A Cross-Domain Systematic Mapping Study on Software Engineering for Digital Twins

Manuela Dalibor, Nico Jansen, Bernhard Rumpe, David Schmalzing, Louis Wachtmeister, Manuel Wimmer, Andreas Wortmann

A Cross-Domain Systematic Mapping Study on Software Engineering for Digital Twins

Keywords: Software Engineering, Digital Twins, Manufacturing, Industry 4.0

Abstract

Digital Twins are currently investigated as the technological backbone for providing an enhanced understanding and management of existing systems as well as for designing new systems in various domains, e.g., ranging from single manufacturing components such as sensors to large-scale systems such as smart cities. Given the diverse application domains of Digital Twins, it is not surprising that the characterization of the term Digital Twin, as well as the needs for developing and operating Digital Twins are multi-faceted. Providing a better understanding what the commonalities and differences of Digital Twins in different contexts are, may allow to build reusable support for developing, running, and managing Digital Twins by providing dedicated concepts, techniques, and tool support. In this paper, we aim to uncover the nature of Digital Twins based on a systematic mapping study which is not limited to a particular application domain or technological space. We systematically retrieved a set of 1471 unique publications of which 356 were selected for further investigation. In particular, we analyzed the types of research and contributions made for Digital Twins, the expected properties Digital Twins have to fulfill, how Digital Twins are realized and operated, as well as how Digital Twins are finally evaluated. Based on this analysis, we also contribute a novel feature model for Digital Twins from a software engineering perspective as well as several observations to further guide future software engineering research in this area.

© 2022 Elsevier Inc. All rights reserved.

1. Introduction

Research and industry leverage Digital Twins to monitor and control (cyber–physical) systems in various domains, including autonomous driving (Chen et al., 2018), biology (Jordens and Jamshidi, 2018), medicine (Lauzeral et al., 2019), smart manufacturing (Um et al., 2018), and many more. They promise tremendous potential to reduce cost and time and improve our understanding of the represented systems. The various Digital Twins serve different purposes, including analysis (Pargmann et al., 2018), control (Verner et al., 2018), and behavior prediction (Knapp et al., 2017), and they are used at different times relative to the represented system, e.g., before it exists to explore its design space (Lutters, 2018) or during its runtime to optimize its behavior (Biesinger et al., 2018). Despite plethora of definitions (Eisenträger et al., 2018; Kostenko et al., 2018; Liu et al., 2019e; Peruzzini et al., 2020; Qi et al., 2018) there is little consensus about what a Digital Twin actually is. This also is reflected in many of the available definitions being (1) ambiguous, by deferring to another undefined term, such as a “virtual representation” (Ardanza et al., 2019), a “computable virtual abstraction” (Ullah, 2019), or a “virtual projection of the industrial facility into the cloud” (Yusupbekov et al., 2018); (2) narrow, by focusing on specific use cases, domains, or technologies, such as a “digital model of the real network environment” (Dong et al., 2019) or a “virtual representation based on AR-technology” (Pargmann et al., 2018); or (3) utopian, due to all-encompassing aspirations, such as an “integrated virtual model of a real-world system containing all of its physical information” (Park et al., 2019), a “complete digital representation” (Mandolla et al., 2019). Instead of producing more of such definitions, we aim to uncover the nature of Digital Twins as documented in literature bottom-up. To this end, we conducted
a systematic mapping study (Kitchenham et al., 2009; Petersen et al., 2008, 2015) on Digital Twins to investigate the following research questions (RQs):

- Who uses Digital Twins for which purposes (RQ-1)?
- What are the conceptual properties of Digital Twins (RQ-2)?
- How are Digital Twins engineered (RQ-3), deployed (RQ-4), and operated (RQ-5)?
- To which extend are Digital Twins evaluated (RQ-6)?

Following a detailed search strategy involving five digital libraries, we initially found 1472 unique publications. Out of these, 624 publications were identified as potentially relevant of which 356 publications were finally selected and categorized using a comprehensive classification scheme focusing on the contribution types, research types, properties, implementation methods, deployment details, operation decisions, and evaluation means of Digital Twins. The resulting research landscape developed by this study can help to understand, guide, and compare future research in this field across different domains. In particular, this paper identifies common and varying Digital Twin features and identifies challenges that seem to be less investigated. The contributions of this paper, hence, are

1. A first systematic and comprehensive study on Digital Twins across different domains, applied implementation technologies, and purposes.
2. A novel feature model of common Digital Twin features to guide researchers and practitioners in making decisions about the design, development, deployment, and operation of Digital Twins.
3. A synthesis of observations on the landscape of Digital Twin research that may guide further research.

In the following, Section 2 presents related studies and discusses how our survey differs in width, depth, and research method. Afterward, Section 3 details our research method, questions, queries, and data collection. Then, Section 4 answers these research questions individually and Section 5 investigates correlations between the answers to these questions. Based on our findings, Section 6 presents a novel model of common Digital Twin features. Section 7 discusses threats to the validity of our survey, and finally, Section 8 concludes the paper with an outlook on future work.

2. Related Studies

Research has produced a variety of studies on Digital Twins, their features, and their application. This section relates these studies to our investigation.

Studies on features and characteristics of digital twins

The survey presented in van der Valk et al. (2020) reduces an initial corpus of 579 publications to 233 included publications to identify a Digital Twin classification schema of eight features. The systematic survey on themes in Digital Twin research presented in Jones et al. (2020) investigates a corpus of 92 publications obtained by using Google Scholar as data source solely and the artificial cut-off criterion to consider the first 500 search results only. In this study, the authors identify 19 themes of Digital Twin research, such as managing a physical entity, fidelity of the Digital Twin, and the twinning process itself. Ultimately, the authors suggest a more detailed comparison of Digital Twins and publications from related fields. Through the study, the authors identify 13 key characteristics of Digital Twins, including the nature of the twinned system and its environment (both of which can be physical or virtual) and how its connections are realized (physical-to-virtual and vice versa), and similar. Hence, the study predisposes interesting assumptions about Digital Twins, such as that there is a connection between the twinned system and the Digital Twin. Another systematic literature study considers 82 publications out of an initial corpus of 1300 publications to identify features and characteristics of Digital Twins in the oil & gas industry (Wanasinhe et al., 2020). For the oil & gas industry, the authors of Wanasinhe et al. (2020) identify asset performance management, asset risks, and support for virtual training to navigate and operate assets as the most important drivers for Digital Twin research. These priorities reflect the complex and often remote nature of assets in the oil & gas industry, and we expect these not to apply to general Digital Twin research in the same order. Moreover, that survey identifies challenges to Digital Twin engineering and identify the lack of scope and focus, the lack of standardization, and security issues as most important in their domain.

In the context of Industry 4.0, various studies touch on the topic of Digital Twins. For instance, a strategic roadmap towards Industry 4.0 (Cherif et al., 2018) identifies Digital Twins as the goal of the smart manufacturing strategy for the transition to Industry 4.0. Here, Digital Twins are considered as a vision combining data analytics provided by intelligent enterprise resource planning (ERP) systems and data collected from the manufacturing machines. Achieving this vision is left as subject for future work.

Studies on engineering digital twins

Various studies address the question of how Digital Twins are engineered by investigating the requirements (Durão et al., 2018), architectures (Kritzinger et al., 2018; Olivotti et al., 2019), and the technologies (Minerva et al., 2020) used with Digital Twins. Some of these focus on a specific domain, such as the study of 43 publications on engineering Digital Twins in manufacturing presented in Kritzinger et al. (2018). In contrast, others have a wider scope, such as the survey of over 137 publications reported in Minerva et al. (2020) or the study of 52 publications on requirements for Digital Twins (Durão et al., 2018). In Kritzinger et al. (2018), the authors categorize a corpus of 43 publications on Digital Twins according to the type of contribution (case study, concept, definition, review), level of integration (which describes the nature of the contribution as being a Digital Twin, a Digital Shadow, or a model), application domain, and employed technologies (including AutomationML, simulation, SAP, RFID, etc.). Based on their data, the authors conclude that research on the Digital Twin is still “in its infancy”. Our study presented in this paper may allow to confirm or reject whether this still is the case.

In Minerva et al. (2020), the authors investigate many interesting research questions from the nature of Digital Twins to their essential features to potentials for the evolution of the Digital Twin idea. Their process for selecting the included 137 publications is not further discussed. In another survey on requirements for Digital Twins in the context of Industry 4.0 (Durão et al., 2018), the authors analyze 52 publications obtained via Web of Science and combine the insights from their corpus with interviews of six industry representatives. Overall, they identify real-time data handling, integration, and fidelity as the main requirements for Digital Twins in their context.

Similar studies focus on particular aspects of Digital Twin engineering, such as the relation of Digital Twins to product lifecycle management systems (Lim et al., 2019), the synchronization of Digital Twins with their counterparts (Modoni et al., 2019), or architectures for installed base management systems (Olivotti et al., 2019). The authors of Lim et al. (2019) investigate an initial corpus of 258 publications and reduce it to include
Studies on literature about digital twins

Moreover, a few meta-studies investigate, for instance, how Digital Twins are described in the literature (Barth et al., 2020), what the most frequently used terms for describing Digital Twin challenges in high-value manufacturing are (Singh et al., 2018), or which definitions are used to describe Digital Twins (Stark and Damerau, 2019). For instance, in Singh et al. (2018), the authors identify the 75 most often used terms to describe Digital Twins (the top 3 being “system”, “data”, and “physical”) and analyze which topics these belong to. Based on that analysis, the authors produce 11 clusters of Digital Twin topics (including engineering, standards, scalability, cost & time, cyber–physical system (CPS), data, user interaction, and more) and summarize the challenges in high-value manufacturing relating to these clusters. The study presented Digital Twins (Stark and Damerau, 2019), does not consider terminology used to describe Digital Twins but considers explicit definitions only. Based on an ad-hoc literature analysis, the authors collected 19 definitions and identify eight Digital Twin dimensions, including “connectivity mode”, “CPS intelligence”, “simulation capabilities”, and “human interaction”.

Synopsis

Most of the mentioned related studies have a particular scope and depth as well as a certain level of systematic rigor. Hence, they only address a subset of the research questions investigated within this study, consider smaller corpora, or cannot be fully reproduced with the information presented in the corresponding publications. For the latter, there is often a lack of information about the selected data sources, search query, and inclusion/exclusion criteria. Thus, a larger and detailed study on Digital Twin concerns across different domains in a fully reproducible manner is still missing, especially when it comes to the software engineering of Digital Twins.

3. Research Method

A systematic mapping study identifies publications within a research field and classifies these according to predefined, structured criteria (Petersen et al., 2008). Thus, it provides an overview of the topics and contribution types for a research domain to analyze the current status, challenges, and general progress. We have based our study on generally approved guidelines (Petersen et al., 2008; Kitchenham and Charters, 2007) and practices of other mapping studies (Wortmann et al., 2020; Budgen et al.; do Nascimento et al.; Kosar et al., 2016). To conduct this study, we have used the following five-step process (based on (Petersen et al., 2008)); (1) Definition of the research questions, (2) Search for primary publications, (3) Identification of inclusion and exclusion criteria and filtering of primary sources based on these, (4) Classification of primary studies by keywording, and (5) Extraction and aggregation of data.

Fig. 1 visualizes this process with its phases and their outcomes. In the first phase, the scope of the mapping study is defined. This includes the research questions as well as the overall topics of the publications to be considered. In the second phase, we collected the corpus of potentially relevant publications for our study. Afterward, in the third phase, we analyzed the corpus according to defined criteria and reduced it to conduct our study only on thematically relevant contributions. Based on abstract and keywords, we then derived an initial classification scheme (cf. Phase 4). Finally, the relevant publications were examined (based on full reads) and classified in Phase 5, mapping the identified classes to the number of findings and findings cross-related as well as mapped to software engineering phases to provide answers to the research questions described in the following.

