layer architecture pattern for constructing and managing digital twins. In: Bures T, Duchien L, Inverardi P, editors. Software Architecture. Cham: Springer; 2019. p. 231–46.

55. Redelinghuys AJH, Kruger K, Basson A A six-layer architecture for digital twins with aggregation. In: Studies in Computational Intelligence vol. 853, pp. 171–182. Springer (2020). https://doi.org/10.1007/978-3-030-27477-1_13

56. Agrawal A, Fischer M, Singh V Digital Twin: From Concept to Practice. arXiv, ??? (2022). https://doi.org/10.48550/ARXIV.2201.06912

57. Agrawal, A., Singh, V., Thiel, R., Pillsbury, M., Knoll, H., Puckett, J., Fischer, M.: Digital Twin in Practice: Emergent Insights from an Ethnographic-Action Research Study. In: Construction Research Congress 2022. Am Soc Civ Eng (2022). https://doi.org/10.1061/9780784483961.131

58. Golovatchev J Digitaler Zwilling für die Realisierung Utility 4.0. In: Doleski, O.D. (ed.) Realisierung Utility 4.0 Band 2: Praxis der Digitalen Energiewirtschaft Vom Vertrieb Bis zu Innovativen Energy Services, pp. 593–609. Springer, Wiesbaden (2020). https://doi.org/10.1007/978-3-658-25589-3_42

59. Gartner: Gartner Identifies Five Emerging Technology Trends That Will Blur the Lines between Human and Machine (2018). https://www.gartner.com/en/newsroom/press-releases/2018-08-20-gartner-identifies-five-emerging-technology-trends-that-will-blur-the-lines-between-human-and-machine

60. Bahrenburg, S., Munk, S., Fischer, J.W., Rützel, H., Fuchs, B.: Studie Future PLM: Product Life cycle Management in der digitalen Zukunft — Wegbereiter für IoT, Industrie 4.0 und Digital Twin (2019). https://www.bearingpoint.com/files/PLM_BearingPoint_Studie.pdf

61. Rüeger B, Hannich F, Hüttermann M, Fuchs R, Suvada A, Kübler DL, Barth L, Heierli R, Rozumowski A, Kiarostami T Swiss Marketing Leadership Studie 2018 : Geschwindigkeit vs. Beständigkeit. ZHAW Zürcher Hochschule für Angewandte Wissenschaften, Winterthur (2018). https://doi.org/10.21256/zhaw-2787

62. Nikolakis N, Alexopoulos K, Xanthakis E, Chryssolouris G. The digital twin implementation for linking the virtual representation of human-based production tasks to their physical counterpart in the factory-floor. Int J Comput Integrated Manuf. 2019;32(1):1–12. https://doi.org/10.1080/0951192X.2018.1529430.

63. Madni AM, Madni CC, Lucero SD Leveraging Digital Twin Technology in Model-Based Systems Engineering. Systems 7(1) (2019). https://doi.org/10.3390/systems7010007

64. Detzner A, Eigner M A digital twin for root cause analysis and product quality monitoring. In: DS 92: Proceedings of the DESIGN 2018 15th International Design Conference, pp. 1547–1558 (2018)

65. Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. J Manuf Syst. 2018. https://doi.org/10.1016/j.jmsy.2018.01.006.

66. Lutters E, Damgrave R. The development of Pilot Production Environments based on Digital Twins and Virtual Dashboards. Proc CIRP. 2019;84:94–9. https://doi.org/10.1016/j.procir.2019.04.228.

67. Ma X, Tao F, Zhang M, Wang T, Zuo Y. Digital twin enhanced human-machine interaction in product lifecycle. Proc CIRP. 2019;83:789–93. https://doi.org/10.1016/j.procir.2019.04.330.

68. Tao F, Zhang M, Nee AYC Digital Twin Driven Smart Manufacturing. Academic Press (2019)

69. Ala-Laurinaho R, Autiosalo J, Nikander A, Mattila J, Tammi K. Data link for the creation of digital twins. IEEE Access. 2020;8:228675–84. https://doi.org/10.1109/ACCESS.2020.3045856.

70. Brockhoff T, Heithoff M, Koren I, Michael J, Pfeiffer J, Rumpe B, Uysal MS, Van Der Aalst WMP, Wortmann A Process Prediction with Digital Twins. In: 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), pp. 182–187 (2021). https://doi.org/10.1109/MODELS-C53483.2021.00032

71. Dalibor, M., Michael, J., Rumpe, B., Varga, S., Wortmann, A.: Towards a Model-Driven Architecture for Interactive Digital Twin Cockpits. In: Gillian, D., Frank, U., Gerti, K., W., L.S., C, M.H. (eds.) Conceptual Modeling, pp. 377–387. Springer, Cham (2020)

72. Martínez GS, Sierla S, Karhela T, Vyatkin V Automatic Generation of a Simulation-Based Digital Twin of an Industrial Process Plant. In: IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, pp. 3084–3089 (2018). https://doi.org/10.1109/IECON.2018.8591464

73. Merkle L, Segura AS, Torben Grummel J, Lienkamp M Architecture of a Digital Twin for Enabling Digital Services for Battery Systems. In: 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS), pp. 155–160 (2019). https://doi.org/10.1109/ICPHYS.2019.8780347

74. Scheibmeir J, Malaiya Y An API Development Model for Digital Twins. In: 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 518–519 (2019). https://doi.org/10.1109/QRS-C.2019.00103

75. Grieves M, Vickers J Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: Transdisciplinary Perspectives on Complex Systems, pp. 85–113. Springer (2017)

76. Mussomeli A, Parrot A, Umbenhauer B, Warshaw L Digital Twins Bridging the physical and digital. Deloitte Insights (2020)

77. Aivaliotis P, Georgoulias K, Alexopoulos K Using digital twin for maintenance applications in manufacturing: State of the Art and

Gap analysis. In: 2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC), pp. 1–5 (2019). https://doi.org/10.1109/ICE.2019.8792613

78. Ashtari Talkhestani B, Jung T, Lindemann B, Sahlab N, Jazdi N, Schloegl W, Weyrich M An architecture of an Intelligent Digital Twin in a Cyber-Physical Production System. at - Automatisierungstechnik 67(9), 762–782 (2019). https://doi.org/10.1515/auto-2019-0039

79. El Saddik A. Digital twins: the convergence of multimedia technologies. IEEE MultiMedia. 2018;25(2):87–92. https://doi.org/10.1109/MMUL.2018.023121167.

80. Landahl J, Panarotto M, Johannesson H, Isaksson O, Lööf J Towards adopting digital twins to support design reuse during platform concept development. DS 91: Proceedings of NordDesign 2018, Linköping, Sweden, 14th-17th August 2018 (2018)

81. Meierhofer J, West S Service value creation using a digital twin. In: Naples Forum on Service, Service Dominant Logic, Network & Systems Theory and Service Science: Integrating Three Perspectives for a New Service Agenda, Ischia, pp. 4–7 (2019)

82. Qi Q, Tao F, Zuo Y, Zhao D Digital Twin Service towards Smart Manufacturing. In: Procedia CIRP (2018). https://doi.org/10.1016/j.procir.2018.03.103

83. Nakajima S Introduction to Total Productive Maintenance. Productivity Press (1988)

84. Leflar, J.: TPM at Hewlett-Packard. In: 10th Total Productive Maintenance Conference (1999)

85. Verband Deutscher Maschinen und Anlagenbau: VDMA 66412-1/A1:2010-05 (2010). https://www.beuth.de/de/technische-regel/vdma-66412-1-a1/131979365

86. Bange UK Overall Service Effectiveness – Eine Adaption des OEE–Modells auf die Digitalisierung von After–Sales/Service. (2019). https://www.linkedin.com/pulse/overall-service-effectiveness-eine-adaption-des-auf-udo-k-bange/

87. Jun H-B, Shin J-H, Kiritsis D, Xirouchakis P. System architecture for closed-loop PLM. Int J Comput Integrated Manuf. 2007;20(7):684–98. https://doi.org/10.1080/09511920701566624.

88. Stark J Product Lifecycle Management (Volume 1): 21st Century Paradigm for Product Realisation vol. 1, 3rd edn. Springer (2015)

89. Wellsandt S, Nabati E, Wuest T, Hribernik KA, Thoben KD. A survey of product lifecycle models: towards complex products and service offers. Int J Product Lifecycle Manag. 2016;9(4):353–90. https://doi.org/10.1504/IJPLM.2016.080985.

90. Boschert S, Heinrich C, Rosen R Next Generation Digital Twin. In: Tools and Methods of Competitive Engineering: Proceedings of the Twelfth International Symposium on Tools and Methods of Competitive Engineering (TMCE), pp. 209–218. Delft University of Technology, Las Palmas de Gran Canaria, Spain (2018)

91. Tao F, Zhang M. Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing. IEEE Access. 2017;5:20418–27. https://doi.org/10.1109/ACCESS.2017.2756069.

92. Abramovici M, Göbel JC, Dang HB. Semantic data management for the development and continuous reconfiguration of smart products and systems. CIRP Ann. 2016;65(1):185–8. https://doi.org/10.1016/j.cirp.2016.04.051.

93. Boschert S, Rosen R Digital Twin–The Simulation Aspect. In: Hehenberger, P., Bradley, D. (eds.) Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and Their Designers, pp. 59–74. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-32156-1_5

94. Tomiyama T, Lutters E, Stark R, Abramovici M. Development capabilities for smart products. CIRP Ann. 2019;68(2):727–50. https://doi.org/10.1016/j.cirp.2019.05.010.

95. Zheng P, Lin TJ, Chen CH, Xu X. A systematic design approach for service innovation of smart product-service systems. J Clean Prod. 2018. https://doi.org/10.1016/j.jclepro.2018.08.101.

96. ...Page MJ, McKenzie JE, Bossuyt PM, Boutron I, Hoffmann TC, Mulrow CD, Shamseer L, Tetzlaff JM, Akl EA, Brennan SE, Chou R, Glanville J, Grimshaw JM, Hróbjartsson A, Lalu MM, Li T, Loder EW, Mayo-Wilson E, McDonald S, McGuinness LA, Stewart LA, Thomas J, Tricco AC, Welch VA, Whiting P, Moher D. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. Syst Rev. 2021;10(1):89. https://doi.org/10.1186/s13643-021-01626-4.

97. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E. Scikit-learn: Machine Learning in Python. J Mach Learn Res. 2011;12:2825–30.

98. Goldberg Y, Levy O word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722 (2014)

99. St. Pierre EA, Jackson AY Qualitative Data Analysis After Coding vol. 20, pp. 715–719. Sage Publications Sage CA: Los Angeles, CA (2014)