Studies on the application of digital twins

Other studies investigate the application of Digital Twins. Some of these also are focused on specific domains, such as the study of 13 publications applying Digital Twins in construction (Kan and Anumba, 2019), the survey of 26 publications on the use of Digital Twins in manufacturing (Negri et al., 2017), or the survey about 23 publications on Digital Twins in smart interconnected factories (Papazoglou and Andreou, 2019). In the latter, the authors identify a “Digital Twin lifecycle” being one of the key enablers for smart manufacturing networks. Some studies with narrow focus include larger corpora, such as the survey about 110 publications on Digital Twins in smart manufacturing (Mehta et al., 2018). Some of these studies are less narrow, such as the investigation of 50 studies on Digital Twins in multiple industrial domains (Tao et al., 2019). The latter’s findings are that (asset) prognostics and health management is the most popular application area for industrial Digital Twins, that modeling is essential for engineering Digital Twins, and that main challenges in Digital Twin application is bridging the gap between cyber parts and physical parts. The survey presented in Aurosalo et al. (2020) investigates how Digital Twin implementations could be evaluated and presents a grading schema for Digital Twins based on an initial corpus of 16 publications.
3.1. Research questions

We aim to identify publications on Digital Twins to investigate how they are defined, how they are engineered and used, and to document the current state-of-the-art. Furthermore, we analyze the different goals that application domains pursue with Digital Twins concerning their real-world counterparts and overall lifetime. This general research interest results in the following research questions. However, not every paper is expected to provide an answer to every question. Therefore, some information on certain facets may not be available (N/A).

1. Who uses Digital Twins for which purposes?

These questions aim to identify which application domains research in Digital Twins targets and which concerns and challenges the publications on Digital Twins address.

RQ-1.1 Which domains employ Digital Twins? With this question we aim to understand where Digital Twins are meant to be employed. This might shed light onto domains that are either particularly interested in Digital Twins or particularly suited for their application.

RQ-1.2 What is the purpose of these Digital Twins? Digital Twins might be investigated for a variety of reasons. This question aims to identify these.

2. What are the conceptual properties of Digital Twins?

A central focus of our study lies on investigating the conceptual properties of Digital Twins and their fundamental concepts. We explore, which properties and parts are associated with the twin and determine the subjects that can be twinned. Moreover, we identify whether a Digital Twin is unique to its counterpart and how these entities communicate. The questions concerning this topic are:

RQ-2.1 What is the real-world counterpart (i.e., the observed entity)? This question aims to understand what is represented by Digital Twins.

RQ-2.2 How are multiple Digital Twins of the same observed entity supported? Research contributes different perspectives on supporting multiple Digital Twins for (aspects) of the same system. We aim to identify how the different perspectives are supported.

RQ-2.3 When is the Digital Twin used in the lifecycle of the observed entity? Digital Twins can be used before the twinned system exists, during its deployment, for its operations, or even after. We aim to find out how the different lifecycle phases are supported by Digital Twins.

RQ-2.4 What stage of the observed entities lifecycle use does it represent? Orthogonal to RQ-2.3, Digital Twins can represent different lifecycle stages of the twinned system, e.g., during system runtime a Digital Twin of the system as-designed might be employed as no other, more up-to-date twin, is available.

RQ-2.5 How does the Digital Twin interact with its real-world counterpart? Some schools of Digital Twin thought propose that a software system can be a Digital Twin only if there is a direct interaction from it to the twinned system. With this question, we aim to find out whether this is a common perspective.

RQ-2.6 What (if anything) does the Digital Twin optimize? A Digital Twin might optimize the behavior of the twinned system, itself, or nothing at all, e.g., if it is only monitoring the twinned system. We aim to find out which optimizations are supported by Digital Twins.

RQ-2.7 What does a Digital Twin consist of? There is a scientific debate whether a Digital Twin is a model, a software system, or whether it even includes physical parts (such as hardware for augmented reality). With this question, we aim to find out what are common parts of Digital Twins.

RQ-2.8 Are Digital Twins decomposable? (De)composition is a quintessential software engineering for supporting reuse. We aim to find out whether research on Digital Twins supports it as well.

3. How are Digital Twins engineered?

These questions aim to identify the means to construct Digital Twins. To this end, it explores how the different parts and properties are realized for implementing these twins. We focus on technical details such as concrete realization, communication, or associations with product lifecycle management. Furthermore, we investigate different tools and technologies that have proven to be promising or essential for constructing Digital Twins. Corresponding related research questions are:

RQ-3.1 How are Digital Twins implemented? There might be different possibilities for realizing Digital Twins. This question aims at exploring these.

RQ-3.2 Which tools are used to engineer Digital Twins? In addition to the method of implementation, we are also interested in which tools are used during development. With this question, we aim to investigate whether there are certain trends concerning the tools used for the Digital Twin implementation.

RQ-3.3 Are Digital Twins developed with their own development process or are they developed together with the observed entity? Since the term twin already suggests a strong similarity to the observed system, we leverage this question to further investigate how this similarity affects the development process of Digital Twins.

RQ-3.4 How is quality assurance for the Digital Twin supported? With this question, we want to explore whether Digital Twins use the same or different methods for quality assurance than the observed system.
RQ-3.5 Has the Digital Twin own requirements? Like most systems developed using engineering methods, a Digital Twin might have requirements to fulfill. With this question, we intend to investigate whether and to which extent such requirements are already considered during the development of Digital Twins.

4. How are Digital Twins deployed?
After their construction, Digital Twins must be deployed appropriately. This research question investigates the initial configuration and system environment of twins. Furthermore, we consider concrete technologies for interconnectivity, resulting in the following research questions:

RQ-4.1 Where is the Digital Twin deployed? This question aims at uncovering whether Digital Twins operate in the cloud, on the edge, directly on the twinned system, or somewhere else.

RQ-4.2 How are Digital Twins connected to the observed entity? Digital Twins can be connected to their counterpart to exchange information. With this question, we want to find out which technologies are used to connect them.

5. How do Digital Twins operate?
Finally, we analyze the operation of Digital Twins, including in- and output, as well as underlying data structures. Furthermore, we investigate the possibilities of current Digital Twins to autonomously perform decision-making. Hence we investigate the following research questions:

RQ-5.1 Does the Digital Twin feature decision-making functions? A probable use case for Digital Twins can focus on its application to make decisions for a system. With this question, we aim to identify the different approaches to realize these artificial decision-making processes.

RQ-5.2 To which events, inputs, or data does a Digital Twin react to? Digital Twins usually rely on information about the physical entity’s state and user inputs. With this question, we identify how the Digital Twin gains required information and which events trigger its actions.

RQ-5.3 Which output does it produce? A frequent use case of Digital Twins is representing the physical entity’s state. With this question, we aim to find out how Digital Twins interact with their environment and how they communicate to and influence their operating context.

6. How are Digital Twins evaluated?
We analyze how the included publications evaluated their contributions. For quantification, we identify and assign classes of the different technology readiness levels. Furthermore, we explore whether the publications provide any metrics related to the proposed Digital Twins that could be reused in future research.

RQ-6.1 Which technology readiness levels do Digital Twin evaluations employ? When constructing Digital Twins, there can be a vast range between how the results are evaluated and to which extend they are ready for application in an industrial context. Thus, this question aims at classifying how mature the proposed twins are.

RQ-6.2 Does the Digital Twin yield any measurable advantages? As their deployment comes with a specific goal of what Digital Twins can achieve or improve, this question investigates potential benefits.

3.2. Search queries and data sources

The search strategy (see Fig. 2) is of major importance for the identification of relevant publications to answer our research questions. To this end, formulating an appropriate search query and selecting the relevant libraries is required. As we aim to find out who uses Digital Twins Digital Twins independently of a concrete domain or application context, we do not restrict our search term any further. Therefore, we ultimately searched in the selected databases for “Digital Twin”, keeping the search query simple and pragmatic. The selected databases are ACM Digital Library, IEEE Xplore, Scopus, SpringerLink, Web of Science. We opted to omit Google Scholar due to its problems with structured literature retrieval (Boeker et al., 2013) and to ensure quality of included sources.

As we conducted a full-text search for “Digital Twin”, we omitted using other related terms, such as “digital thread” or “digital shadow” as we expect publications contributing to Digital Twin research should at least use this term in either related work or referenced literature. However, we cannot guarantee to not miss a small amount of relevant publications, but argue that searching this way seems more appropriate than just searching in titles and abstracts for keywords. Moreover, we also did not put any lower bound as year limit and included papers published until October 2019. We extracted the results as comma-separated lists and manually merged these into a single list of unique publications.

3.3. Screening publications

The inclusion of a study into the classification phase of a systematic mapping study usually is decided based on its quality and accessibility as well as on its title, abstract, and keywords (Petersen et al., 2008). To reduce the corpus and enable reproduction of the study, we used the following explicit inclusion and exclusion criteria.

Inclusion criteria. From the initial corpus we identified the potentially relevant publications based on the following four criteria:

1. Studies published in peer-reviewed sources namely journals, conferences, and workshops.
2. Studies are electronically accessible.
3. Studies are available in English.
4. Studies where we could deduce from title, abstract, or keywords that their main topic of study is the conception or application of Digital Twins.

Exclusion criteria. Publications fulfilling the inclusion criteria were still excluded based on the following four criteria:

1. Studies from sources without systematic peer-review processes, such as books, magazines, and websites.
2. Short papers of less than 5 pages excluding references, such as editorials, reviews, or tool demonstration teasers.
3. Studies where Digital Twins are related work, further applications, or a broader context only.
4. Studies presenting literature reviews on Digital Twins (which are already discussed in Section 2).

While we did not limit the search to any time frame, the final corpus considered relevant does not include any papers from before the year 2011. This is due to the term only then gaining popularity and contributions not meeting our inclusion criteria.

We each analyzed and classified the first 30 (about 2%) publications of the 1472 unique publications of the corpus to build a shared understanding of Digital Twins, the research questions, and the classification scheme. We then discussed the analysis results to align our understanding of the publications, our analyses,
and the research methodology. To filter publications based on unambiguous exclusion criteria, we evenly distributed the remaining 1442 publications. Afterward, we determined inclusion based on whether a publication’s main contribution is towards Digital Twin research by screening titles, abstracts, and keywords only. We delayed the inclusion decision to the classification phase for publications where abstract and title screening did not suffice to determine inclusion. In this classification phase, we then decided the inclusion based on the publications’ full text to not exclude relevant publications with sub-optimal phrasing of abstract or title.

Eliminating 530 duplicates and 848 publications outside the scope of our study left 624 publications for review. These publications were again distributed between the authors of this paper for a detailed review and classification. Furthermore, we discussed the classification, exclusion, and inclusion of publications to align and refine our understanding whenever needed. During these discussions, we excluded additional publications and refined our shared understanding of the classification scheme. During the reviews and discussion, further unrelated publications were excluded. However, we did not exclude publications based on their venue or comprehensibility alone, and we also did not perform any additional quality evaluations.

3.4. Classification schema

To investigate Digital Twins appropriately, we have developed a corresponding classification scheme. This scheme is inspired by Petersen et al. (2008) and adapted for the landscape of Digital Twin research. The specific facets are based on our research on digital twins (e.g., Bibow et al., 2020; Bolender et al., 2021; Lehner et al., 2021) and have been revised and/or refined iteratively while discussing the papers among the authors as well as with digital twin experts of the “Internet of Production”2 excellence cluster and the “Christian Doppler Laboratory for Model-Integrated Smart Production”3.

After the initial screening, we analyzed the remaining 624 potentially relevant papers in the classification phase. We have read the remaining papers completely to extract all relevant information and excluded publications that turned out to be irrelevant for Digital Twins. We categorized the 356 remaining papers as follows.

**Contribution Type Facet.** Distinct papers may include different facets of contribution. Thus, inspired by Petersen et al. (2008), we classified the publications for the type of contribution. By this means, we used five contribution types (Petersen et al., 2008) to examine the overall kind of benefit the analyzed papers provide:

- **Analyses:** Papers presenting investigations without constructive contributions, e.g., (Song and Jang, 2018; Bekker, 2018; Schluse and Rossmann, 2016).
- **Concepts:** Papers presenting ways of reasoning about things, such as new metamodels or taxonomies, e.g., (Biesinger et al., 2018; Khakimov and Shcherbo, 2018; Vatn, 2018).
- **Methods:** Papers presenting ways of doing things, e.g., (Liu et al., 2019a; Sun et al., 2020).
- **Metrics:** Papers presenting ways of measuring things, e.g., (Allemang et al., 2014; Worden et al., 2020; Mavris et al., 2018).
- **Tools:** Papers presenting novel software tools related to implementing Digital Twins, e.g., (Mukherjee and DebRoy, 2019; Dröder et al., 2018; Dong et al., 2019).

We classified each publication uniquely to a contribution type.

**Research Type Facet.** A further important question relates to the research type of elaboration. It describes how the findings are conducted and presented. Again inspired by Petersen et al. (2008), we further distinguished the publications by their research type. We adjusted the originating classes to better match our corpus, e.g., by excluding philosophical facets, as these did not occur in our study. The five resulting research types based on (Petersen et al., 2008) are:

- **Evaluation:** Papers evaluating existing techniques, cf. e.g., (Damjanovic-Behrendt and Behrendt, 2019; Armendia et al., 2019; Fei et al., 2018).
- **Experience:** Papers reporting personal experiences, e.g., (Dröder et al., 2018; Gockel et al., 2012; Weiss et al., 2017).
- **Solution:** A novel solution is presented and argued for with case studies, e.g., Biesinger et al. (2018, 2019).
- **Validation:** Papers presenting novel techniques and experimenting with them, e.g., Wantia and Roßmann (2017), Jacob et al. (2019), Gomez-Escalonilla et al. (2020).
- **Vision:** Research agendas, e.g., a vision of model-based logistics engineering presented in Jain et al. (2020), Damjanovic-Behrendt (2018).

These five facets provide an overview of the research focus of the analyzed papers. The classification was disjoint, and we discussed contributions when in doubt.

**RQ-1.1 - Digital Twin Application Domain Facet.** When considering the application domains of Digital Twins, smart manufacturing often comes to mind first. To better understand why this is the case and which domains employ Digital Twins, we investigate RQ-1.1. To classify the different domains, we employed the Level 1 classes of the Statistical Classification of Economic Activities in the European Community (Anon, 2006), which comprise all economic areas currently considered by the European Parliament. Consequently, the application domain facet of our survey comprises all 20 Level 1 classes specified below:
A Digital Twin is little without its physical counterpart. Concerning RQ-1.2, we aimed to understand these purposes and therefore differentiated between the following dimensions:

- **CPS Data Processing, Integration, Persistence** summarizes purposes related to data processing, integration, and persistence checking, such as knowledge collection (Padovano et al., 2018), privacy enhancement (Damjanovic-Behrendt, 2018), or data integration into a shop floor environment (Urbina Coronado et al., 2018).
- **CPS Maintenance** subsumes purposes related with maintaining a CPS, such as predictive maintenance (Zaccaria et al., 2018), fatigue testing (Gomez-Escalonilla et al., 2020), or damage evaluation (Utzig et al., 2019).
- **CPS Monitoring** describes purposes related to collecting, analyzing, and visualizing data about the state of a Cyber-Physical System (CPS), such as real time monitoring of building operation efficiency (Carbonari et al., 2020), health monitoring (Zakrzyszek and Mall, 2017), or process parameter monitoring (Desai et al., 2020).
- **CPS Behavior Prediction** summarizes purposes to predict future CPS behavior, such as fuel consumption prediction (Uzun et al., 2019), driver behavior prediction for crash analysis (Chen et al., 2018), or predict pulsation and velocity inside the vessel of a human heart (Naplekov et al., 2018).
- **CPS Behavior Optimization** subsumes purposes related to optimizing a CPS's behavior, such as path planning for robots (Broder et al., 2018), running mode optimization of CNC tools (Luo et al., 2019), or reduce fatigue damage (Schirmann et al., 2018).
- **CPS Validation** describes purposes related with verification and validation activities, such as structural integrity analysis (Sharma et al., 2018), damage modeling for automotive low-carbon structural steel validation (Shcherba et al., 2018), or robot algorithm validation (Grinshpun et al., 2016).
- **CPS Reuse** describes purposes related to CPS reuse, such as design reuse (Landahl et al., 2018), or reconditioning (Ayani et al., 2018).
- **Design Space Exploration** subsumes purposes in the context of design space exploration, e.g., variation analysis (Wang et al., 2018) or virtual prototyping (Poppe et al., 2019).
- **Enterprise decision making** summarizes purposes that evolve around complex enterprise processes and decision making, such as macro perspective analysis (Block and Kuhlenkotter, 2019), or smart process planning (Liu et al., 2019).
- **Teaching** describes the purposes related to initial and continuing education, such as manufacturing machine exploration (David et al., 2018), teaching of manufacturing cell handling (Gordon et al., 2018), or robot manipulation training (Verner et al., 2019a).
- **Visualization & Representation** summarizes purposes directly related to visualizing a physical counterpart, such as represent a production site in a virtual environment (Ellgass et al., 2018), or visualizing object properties in augmented/mixed reality (Peukhurinen and Mikkonen, 2018).

Since Digital Twins might have more than one purpose, e.g., the health monitoring the approach presented in Zakrzyszek and Mall (2017) relates to monitoring as well as maintenance, the selection mentioned above is not disjoint.

**RQ-2.1 — Counterpart Facet.** A Digital Twin is little without its counterpart. To better understand what it is that is “twinned”, we classify our corpus’s publications according to the various counterparts described. Our classification schema of Digital Twin counterparts comprises:

- **Biological Beings**, such as factory employees (Graessler and Poehler, 2018), fishes (Joordens and Jamshidi, 2018), or sports players (Balachandar and Chinnaiyan, 2019).
- **Individual Systems**, such as automated cars (Atorf and Roffmann, 2018), gas turbines (Dawes et al., 2019), or manufacturing machines (Debroy et al., 2017).
- **Processes**, e.g., business processes (Rambow-Hoeschele et al., 2018), medical processes (Kakra et al., 2018), or recycling processes (Popa et al., 2018).
- **Products**, such as reinforced plastics (Wang et al., 2018), sunroof ring frames (Wärnfjord et al., 2017), or wearable masks (Zheng et al., 2018).
- **Systems of Systems**, such as complete factories (Biesinger et al., 2018), oil wells (Kosenkov et al., 2018), or railway systems (Vatn, 2018).
- **Other counterparts**, e.g., arbitrary physical bodies (El Saddik, 2018) or unspecified manufacturing resources (Lu and Xu, 2018b).

To distinguish whether a publication reports on an individual system, product, or system of systems, we discussed these publications and together decided about their specific focus regarding the Digital Twins’ counterpart(s). We also encountered some publications that report on Digital Twins for more than one counterpart or a combination of counterparts, such as processes and related systems of systems (He et al., 2019) or products and related systems (Lechler et al., 2019). Such contributions add to multiple counterpart facets accordingly.

**RQ-2.3 — Digital Twin Lifetime Facet.** Digital Twins can reflect, monitor, and support all phases of the physical entity. In some application scenarios building the physical entity may be very time-
or cost-intensive. In these cases, a Digital Twin can be applied during the design phase of the physical entity to communicate design decisions or to simulate multiple designs. At runtime, a Digital Twin may monitor the physical entity's actions and suggest further steps, e.g., for minimizing raw material waste or energy consumption. Consequently, we introduce a Digital Twin lifetime facet that distinguishes:

- **Design-time**, to characterize Digital Twins that are employed during the design phase of the physical entity, e.g., to evaluate different product variants (Atof and Rosßmann, 2018).
- **Runtime** to characterize Digital Twins that are employed while the physical entity is already operating. These Digital Twins may predict future behavior or control and optimize the physical entities' next steps (Sun et al., 2020).

Whether a publication describes a Digital Twin that is used at design- or runtime was usually explicit. In cases where a reviewer could not classify the described Digital Twin, we also discussed the respective paper among the authors. We also encountered publications that report on Digital Twins for more than one lifetime or even Digital Twins that were transferred from design- to runtime (Xu et al., 2019). Such contributions add to multiple lifetime facets accordingly. In contrast to RQ-2.3 this question focuses on the lifetime stage the Digital Twin represents and on the time when the Digital Twin is used.

**RQ-2.4 — Digital Twin Lifecycle Facet.** Digital Twins application scenarios can be distinguished along the system's lifecycle specified in the ISO/IEC 15288 system lifecycle (International Organization for Standardization, 2015). This norm distinguishes roughly three product lifecycle phases: the design phase including the conceptualization and modeling of the system, the manufacturing phase where the system is brought into existence, and the operation phase where the system operates and fulfills its intended purpose (Lu et al., 2015).

Depending on the twinned system's lifecycle phase, Digital Twins serve different purposes (Tao et al., 2018b). For instance, Digital Twins might represent a system as it is designed for design-space exploration of that system before it is constructed or Digital Twins focus on the running system as it is in use to represent its current state and serve for the maintenance prediction. Consequently, we introduce a Digital Twin lifecycle facet that distinguishes:

- **As-Designed**, to describe Digital Twins that represent the physical counterpart during its design phase. These Digital Twins are e.g., useful for optimizing the production process (Zhang et al., 2017a).
- **As-Manufactured** also integrates data that characterizes the manufacturing process of the physical counterpart. Thus, it may include runtime data that provides insights for maintenance (Gruender, 2017) of predicting material fatigue (Wanger et al., 2019).
- **As-Operated** describes Digital Twins that represent the usage and operation of the physical counterparts, e.g., for supervising and optimizing (Tan et al., 2019; Yusupbekov et al., 2018) or for predicting future behavior (Okita et al., 2019; Kumar et al., 2018).

To distinguish whether a publication describes a Digital Twin as designed, manufactured, or used, we discussed the categories a-priori in detail. If case a reader could not classify the described Digital Twin we also discussed the respective paper among the authors. We also encountered publications that report on Digital Twins for more than one lifecycle (Halena et al., 2019; Rios et al., 2016). Such publications contribute to multiple lifecycle facets accordingly. In contrast to RQ-2.3 this question focuses on the lifecycle stage the Digital Twin represents and on the time when the Digital Twin is used.

**RQ-2.5 — Interaction Facet.** Literature exhibits various understandings of Digital Twins from precise models used at system-design time (Jain et al., 2020; Zambal et al., 2018; Zhang et al., 2017a) that are used to prescribe a system to be to software systems twinning another system and directly manipulating its behavior (Schlußle et al., 2018; Graessler and Poehler, 2017; Qi et al., 2018). With this facet, we, thus, investigate whether Digital Twins tend to support direct interaction with the observed system. To this end, we distinguish two cases:

- **Direct Interaction** comprises Digital Twins that are directly connected to their counterpart through various communication measures, such as message buses, networks, or Internet technology.
- **No Direct Interaction** describes Digital Twins in which interaction either is indirect, e.g., by informing a human operator to execute system manipulation or there is no interaction at all, such as where the Digital Twin is interpreted as a dataset recorded from the twinned system.

Each publication contributes to exactly one of these classes.

**RQ-2.6 — Optimization Facet.** Many Digital Twins seem to optimize either themselves, their real-world counterpart, or both. However, not all Digital Twins strive to optimize. With this facet, we aim to investigate whether the optimization is considered in the development of Digital Twins and whether the Digital Twin optimizes itself, or its real-world counterpart. In our classification schema we, therefore, distinguished as follows:

- **Digital Twin Optimization** incorporates Digital Twins which optimize themselves without aiming to influence their twinned entity directly with this optimization. In Tygesen et al. (2018) for example, the Digital Twin collects data from a health monitoring system to optimize its structural health model of the real-world counterpart.
- **Counterpart Optimization** subsumes Digital Twins aiming to only optimizing their real-world counterpart. For instance, in Guerra et al. (2019) a Digital Twin is initialized with real tech parameters and then used to optimize the real-world counterpart, without updating the Digital Twins simulation model.
- **Digital Twin & Counterpart Optimization** subsumes Digital Twins that not only optimize their counterpart alone, but also use information from their counterpart to optimize themselves. In Gonzalez et al. (2018), for example, the Digital Twin is used for state estimation in non-linear electromechanical systems and optimizes not only the electromechanical system but also the Digital Twin itself.

Of course, we also encountered multiple publications where counterpart or self-optimization was explicitly not the purpose of the Digital Twin concept, as they focused on visualization (Blaga and Tamas, 2018) or monitoring (Eyre et al., 2018) alone. Moreover, some authors decided not to mention the possibility that the Digital Twin performs such optimization at all as, e.g., in Morais et al. (2018). However, as we cannot differentiate between publications where optimization is thought of as irrelevant for the purpose of Digital Twins and publications where the optimization was just not relevant for the published aspect, we decided to subsume these papers in an additional category.

**RQ-2.7 — Digital Twin Parts.** A Digital Twin usually is a logical unit, which is composed of different parts. For example, we can
distinguish between data, services, a virtual models and physical entities (Qi et al., 2021b). To understand how Digital Twins are developed and which components are necessary for software to become a Digital Twin, we collected information about Digital Twin parts.

- **Data** describes live data about the physical entity (Biesinger et al., 2019), historical data about the physical entity (Lauzeral et al., 2019), or data from other data sources that provide contextual information about the application scenario of the physical entity (Uzun et al., 2019).
- **Hardware Components** captures Digital Twins, which also contain physical components, such as equipment (Burrafato et al., 2019).
- **Models** describes software artifacts that are classified as models according to Stachowiak (Stachowiak, 1973), i.e., they have a purpose, perform abstraction and have a physical entity. Frequent examples were simulations (Kubota et al., 2018) and CAD models (Gregorio et al., 2019). We further classified models according to the aspects they describe. Thus we identified
  - structure of the physical twin, e.g., inner components,
  - behavior of the physical twin, e.g., interaction with its environment,
  - appearance of the physical twin, e.g., material information, and dimensions,
  - constraints of the physical twin, e.g., physical laws.
- **Software Components** characterizes custom (Rauch and Pietrzyk, 2019) and external (Urbina Coronado et al., 2018) software services that are described as part of the Digital Twin and cannot be classified as models.

Where publications ambiguously define which components they consider as part of the Digital Twin, we decided in favor of including these components as parts of the Digital Twin. In cases where no categorization was possible, we discussed the papers between us until we could reach an agreement. Many Digital Twins consist of multiple components, which can be assigned to different facets. These publications contribute to several facets.

**RQ-3.1 — Implementation Technique Facet** Digital twins are created using various implementation techniques depending on their purpose, lifetime, and more. With **RQ-3.1**, we analyze the different facets in which they are realized, including various models, programming languages, or simulations. Thus, we identified the following classes.

- **CAD/3D Models** describing the geometric representation of a physical component (Biesinger et al., 2018).
- **Data and Databases** covering collecting and analyzing operation data (Liu et al., 2018b) as well as different data formats (Moreno et al., 2017).
- **General Purpose Languages**, such as Java (Leng et al., 2019), C++ (Song and Jang, 2018), and Matlab (Saini et al., 2018).
- **Mathematical/Physical Models**, such as finite element (Wang et al., 2019b) and multi-physics models (Seshadri and Krishnamurthy, 2017).
- **Model-Driven Engineering** (MDE), such as UML or SysML models (Delbrügger and Rossmann, 2019), language workbenches (Oquendo, 2019), and AutomationML (Schroeder et al., 2016).
- **Simulation and Analysis**, such as Simulink (Raineri et al., 2018), Verosim (Grinshpun et al., 2016), or AnyLogic (Damiani et al., 2018).

The selection is not disjoint as contributions may use particular techniques as a foundation for implementing Digital Twins. For instance, Wang et al. (2019b) combines geometric data with physical models, which contribute to the construction of the system.

**RQ-3.2 — Tooling Facet** Engineering Digital Twins of different counterparts and for different purposes efficiently demands corresponding tool support. With this facet, we aim to uncover which kind of tools are used in the development and operations of Digital Twins. Our classification schema regarding tools applied to the engineering or operations of Digital Twins comprise tools that were mentioned 7 or more times by the publications of our corpus:

- **Artificial Intelligence Software**, such as Apache MXNet (Uzun et al., 2019), the IBM Watson software development kit (Dingli and Haddad, 2019), or TensorFlow (Um et al., 2018).
- **Communication Software**, including ROS (Ponnamarev et al., 2018), OPC UA (Ayani et al., 2018).
- **Computer-Aided Design** (CAD) and 3D modeling, such as SolidWorks (Ellgass et al., 2018), Siemens NX (Anand et al., 2018), or Autodesk Revit (Kaewunruen and Xu, 2018).
- **MDD Software**, such as AutomationML (Bao et al., 2019), Modelica (Malozemov et al., 2018), or SysML (Schluse et al., 2018).
- **Data Management Software**, such as Apache SOLR (Longo et al., 2019), SQL databases (Carbonari et al., 2020), or SAP HANA (Pargmann et al., 2018).
- **Process Management Software**, including ChemSiemens10 Tecnomatix (Caputo et al., 2019), UniSim Design (Yusupbekov et al., 2018), or in-house developed solutions (Baruffaldi et al., 2019).
- **Product Lifecycle Management Software**, such as Siemens PLM (Anand et al., 2018).
- **Programming Languages**, including Python (Karanjkar et al., 2018), Java (Leng et al., 2019), and others.
- **Simulation Software**, such as Abaqus (Denos et al., 2017), Gazebo (Mejia et al., 2017), the MAYA simulation framework (Ciavotta et al., 2017), or Verosim (Wantia and Roßmann, 2017).
- **Visualization Software**, such as Unity (Chen et al., 2018), OpenCV (Chakshu et al., 2019), or APIs for augmented reality devices (Utzig et al., 2019).
- **Other Software**, including various programming languages (Kloibhofer et al., 2018), specific self-developed toolsets (Konstantinov et al., 2017), APIs for communication (Kubota et al., 2018), or website development tools (Radchenko et al., 2018).

Some contributions use the same software, such as MDE software or the various programming languages for multiple purposes and employ a wide variety of software to engineer or operate Digital Twins. Consequently, our classification schema for the tooling facet allows for more entries than the number of publications included in the corpus.

**RQ-3.3 — Digital Twin Engineering Process Facet** Digital Twins can be developed before the twinned system, together with it, or after it. Developing the Digital Twin before the twinned system can facilitate design space exploration by frontloading of the twinned system as the Digital Twin might be used as substitute to explore properties of the twinned system at higher levels of abstraction. Developing the Digital Twin together with the observed system enables optimizing their interaction by, e.g., joint design-space exploration of both, the Digital Twin and the twinned system. Developing the Digital Twin after the twinned
system enables adding advanced functionality to existing systems and makes these accessible for analyses typically related to Digital Twins, such as behavior prediction. To better understand whether Digital Twins are generally developed together with their counterparts or detached from them, we have grouped the publications of our study accordingly. Our classification scheme of the development process of Digital Twins, thus, includes the following categories:

- **Joint Engineering** incorporates publications where the engineering and the evolution of Digital Twins are intertwined with the engineering of their counterparts.
- **Subsequent Engineering** incorporates publications where the engineering of the Digital Twin i.e., the process itself, succeeds the engineering of their counterpart. Here, the counterpart to be twinned, or previous versions of it, already exists, and that this information about the counterpart can be leveraged for its twinning.
- **Explorative Engineering** means that the development of the Digital Twin frontloads the development of their counterpart. Here, Digital Twins are developed from scratch without including information about their existing counterpart. Instead, they can be used to explore the properties of their counterpart.

We encountered some publications reporting that Digital Twins can be engineered in both fashions, either in a joint process together with their counterpart or in their own process independent of the engineering process of their counterpart. Of those publications that reported that a Digital Twin has its own engineering process, only some clarified if the Digital Twin was to be developed before or after the system. If a publication did not report on the engineering process of Digital Twins, or if that information was not derivable from the purpose of Digital Twins, then we regarded corresponding information as not available.

**RQ-3.4 — Quality Assurance Facet.** The Digital Twin can be understood as a precise design-time model, e.g., used for design space exploration of the system under development, prediction of its future behavior, and general frontloading, or as a software system observing another system at the other system’s runtime. Consequently, different means of quality assurance need to be employed to produce, operate, and maintain a high-quality Digital Twin. Concerning RQ-3.4, we aim to understand the state of quality assurance for Digital Twins. Our classification schema of Digital Twin quality assurance comprises the following dimensions:

- **Consistency monitoring**, e.g., by monitoring the differences between Digital Twin predictions and data obtained from the twinned system at its runtime (Yan et al., 2018; Song and Jang, 2018).
- **Simulation**, used at system design-time (Jain et al., 2020; Liu et al., 2019c).
- **Testing** other than simulation (Constantinescu et al., 2018), also employed at design time.
- **Other verification**, such as model-checking, also applied at design time (Bakliwal et al., 2018; Lohtander et al., 2018a).

As the boundaries between simulation, testing, and other verification are not used strictly through our corpus, we followed the terminology employed by the respective authors.

**RQ-3.5 — Requirements Facet.** Like any engineered system, Digital Twins are likely to have certain requirements to meet. So we addressed the question of whether Digital Twins have requirements of their own and considered the requirements that are typically placed on Digital Twins. To this end, we identified the following facets based on the publications in our corpus:

- **Real-time capability** requires from the Digital Twin that it provides its services or responses within a specified time constraint as e.g., required in Jain et al. (2020), Pargmann et al. (2018).
- **Digital Twin reaction matches real-world behavior** subsumes various verification and validation requirements aiming to ensure that the behavior of the observed system meets the reaction of the Digital Twin (Landahl et al., 2018).
- **Reusability** requires the Digital Twin to be reusable in different contexts or for closely related systems (Landahl et al., 2018; Martin et al., 2019).

As no other facet was mentioned more than once in the publications of our corpus, we decided not to further investigate these possible facets.

**RQ-4.1 Digital Twin Host Facet.** A Digital Twin may operate in some context, such as in some cloud, on an edge device, or directly on the twinned system. To understand where Digital Twins operate we classify the publications in our corpus according to the various hosts described:

- **Cloud** incorporates publications where the Digital Twin is deployed in some cloud, either named or none-specific (Sun et al., 2020).
- **Fog** means that the Digital Twin is deployed on another device than its counterpart but still resides in a local network (Kloibhofer et al., 2018).
- **Edge** when the Digital Twin is deployed on the same device as its counterpart (Saini et al., 2018).

This facet also includes data regarding Digital Twins host of unspecified provenance with respect to the hosting alternatives above but on the kind of system hosting the Digital Twin:

- **Data Management System** if the host of the Digital Twin is a database or some other data-centric application (Rauch and Pietrzyk, 2019).
- **Simulator** incorporates Digital Twins that are employed as part of a simulation, such as 3D simulation models (Ayani et al., 2018).
- **Virtual Reality** for Digital Twins deployed in a virtual reality (Mohammadi and Taylor, 2017).

Reported findings describe the hosts’ location, such as in the cloud, on an edge device, or directly on the twinned system; or describe the host’s kind, including data management systems, simulations, and virtual reality. As such, we encountered publications with multiple findings in this facet, e.g., if the Digital Twin was deployed on some database in the cloud. But also findings with multiple reports of the host’s location where possible, in cases where Digital Twin could be deployed either in the clouded or some local system.

**RQ-4.2 Digital Twin Counterpart Connection Facet.** As Digital Twins seem to be connected with their real world counterpart, we investigated in the context of RQ-4.2 which technologies are used to connect the Digital Twin with its real world counterpart. To understand these connections we classify the used technologies as follows. First, we identified a set of hardware or technology related connections that were often described in our corpus:

- **Local Networks** connect Digital Twins by establishing BUS systems (Jain et al., 2020), Ethernet (Zhang et al., 2017a) or WiFi (Luo et al., 2019).
- **Short distance wireless communication** subsumes non-LAN short distance communications, such as RFID (Kannan and Arunachalam, 2018) or Bluetooth (Krajcovic et al., 2018).
• **Data** connect Digital Twins with their counterparts by its data as for example described in Yun et al. (2017) which use a database.
• **Server/Cloud/Proxy** in this connection type the Digital Twin is connected with the real world counterpart based on a remote server such as a cloud application (Verner et al., 2018).

On the other hand, we also found connection descriptions that solely focus on the used communication protocol:
• **Industrial protocols**, such as M-Connect (Hu et al., 2018) or OPC UA (Liu et al., 2018a).
• **Internet protocols**, such as TCP/IP (Liu et al., 2019a), UDP (Kuts et al., 2017), or HTTP (Ding et al., 2019).
• **IoT Protocols**, such as (Eyre et al., 2018).

Of course, not all publications of our corpus strictly differentiate between the communication technology and the protocol. Thus, multiple selections were possible.

**RQ-5.1 — Decision Making Facet** To influence the observed entity, a Digital Twin should be able to make decisions based on its counterpart’s current state and condition. Thus, we distinguish in **RQ-5.1** between the following classes of decision-making functions to better understand the nature of these decisions.

- **Data Mining**, such as big data methods (Tao et al., 2018a), or data cleansing techniques (Yusupbekov et al., 2018).
- **Machine Learning**, such as artificial neural networks (Ding et al., 2019), or deep learning (Uzun et al., 2019).
- **Reasoning** techniques, further distinguished into
  - **Case-based Reasoning**, e.g., (Kaivo-oja et al., 2019).
  - **Symbolic Reasoning**, such as (Wantia and Roßmann, 2017).
  - **Stochastic Reasoning**, for instance (Li et al., 2017).
  - **Other Numeric Reasoning**, classifying remaining methods such as (Yan et al., 2018).

- **Simulation**, such as finite element analysis (Zambal et al., 2018), virtual testbeds (Di Maio et al., 2018), or rigid body dynamics (Mars et al., 2018).

As some contributions employ multiple decision-making functions for their Digital Twins or use combinations of different techniques, these classes are generally not disjoint. For instance, in Renzi et al. (2017), the Digital Twin uses a data mining to process gathered data and furthermore simulates values that could not be obtained from the physical counterpart. If a publication did not explicitly specify any corresponding functions, we classified the Digital Twin to offer no decision-making functionality.

**RQ-5.2 — Digital Twin Input and Events Facet** Digital Twins rely on specifications that define how the Digital Twin should behave in different situations, e.g., when context changes occur, when external inputs are given, or when the equivalent acts in a certain way. Inputs for Digital Twins have different sources, as humans that explicitly control the Digital Twins’ actions or models that specify the Digital Twin’s behavior. Digital Twins also react to events that occur in their operating context or in the physical entity that they represent. We introduce the input facet to classify the input data that the Digital Twin relies on upon and differentiate between different input data sources as follows:

- **Machine Data** specifies all data captured by sensors (Padovano et al., 2018) or emitted by machines (Lu and Xu, 2018a).
- **Models and Simulations** includes data that is provided by simulations (Ciavotta et al., 2017) or specified through models (Thomas et al., 2018).
- **User Specifications** characterizes all Digital Twins that a configured via direct user input, e.g., through a user interface (Zhang et al., 2017a) or motion capturing (Peruzzini et al., 2020).

For the papers where a description was given, this facet could generally be identified unambiguously. However, in some cases, it was not reported how the DT was configured. We did not assign a facet to these papers and classified these as non-assignable.

**RQ-5.3 — Digital Twin Output Facet** Digital twins that fulfill some kind of purpose other than modeling their counterpart often interact with their environment through inputs and outputs. Outputs can take different forms and have various intentions. To better understand what kind of outputs Digital Twins can produce, we have grouped the contributions according to the following schema:

- **Observations**, if the Digital Twin represents the current state of the twinned system, i.e., monitoring data.
- **Prescriptions**, for Digital Twins emitting instructions send by the digital twin about changes that should be applied, incorporating parameter configuration changes, detailed control commands, and planning data.
- **Predictions**, when the Digital Twin produces predictions or estimations about the system behavior, failures, and life expectancy, i.e., what can be.
- **Other Data**, when Digital Twins output data but the nature of this data is not further specified.
- **Visualization & 3D Models**, when the output of Digital Twins are changes to a UI or 3D models for visualization of their counterpart.

As these categories overlap, publications may be assigned multiple times.

Generally, contributions reported that the output of Digital Twins can represent the current state of the system, contain predictions or estimations, or describe changes and modifications that should be applied. In some cases, the output of Digital Twins was described as less detailed, only stating that Digital Twins emit some kind of analysis result or some kind of data. Also, some contributions reported visualization of Digital Twins through a UI or 3D models as an output of Digital Twins. If a publication did not explicitly report outputs for Digital Twins, then we regarded the corresponding information is not available.

**RQ-6.1 — Digital Twin Evaluation Maturity Facet** Digital Twins are expected to improve our understanding and use of systems. To understand how mature the research results within our corpus are, we classify their contributions according to the technology readiness level (TRL) (Hédé, 2017) of their evaluations or case studies. As publications rarely can provide fully detailed evaluations due to, e.g., page limitations or confidentiality considerations, a precise estimation of evaluation maturity is rarely possible as well. Hence, we classify evaluation maturity as follows:

- **Proof of Concept** (TRL 1-3), includes evaluations in which at least basic principles of the research can be observed and at most an experimental proof of concept is reported.
- **Technology** (TRL 4-6), includes evaluations where technology is at least evaluated in a laboratory context and at most in a relevant environment.
- **System** (TRL 7-9), includes evaluations in which at least a system prototype is demonstrated in an operation environment.

Each publication was assigned a single TRL.
4. Vertical Analysis

During the vertical analysis, we aim to provide quantitative results for all research questions, where an answer for this research question without considering other research questions was possible. In the following subsections, we present the results of this analysis for our questions. As not all research questions could be answered unambiguously or with sufficient significance based on the publications in our corpus, observations and potential insights to these research questions are briefly revisited in the discussion.

4.1. Contribution type

The papers in our corpus address various topics and represent different types of contributions. Accordingly, we classify the publications in our study by their type of contribution (Petersen et al., 2008). The classified contribution types are disjoint, and thus, each publication was classified to exactly one contribution type. Publications suitable to more than one contribution type were classified to the most suitable.

Overall, the majority of contributions address methods (223, 62.64%) or concepts (98, 27.53%), whereas contributions mainly addressing tools (19, 5.34%), analyses (13, 3.65%), or metrics (3, 0.84%) are considered far less often. The distribution of contribution types is shown in Fig. 3.

The major focus on concepts and methods and the lack of analyses may be a symptom of the still young field of research on Digital Twins. Research and industry have not advanced enough, and exhaustive solutions employing Digital Twins exist whose effect could be analyzed in detail. Surprising is the lack of tooling, which would be needed to employ Digital Twin solutions. While some contributions on metrics for Digital Twins exist, most of these are employed in a broader research concern, with only a few publications focusing on metrics in particular. Focusing on tooling for Digital Twins realizing the presented concepts and methods could move research on Digital Twins forward in the future.

4.2. Research type

Besides the contribution type, we also analyzed the publications research type and classified the publications according to the schema presented in . Again, each paper was classified to exactly one research type, that is to the most suitable research type if a paper was eligible for more than one category.

We found that solution proposals are the most common research type in Digital Twin research, which make up 233 (65.45%) of all publications in our corpus. Other research types are far less common. Out of the 356 publications in our corpus, vision papers only contribute 59 (16.57%), experience reports only 37 (10.39%), evaluation reports only 16 (4.49%), and validation papers only 11 (3.09%) publications to the overall corpus. The distribution of research types is shown in Fig. 4. It is similar to the distribution of the classification of contribution types in the sense that two-thirds of the publications also contribute to a single class.

A similar distribution of research and contribution types might reflect constructive research on Digital Twins, which is reinforced in particular by the large number of publications describing solutions. Despite the many papers describing methods or concepts, there are only a few publications that focus primarily on validation. One of the upcoming goals in Digital Twin research should be to validate and evaluate existing approaches and solutions in detail.

4.3. RQ-1.1 — Digital Twin application domains

While manufacturing and Industry 4.0 might come to mind first, when considering the application domains of Digital Twins, there are plenty of other domains Digital Twin research is applied to. By classifying the studies according to the schema presented in the corresponding facet, we identified eight domains that research on Digital Twins focuses on. The domains, together with the number of publications addressing these, are illustrated in Fig. 5.

All 356 (100%) publications of our corpus relate to a specific domain or contribute to generic research on Digital Twins. The large majority of research on Digital Twins focuses on (C) manufacturing (252, 70.79%) and on (J) Information and Communication (47, 13.2%). The remaining (16.01%) publications are almost equally split into further domains and generic Digital Twin research. Of the other domains, (D) energy (17, 4.77%) and (F) construction are (12, 3.37%) more often addressed than (Q) human health (8, 2.25%), (B) mining (9, 2.53%), or (H) transportation and storage (6, 1.68%). We also found publications conducting research on Digital Twins for (P) education (4, 1.12%) and (A) agriculture (1, 0.28%). For the other twelve classes of economic areas, we did not find research on Digital Twins.

This especially holds for areas where digitalization and automation might not be as advanced as in manufacturing, such as (E) water supply and sewage, (G) wholesale, (L) accommodation and food service activities, (L) real estate activities, or (R) arts and entertainment. This maybe be because they do not need a Digital Twin, use different terminology, or we have simply missed publications as they are not indexed by our databases or the people involved do not publish their results in scientific literature.
Many of the areas not addressed by publications in our corpus also are areas in which human actions and decisions are central to creation of added value, such as wholesale, accommodation, financial activities, real estate activities, public administration, arts and entertainment. The lack of publications addressing these might be a symptom of properly capturing Digital Twins of human actors, which is in line with the small number of Digital Twins of beings as observed regarding RQ-2.1.

4.4. RQ-1.2 — Digital Twin purpose

One question any application of new concepts, methods, or tools are faced with is their purpose. Thus, we analyzed which purposes Digital Twins fulfill. Based on the classification scheme presented in the corresponding facet, we aimed to answer RQ-2 and concluded how many publications mention a purpose and how these purposes distribute over the purpose facets. Of our corpus (356, 100%), 347 (97.47%) publications make the purpose of the Digital Twin explicit. Consequently, a majority mention the purpose of the Digital Twin explicitly, which distribute over these purposes as illustrated in Fig. 6. Moreover, we found out that most Digital Twins have more than one purpose (196, 55.06%).

4.5. RQ-2.1 — Digital Twin counterparts

To uncover which kinds of counterparts Digital Twins are used with, we analyzed the publications in our corpus for this aspect. We found that mostly all publications made explicit what the counterpart of the presented Digital Twin concept is. Out of the 356 (100%) publications, a total of 350 (98.31%) publications make the counterparts of their Digital Twins explicit. Overall, 31 (8.99%) publications present research in which the Digital Twin supports more than one counterpart, for instance, when Digital Twins for the production system and the produced product (Lohtander et al., 2018a) or a production process on an individual system (Liu et al., 2019a) are considered. Overall, 384 counterparts are reported by the publications included in our survey, as illustrated in Fig. 7.

The predominant counterparts of Digital Twins are individual systems and systems of systems (210, 54.69%) and systems of systems (104, 27.08%), which make a total of 314 (81.77%) of the Digital Twin counterparts identified in our corpus. Digital Twins for beings, processes, products, and other counterparts are significantly less common and make only a total of 70 (18.23%) publications to the counterparts of Digital Twins.

Finding most Digital Twins relating to counterparts that are individual systems and systems-of-systems is not unexpected. However, the latter entails questions regarding the communication and (de)composition of Digital Twins that are further investigated in the context of RQ-2.8. Especially when individual systems can flexibly enter or leave system-of-systems structures, such as within smart manufacturing, automated convoys, or distributed Internet of Things systems, the interfaces of Digital Twins, their interactions, and means for flexible composition need to be understood.

The low number of publications contributing Digital Twins of products is unexpected as the smart product of lot-size 1 is one of the driving visions of Industry 4.0, and Industry 4.0 is one of the main disciplines driving research on Digital Twins. However, in line with the increasing number of publications on Digital Twins...
and the finding that the digital representation of assets still is one of the prime research topics, at least in modeling for Industry 4.0 (Wortmann et al., 2020) suggests that before the product can be twinned, first the assets and the processes relating to its production must be considered. Yet, there also is a small number of publications on Digital Twins relating to processes that are not tied to one or more systems directly.

The overwhelming focus on contributions to engineering Digital Twins for systems consequently indicates that research still is in a very early stage of understanding the systematic engineering of Digital Twins, means to reuse parts of Digital Twins for Digital Twins of different counterparts, and suggests that established reuse techniques from software engineering, such as encapsulation, type-based substitution, product lines are not as common for Digital Twins yet. We assume the latter is due to the different perspectives on Digital Twins as (design-time) models, (run-time) systems, or something in-between and the heterogeneous implementation techniques that are employed accordingly. Research on heterogeneous modeling (Lee, 2010) and software language engineering (Kleppe, 2008; Hölldobler et al., 2018) can contribute to closing the gaps between the different technological spaces (Kurtev et al., 2002) and applying established reuse techniques to Digital Twins systematically.

4.6. RQ-2.4 – Digital Twin lifecycle

To understand how Digital Twins are applied to the different lifecycle phases of their counterparts, we classified the publications accordingly. As designed Digital Twins consider the ideal design of their physical entities, thus not taking into account minor derivations that may occur during the construction of the counterpart. As-manufactured Digital Twins do consider these derivations, while as-operated Digital Twins also include usage data which may inflict the physical counterpart’s behavior or appearance. Fig. 8 shows the distribution of described Digital Twins as a Venn diagram. In our corpus of 356 (100 %), all but 17 (4.77 %) publications made explicit which lifecycle the presented Digital Twin represents. A total of 60 (16.85 %) publications describe Digital Twins for more than one lifecycle, for instance, when Digital Twins for the design and manufacturing were combined (Wärnemjord et al., 2017). Overall, 29 (8.15 %) publications also presented Digital Twins that were used across all lifecycle phases.

Most Digital Twins (266, 74.72 %) represented the operation lifecycle phase of the physical entity. This might indicate that Digital Twins are often employed for simulating the physical entity’s behavior, e.g., if it is not built yet, or to test new application scenarios before they are realized. Also, when Digital Twins fulfill informative and representing requirements, they also integrate runtime sensor data to mirror the entity’s state. It should also be pointed out that only 57 (16.01 %) publications consider the manufacturing of the physical counterpart, and only 15 (4.21 %) publications focus the manufacturing exclusively. Many of the Digital Twins described in the literature represent CPS with a long lifecycle, e.g., production machines. In such systems, sensors are often retrofitted, and Digital Twins are developed while the machine is already operating (brownfield Digital Twin development), to represent their counterparts as-operated, which explains the high number of publications reporting on Digital Twins for this lifecycle phase. However, representing the design of future systems by a Digital Twin can be beneficial to evaluate, e.g., different variants before the system is realized. Therefore an increase in design-time Digital Twins is expected in the future.

Another possible explanation for the derivation between frequency of as-designed and as-operated could be that the physical twin is designed once, but then multiple instances conforming to this design are produced. Thus, leading to only one as-designed Digital Twin but multiple as-operated Digital Twins where the as-operated Digital Twins represent different instances conforming to the same design. To enable co-evolution of physical objects and their Digital Twins, future research also should be conducted on the transformation from as-designed Digital Twins to as-manufactured Digital Twins and to as-operated Digital Twins.

Combinations of the different lifecycle phases are generally not researched thoroughly. Especially, transitions between Digital Twins (i) as-designed and as-manufactured; (ii) as-manufactured and as-operated; and (iii) as-operated back to as-designed yield promising potentials for a pervasive model-driven DevOps of Digital Twins that saves development time in the future.

4.7. RQ-2.6 – Digital Twin optimization

With RQ-2.6, we investigate whether Digital Twins are used for optimization and whether the Digital Twins optimize their counterpart, themselves, or both. To this end, we classified Digital Twin optimizations mentioned in the publications of our corpus according to the classification schema presented in Section 3.4.

Overall, we found that out of 356 (100 %) publications, 193 (54.21 %) publications explicitly perform optimization, whereas 163 (45.79 %) publications do not mention or consider Digital Twin-based optimizations. Of the 193 publications making the Digital Twin optimization explicit, only 6 (1.68 %) publications present Digital Twins that only optimize the Digital Twin. Most publications present Digital Twins that optimize their counterpart (138, 38.76 %) or optimize both (49, 13.76 %) (see Fig. 9).

These numbers show that most of the publications on Digital Twins of our corpus describe Digital Twins that optimize their
twinned counterpart. Moreover, there is a clear trend to only optimize the virtual counterpart without adapting or optimizing the Digital Twin itself. As there is also a smaller proportion of papers that either describe the optimization of the Digital Twin alone or the Digital Twin and the observed system, it can be followed that further research on methods to optimize the Digital Twin parallel to the observed system might be required. In this context self-optimizing Digital Twins might benefit from models at runtime or self adaptive sources (Padovano et al., 2018).

4.8. RQ-2.7 — Digital Twin parts

With research question RQ-2.7, we aim to find out the essential components that are part of Digital Twins (see Fig. 10). We found that out of 356 (100 %) all but 33 (9.27 %) publications made explicit what they consider part of the reported Digital Twin. Of the publications making the parts explicit, a total of 144 (40.45 %) publications describe Digital Twins composed of more than one of the facets of parts, while none of the publications report on a Digital Twin that is composed of elements from all facets.

Hardware components are also named as components of the Digital Twin. Since the Digital Twin is a digital object, the number of 49 (13.76 %) papers that also name hardware components seems surprisingly high. A possible explanation are cyber-physical components whose functionality is realized by a combination of hardware and software components. This makes the boundary blurry and since software and hardware are delivered together, it is less recognizable for the user. In many applications, a combination of a Digital Twin together with a physical model is utilized to provide the user with haptic feedback. This is especially the case for Digital Twins used in training, e.g., for medical professionals (Laakki et al., 2019).

Most publications (274, 76.97 %) mention models as parts of the Digital Twin. The terms “model” and “Digital Twin” are even used synonymously (Talkhestani et al., 2018b). These models mainly describe the physical counterpart’s constraints (133, 37.36 %) and its appearance (94, 26.4 %). Only a few publications (18, 5.06 %) apply models for describing data structures and only 30 (8.43 %) publications describe Digital Twin behavior through models. The most frequently reported model types were simulations, physical models, and geometric models. This is consistent with the fact that many described Digital Twins come from the engineering field, where these types of models are highly prevalent (Glaessgen and Stargel, 2012).

Digital Twins often offer services, e.g., to evaluate system states (Pargmann et al., 2018) or to influence system behavior (Graessler and Poehler, 2018). Thus, it is not surprising that (39, 10.95 %) publications mention software as a part of the Digital Twin.

Since Digital Twins often monitor CPSs at runtime, data is also mentioned as part of the Digital Twin frequently (129, 36.24 %) as well. To fulfill their representative purpose Digital Twins need information about the underlying system, which makes the data a reasonable part. Therefore, intelligent data processing and storage could become an important functionality of Digital Twins in the future, enabling them to remain up-to-date representation of the twinned system despite further growing data volumes.

4.9. RQ-3.1 — Implementation

Multiple publications not only elaborate on the conceptual foundations of Digital Twins but also provide a detailed explanation about used techniques for their implementation. Thus, RQ-3.1 analyses different facets of realizations. Our goal is to identify key technologies or methodologies to implement Digital Twins. Overall, 191 publications contain information on implementation details. Thus, the following classification results refer to the total number of publications that actually contribute to the research question. As different technologies can be used in combination for realization, the following statistics are not disjoint.

Overall, 71 (37.17 %) Digital Twin concepts are implemented using CAD or 3D models. Furthermore, 31 (16.23 %) of the papers follow data-driven approaches, such as standardized data formats (e.g., JSON, XML) or complete database systems. General-purpose programming languages make a total of 65 (34.03 %) publications. 59 (30.89 %) approaches are realized via mathematical or physical models, and 38 (19.9 %) use a model-based or model-driven approach. Finally, 42 (21.99 %) papers consider simulations or similar analyses when implementing these twins. Fig. 11 presents the corresponding distribution.

In summary, most of the considered publications use 3D and mathematics-based models as well as general-purpose languages. Simulations, model-based, and data-driven approaches are also widely applied for implementing Digital Twins. Most publications also describe a composite approach of several technologies to realize Digital Twins. Furthermore, 8.71 % of all publications describe the appliance of types or another kind of reuse. This relatively low percentage shows that many Digital Twins are still created in a purpose-driven way, without relying on a consistent foundation.

CAD or, more general, 3D models being the most prominent way of implementing Digital Twins is consistent with the findings of RQ-3.2 and could indicate that a replica comprising the physical characteristics is often required. Furthermore, the use of

---

**Fig. 10.** Distribution of Digital Twin constituents with models as the predominant factor.

**Fig. 11.** Distribution of implementation techniques for Digital Twins.
general-purpose languages is significantly high. This suggests that for many aspects of a real-world entity or process, suitable tools do not yet exist, such that many features have to be implemented manually on an individual basis. Additionally, Digital Twins are often based on physical models, model-based techniques, simulations, and slightly less on data. Overall, this distribution seems to indicate that currently, these twins primarily rely on model-driven or analytic approaches rather than purely data-driven techniques.

4.10. RQ-3.2 — Digital Twin tooling

The purpose of RQ-3.2 is to understand which kinds of software tools are applied to the engineering and operations of Digital Twins. To this effect, we classified the tools mentioned in the publications of our corpus according to the classification schema presented in the corresponding facet.

Overall, we found that only 186 (52.25 %) publications of 356 publications make the tools employed explicit. Various publications to Digital Twin architectures or infrastructures present concepts or methods that are unrelated to specific tools. Out of the 186 publications making the employed tools explicit, the most popular category of tools is simulation tools (70, 19.66 %), which includes the Maya Simulation Framework (Ciavotta et al., 2017), Siemens PLC Sim Advanced (Wuttke et al., 2019), Simumatik 3D (Ayani et al., 2018), Verosim (Wanta and Roßmann, 2017), and more. The second most often category of tools focuses on computer-aided design and manufacturing (CAX) tools (53, 14.89 %), which includes Autodesk Revit (Kawunruen and Xu, 2018), CATIA (Gregorio et al., 2019), Delima 3D Experience (Demkovich et al., 2018), Siemens NX (Anand et al., 2018), SOLIDWORKS (Thomas et al., 2018), and similar tools.

Tools for process management (37, 10.39 %), communication (24, 6.74 %), data management (32, 8.99 %), visualization (35, 9.83 %), as well as the direct use of programming languages (34, 9.55 %), and model-driven development tools (33, 9.27 %), are less common than simulation and CAX. Moreover, there also is a large number of publication using various other software (36, 10.11 %), which includes website development for Digital Twin representation with Apache Kepler (Radenko et al., 2018), interfacing specific robot APIs (Meng et al., 2019; Yan et al., 2018), data modeling with Microsoft Excel (Caputo et al., 2019), or specific programming environments (Priggenmeyer et al., 2018). Overall, the main categories of tools employed in the engineering and operations of Digital Twins are as illustrated in Fig. 12.

The most prominent tools to engineer Digital Twins are simulation tools and CAX tools. This might indicate that some notions of Digital Twins indeed aim for a sufficiently precise replica of the twinned system that can be subjected to experiments as a substitute for the twinned system itself. Moreover, this might entail that Digital Twins are predominantly researched in domains being used to describing systems for a physical–geometrical perspective (cf. RQ-1.1). As both simulation tools and CAX tools traditionally are employed to engineer systems, i.e., prior to the deployment of the system under development and its operations, this furthermore might suggest that there is a focus on Digital Twins used at design time of the twinned system. On the other hand, the widespread use of process management software, communication software, and data management software indicates that there also is extensive interest in observing and possibly optimizing the behavior of the twinned system at runtime.

The horizontal analysis on the use of tools relative to the domain (RQ-1.1), its purpose (RQ-1.2), and lifecycle (RQ-2.4) of the respective Digital Twins discusses this.

The number of process management tools and data management tools, ranging from traditional databases to data analysis tools might portend that Digital Twins also are about better understanding the twinned system and its operations in its context. Yet, in contrast to RQ-2.1 according to which thinning systems-of-systems is important, the low number of communication tools employed to engineer and operate Digital Twins might suggest that from a Digital Twin perspective, Digital Twins of systems-of-systems are generally considered a single, monolithic system instead. This might be due to the lack of support for composing Digital Twins (cf. RQ-2.7).

With one interpretation of Digital Twins being that these are models of the twinned systems—which appear to be the predominant perspective on Digital Twins for the researchers employing simulation tools and CAX tools—the lack of applications of MDE tools is surprising. Again, this might be due to the large number of publications focusing on Digital Twins in manufacturing included in our corpus. Also, the number of papers leveraging general programming languages is relatively low. This might suggest that Digital Twins often are engineered and operated by reusing existing software, such as specific tools for simulation, data management, or visualization. If research on Digital Twins does not require new software, this might suggest that Digital Twins are not a new paradigm or kind of software per-se, but the combination of existing paradigms, methods, and tools for a new purpose. This also is in line with the observed lack of special software solely focusing on Digital Twin engineering or operations. The horizontal analysis of the use of tools (RQ-3.2) relative to the purpose (RQ-1.2) highlights this.

Finally, we found the lack of research employing artificial intelligence tools, including machine learning, knowledge representation, and planning, surprising.

4.11. RQ-3.3 — Digital Twin development processes

Analyzing whether Digital Twins are developed together with their counterpart or in an independent process in RQ-3.3, we identified that only 232 (65.17 %) make the engineering process of the Digital Twin explicit. Various publications address the usage of Digital Twins or broader concepts and do not address the engineering of Digital Twins or their counterparts.

The publications that made the engineering of Digital Twins explicit could be categorized into two categories, those which describe an engineering process intertwined with the counterpart and those that describe that Digital Twins are developed in a separate process. Furthermore, we identified whether the development of Digital Twins incorporated knowledge about the manufactured system or not if they are not developed together.
with their counterpart. Of the 232 (65.17%) publications making the development process of Digital Twins explicit, a total of 7 (1.97%) of 356 publications describe processes for Digital Twins both together with the development of the counterpart and also independent of it. Overall, 29 (8.15%) publications report that the Digital Twin is developed together with the system, and 196 (55.06%) that it is not. Of the latter, 14 (3.93%) publications report that the development of Digital Twins is independent of any manufactured counterpart, whereas 26 (7.3%) report that the development of a Digital Twin incorporates knowledge about the manufactured counterpart (see Fig. 13).

It is striking that Digital Twins are primarily developed in a separate development process, i.e., not in a joint engineering process with the actual system. This could be a result of historical developments or because the Digital Twin and its counterpart are primarily regarded as separate entities. Moreover, this might entail that Digital Twins are mostly researched in domains where there already exists strong engineering processes independent of Digital Twins. Integrating Digital Twins and the information they provide into a complex engineering process could present greater challenges and require a corresponding shift in mindset. After all, some articles describe the development of Digital Twins as anticipating the development of their counterparts. This suggests that information gained from Digital Twins is already flowing into the development of their counterparts, for example, for design space exploration. The next step here would probably be to develop Digital Twins and counterparts together in order to allow information to flow iteratively into the development process and thus to be able to react to changes in the development. However, it is also interesting that in some cases, the Digital Twin is developed after the actual counterpart. This could occur, for example, in cases where the counterpart is retrofitted with a Digital Twin that then interacts with the system at runtime, for example, to control or influence the system. However, we found that for the actual engineering of and with Digital Twins, there is a lack of research.

4.12. **RQ-4.2 – Digital Twin connections**

We investigate how Digital Twins are connected to their counterpart. To this end, we classified Digital Twin connections mentioned in the publications of our corpus according to the classification schema presented in Section 3.4.

We found that 77 (21.63%) explicitly connect the Digital Twin with their real-world counterpart. Out of these publications, 32 (8.99%) explicitly name the technology they used for this connection. The technologies, together with their occurrences, are illustrated in Fig. 14.

Most of the connected Digital Twins are connected with their counterparts via a local networks (15, 4.21%), or short distance wireless communication (8, 2.25%), such as Bluetooth (5, 1.4%) and RFID (3, 0.84%). Other publications mention that the Digital Twin is connected to the twinned system through data access via some database or data format (7, 1.97%) and via some cloud or server (2, 0.56%). Furthermore, we classified the used communication scheme as also described in Section 3.4. The results are illustrated in Fig. 15.

Most of the publications used protocols from industrial control system environments (21, 5.9%). Of these publications, the majority (15, 4.21%) use OPC UA and MThread (6, 1.68%) to connect the Digital Twin with its real-world counterpart. Some publications mention MQTT (5, 1.4%) and other IoT protocols (11, 3.09%). Furthermore, internet protocols (8, 2.25%) are used in some cases to connect the Digital Twin with its real-world counterpart.

From these numbers, it becomes clear that Internet technologies and protocols are the predominant means to connect Digital Twins with their counterpart. Moreover, it is easy to see that technologies and networks from IoT applications are also an important part of Digital Twin development and connection. The large number of industrial control system communication protocols and IoT protocol also meets our observation from **RQ-1** that Digital Twins are mostly used in the manufacturing domain in the context of industrial control systems. However, we can see from the relatively small number of publications that communication between the Digital Twin and its counterpart is currently not the main focus of research.
4.13. **RQ-5.1 — Decision making functions**

The notion of Digital Twins often comes with the intention of optimizing systems or process optimization. Therefore, **RQ-5.1** investigates the possibility of decision making functions. The goal was to explore the most prominent decision methods used for Digital Twins. In general, it is noteworthy that 234 (65.73%) publications do not describe any explicit decision making functionality. Therefore, the following statistics apply to the remaining 122 publications.

As certain publications use several decision making functions, the categorization is not disjointed. Overall, 55 (45.08%) Digital Twin concepts use some kind of reasoning. Further investigation shows that only a few case-based reasoning is performed (four publications in total) compared to symbolic, stochastic, or other numerical reasoning methods (14 to 19 papers each). Other numerical reasoning represents the subset of reasoning methods that do not fit into the remaining three categories. Furthermore, 41 (33.61%) of the presented Digital Twins use machine learning techniques, and 31 (25.41%) rely on simulation. Data Mining techniques have the least impact, with only 12 (9.84%) publications reporting on this topic. Fig. 16 presents the corresponding distribution.

In summary, only one-third of the publications describe decision making in combination with Digital Twins. Methods of reasoning, simulation, and machine learning seem to have made significant advances. Data Mining techniques are severely underrepresented. This is an interesting fact indicating that most decision making processes rely on analyzing near real-time data and do not perform exhaustive computations on historical data. This might be due to the lack of historical data and change in the future accordingly.

4.14. **RQ-5.2 — Digital Twin inputs and events**

We aim to understand how Digital Twins gain information about their counterparts and the operating context and to which events and external inputs they react (see Fig. 17). We found that all but 129 (36.24 %) publications made explicit on which input the presented Digital Twin relies. Of the publications making the Digital Twin’s inputs explicit, a total of 45 (12.64 %) described Digital Twins processing multiple types of inputs. For example, several Digital Twins react to changes of machine data but also support reconfiguration (Mukherjee and DebRoy, 2019), where a Digital Twin of a 3D printing machine automates experiments to detect parameters for achieving desired product attributes. For this purpose, the Digital Twin analyzes sensor data and also provided 3D specifications of the produced part. Most Digital Twins (177, 49.72 %) react to machine data as input. This aligns with the large number of Digital Twins from manufacturing and might simply stress that Digital Twins are often applied for automation in manufacturing. From these numbers, it is easy to follow that many Digital Twins tend to react to the data they receive or measure from their observed systems, from the system’s users, or the systems environment. In addition, some Digital Twins appear to react to damage or fault events, which is consistent with our observation from **RQ-1.2** that Digital Twins can be used for maintenance purposes.

4.15. **RQ-5.3 — Digital Twin output**

We also aim to uncover whether Digital Twins produce outputs and, which kinds of outputs they produce (see Fig. 18). Thus, we identified publications that reported that Digital Twins emit some output or affect the environment they are deployed in. We classified the kind of these outputs according to the classification schema presented in the corresponding facet.

Of the examined publications, 227 (63.76 %) made explicit that Digital Twins emit some kind of output. In cases where Digital Twins do not provide outputs, they could represent structural models instead of software that provides analyses. In some cases, Digital Twins performed analyses. However, it was not stated what happens with the analyses results or how these influence the environment or the counterpart.

Out of the 227 publications making outputs of Digital Twins explicit, a total of 66 (29.07 %) broadly stated that Digital Twins produce some kind of analyses result or emit some kind of data, but did not report any specifics on these outputs. This was, e.g., the case in publications that presented a high-level concept of Digital Twins that could fulfill varying purposes. Interestingly, 33 (14.54 %) papers reported that Digital Twins had as output some kind of effect on visualization by updating information shown in user interfaces or even updating producing whole 3D models. A total of 44 (19.38 %) publications reported that the output of Digital Twins represents the current state of the counterpart, including information about material or energy consumption, defect information, or current system behavior. Instructions or modifications, i.e., what should change (76, 33.48 %), or predictions and estimations (42, 18.5 %) are also common outputs of Digital Twins.

Digital Twins can produce various outputs and consequently affect their environment, respectively counterpart in different ways. Most prominent are outputs that provide controlling data or instruction to the counterpart, ranging from changes of the parameterization to elaborated planning. These outputs are intuitive as they describe a strong interaction between Digital Twin and counterpart. The Digital Twin serves in particular as controller of the counterpart. With pure monitoring approaches, which also make up a substantial part of the examined publications, the question arises why Digital Twins are needed here. Monitoring approaches are probably intended to record the current state of the system as accurately as possible and make this information available to other systems. More sophisticated Digital Twins can not only examine the current state of the system but also make predictions, such as analyzing expected lifetime or predicting failure probabilities. Such Digital Twins could be used to monitor safety-critical systems in particular.
4.16. Further insights

Not all research questions could be answered reliably by the publications included in our corpus. For these questions, this section presents our observations.

RQ-2.2 – Multiple Digital Twins. In our survey, we investigated whether the described Digital Twins are unique or if the described physical entity may have multiple Digital Twins. The majority of papers (251, 70.5 %) did not explicitly state how many Digital Twins are supported by their approach. Overall, only 63 (17.7 %) publications explicitly excluded the possibility of multiple Digital Twins while 42 (11.8 %) publications supported the idea of multiple Digital Twins.

RQ-2.3 – Lifetime. Throughout the mapping study, we also tracked whether the described Digital Twins were operated at design time or at runtime of the physical counterpart. Most publications (282, 79.21 %) report on runtime Digital Twins and 97 (27.25 %) publications describe Digital Twins that are operated at design time of the physical twin.

RQ-2.5 – Interaction Facet. We also investigated whether current research considers the Digital Twin to interact with a twinned system. In our corpus, the majority of publications makes the interaction or the lack of it explicit (277, 77.81 %). Among the publications making interaction explicit, the majority (164, 46.07 %) supports interaction between the Digital Twin and its counterpart. The other (113, 31.74 %) publications do not support such interaction. This might be due to the different times a Digital Twin is employed in the lifecycle of its observed system and is discussed in the horizontal analysis.

RQ-3.4 – Quality Assurance. We also found that only a small number publications (51, 14.32 %) of our corpus consider quality assurance for Digital Twins at all. Where quality assurance was considered, testing (24, 6.74 %) was more prominent than simulation (21, 5.9 %). Also, the number of publications considering the online verification of Digital Twins with their counterparts is vanishingly low (7, 1.97 %). This implies a need for further research regarding the quality assurance of Digital Twins (1) at design-time of the Digital Twin; (2) during design-time of the twinned system; and (3) during runtime of both systems. Especially, the fidelity of Digital Twins, i.e., the verification that these can properly represent the twinned system at needs further investigation.

RQ-3.5 – Requirements. In addition to quality assurance measures, we also investigated whether own requirements for Digital Twins and their development are considered in the publications of our corpus. As a result, we found that only 38 (10.67 %) publications discuss own requirements for Digital Twins. Of these publications, most prominently the necessity of real-time capability was mentioned (7, 1.97 %) closely followed by the requirement that the behavior of the Digital Twin must match the behavior of its real-world counterpart (7, 1.97 %). Finally, it was mentioned that Digital Twins have to be reusable in only (2, 0.56 %) publications, which is almost negligible. The remaining publications that mentioned Digital Twin requirements, were either the only source in our corpus mentioning this requirement, or discussed the necessity of Digital Twin requirements without going into the details.

Since apparently only a few authors have investigated the specific requirements of Digital Twins, it is reasonable to conclude that Digital Twin specific requirements are currently not focused in research. However, the above-mentioned Digital Twin specific requirements obviously focus on important aspects of Digital Twin development such as real-time capability or the Digital Twins behavioral relationship to its physical counterparts. Thus, we think that further research on the requirements for Digital Twins and their implementation may be necessary in future works on Digital Twins.

RQ-4.1 – Digital Twin Host. We were furthermore interested which systems host Digital Twins. The investigation showed a wide variety of concept and technologies used, ranging from the Digital Twin living on the same device as its counterpart to the Digital Twin being deployed in a cloud. Only few publications (3) report that the Digital Twin lives on the edge of its counterpart. Of the publications that reported that the Digital Twin is deployed further away from its counterpart, 81 reported that the Digital Twin its deployed in the cloud, 8 reported that the Digital Twin lives on a specifically named platform, while 11 contributions do not further clarify on what kind of external system the Digital Twin is deployed. Also, 5 publications report on deploying the Digital Twin on the edge of its counterpart. For another 4 publications, the Digital Twin is managed in a database, and for 2 contributions it is part of a virtual reality. In summary, the different implementations show the diverse perception of Digital Twins, but it is clear that network-based systems and management in the cloud are strong pioneers here.

RQ-6.1 – Digital Twin Evaluation Maturity Facet. We found that the majority of publications in our corpus feature some form of evaluation (271, 76.12 %). Overall, proof-of-concept (TRL 1-3) evaluations are significantly prevalent (181, 50.84 %).
Evaluations featuring technologies employed in a laboratory or relevant environment (TRL 4–6) are reported less often (77, 21.63 %) and evaluations featuring system prototypes in operation environments (TRL 7-9) are very rare (13, 3.65 %). As Digital Twin research often aims at the application domains of manufacturing, energy, or construction, the low number of evaluations featuring systems in their operation environments is comprehensible: evaluating a research product in a real factory, power plant, or construction site is challenging and costly. Yet, with Digital Twins aiming to improve productivity, this validation in the field ultimately is necessary to promote Digital Twin research into industrial practice.

5. Orthogonal Analysis

The orthogonal analysis investigates the potential correlations between related dimensions of our classification framework. To this end, we juxtapose several dimensions and further group their data to generate interesting findings. In addition, we investigated other pairs of dimensions which are not explicitly presented in this paper. Based on these investigations, we present the six most interesting analyses. Further analyses can be performed based on the replication package our replication package.

5.1. Digital Twin purpose (RQ-1.2) vs. Lifetime (RQ-2.3)

Out investigation of RQ-2.3 revealed that most Digital Twins operate at runtime of the twinned system rather than at its design-time. However, since we suspect a correlation between the purpose of Digital Twins and their lifetime, we examine this relationship in more detail. Since for both, Digital Twin lifetime and Digital Twin purpose, the identified categories are non-disjoint and a publication can therefore be assigned more than once, the number of combinations considered here is larger than the number of publications. That is, the 356 publications of our corpus contribute to 638 combinations of Digital Twin purpose and Digital Twin lifetime, as shown in Fig. 19.

In the vertical analysis, we identified that Digital Twins are more often employed at runtime than at design-time of the twinned system. Following this trend, CPS Behavior Prediction (145 observations, (79.31 %)) and Monitoring (116 observations) are predominantly performed at runtime. On the other hand, validation is more often performed at design-time (92 observations, (66.3 %)) than the overall trend suggests. Only design-space exploration is more often mentioned in publications presenting Digital Twin at design-time of the twinned system than at runtime, which also meets the intuition that design-space exploration is performed at the design-time of a system.

Interestingly, a high percentage of contributions to design-space exploration of Digital Twins refers to runtime data for the design-space exploration. As the idea of using runtime data of system under development seems counterintuitive, we looked at the publications again to get a better understanding of these cases. By this, it became clear that most of these publications consider multiple purposes at different times of the Digital Twins lifecycle (Rauch and Pietrzyk, 2019; Peruzzini et al., 2020; Chinesta et al., 2020), use real time data from similar products (Kaewunruen and Xu, 2018; Schleich et al., 2018), or perform experiments to get the required real time information at design-time (Debroy et al., 2017).

5.2. Digital Twin purpose (RQ-1.2) vs. lifecycle phase (RQ-2.4)

While the vertical investigation of RQ-2.4 finds that most Digital Twins aim to describe, monitor, or control the twinned system as-operated, this section relates the lifecycle phases of the twinned system that are observed or represented by Digital Twins to the Digital Twins purposes uncovered through RQ-1.2. Through this analyses we aim to better understand for which purposes Digital Twins are used with respect to the observed systems lifecycles and whether there are gaps on this. While some purposes might appear to be obviously related to certain lifecycle phases, such as that Digital Twins with the purpose of supporting system maintenance might more often be used with the systems as-operated, other purposes, such as behavior prediction or enterprise decision making are equally suited for Digital Twins twinning systems as-designed, as-manufactured, or as-operated. Understanding the relation between Digital Twin purpose and the twinned lifecycle phases of the observed system sheds light the use of Digital Twins and can guide further research.

Overall, the 356 publications of our corpus contribute research to 799 combinations of purposes and lifecycle phases. This is due to many publications considering multiple purposes for Digital Twins presented in their research. For instance, the Digital Twin presented in Zambal et al. (2018) aims to ease CPS data processing, monitoring, behavior prediction, and behavior optimization. Hence, this publication contributed four entries to this orthogonal analysis as presented in Fig. 20.

Generally, we have found five times as many publications that focus on Digital Twins of systems as-operated than on the

4 https://zenodo.org/record/6560195
Digital Twin purposes relative to observed lifecycle phases.

As Digital Twins exist at different times (RQ-2.3) and may consist of different parts (RQ-2.7), this section examines the correlation of both corresponding research questions (see Fig. 21). We investigate which constituents are prominent as their diversity is very likely concerning the different purposes of a design-time and runtime twin (cf. Section 5.2). While some parts can obviously be related to a certain lifecycle phase, such as the use of models during design-time, there are also constituents that may exist to unexpected phases (e.g., historical data during design-time). Understanding the relationship between the lifecycle of Digital Twins and their parts assists their further engineering.

Overall, 591 combinations have emerged from comparing the lifecycle and constituentsof Digital Twins. While 428 (72.42%) of the combinations describe a twin at runtime, some interesting correlations still appear.

Generally, Digital Twins that exist during the runtime of the observed entity tend to use relatively more hardware components as during design-time. The same observation can be made for historical and live data. Additionally, this leads to the intriguing question of how a Digital Twin can access such data in the first place since it exists before the system under investigation is put into operation. In general, models are often part of Digital Twins as they are involved in 312 (52.79%) realizations.

The frequent use of hardware components and data at runtime of the system is intuitive since they are effectively available at this lifecycle phase. Generally, the question arises to what extent hardware can be part of a purely digital system at all; however, this result can be related to the authors’ interpretation of Digital Twin constituents. For instance, sensors that are dedicated to produce input for a twin’s computation could be considered part of a Digital Twin. Thus, some authors seem to include the required hardware infrastructure, while others clearly distinguish between hardware and software.

The use of hardware, as well as historical and live data at design-time, can have different origins. The Digital Twin could use hardware prototypes and simulations that produce input data. Furthermore, a twin might be subject to a bootstrapping process, in which a system is developed from previous versions of a similar system, enabling access to legacy components and recorded data traces.
The overall outstanding use of models as part of Digital Twins indicates a growing application of model-driven techniques. Beyond the extensive benefit of models at design-time, however, their predominant use at runtime of a system is also significant. One explanation might be that models that were already created during development are also stored in the Digital Twin (e.g., for documentation purposes). Furthermore, this finding might indicate a growing relevance for models at runtime, applying changeable models in the behavior of the overall system.

5.4. Digital Twin decision making (RQ-5.1) vs. lifetime (RQ-2.3)

This subsection relates the Digital Twin’s lifetime with its decision-making capabilities. Fig. 22 shows a mapping between the possible lifetimes of a Digital Twin that can either be design-time or runtime and different techniques for decision-making that were reported as Digital Twin capabilities. The Digital Twin’s ability to respond to its context seems to be related to its lifetime. Of the publication on Digital Twins investigated in our study, 93 (26.12 %) are able report decision-making capabilities. Of these, 30 (8.43 %) publications reported on decision making at design time while 126 (35.39 %) publications applied decision making at runtime. As these numbers indicate, there must be an overlap between Digital Twins that apply decision making at design-time and those that apply decision making at runtime. More specifically, all Digital Twins that apply decision making at design time also apply decision making at runtime.

At design-time, simulation is especially applied for decision making (11, 36.67 %). At runtime, 40 (31.75 %) Digital Twins applied machine learning, 26 (20.63 %) relied on simulations, and 15 (11.9 %) Digital Twins used symbolic reasoning.

While it is intuitive that Digital Twins perform adaptations autonomously at runtime, there seems to be a research gap for Digital Twins that act on their own at design-time. However, Digital Twins that evaluate different designs at design-time and create an optimal configuration of the designed product could decrease development times in the future.

5.5. Digital Twin connection types (RQ-4.2) vs. lifetime (RQ-2.3)

The goal of this section is to understand which connection techniques can be applied at which lifetime of the twinned system. For example, if communication requires spatial proximity between the communicating entities. In total, 14 (3.93 %) publications reported a Digital Twin that was connected and applied at design time, respectively 114 (32.02 %) publications reported a Digital Twin that was connected and applied at runtime of the counterpart. This discrepancy is quite intuitive since many Digital Twins encapsulated sensor data (Section 4.8) as part of the Digital Twin and thus require a connection to the physical counterpart to acquire this data. These runtime Digital Twins were often connected via local area networks (28, 24.56 %), or support OPC UA (21, 18.42 %) or internet protocols (15, 13.16 %).

Integrating sensor data or historical data into Digital Twins that are employed at design time, can support learning from usage information and adapting future versions of the physical twin. Of the design time Digital Twins (6, 42.86 %) were connected via OPC UA, thus integrated runtime data of operating physical things.

Considering Fig. 23, the majority of design time Digital Twins do not mention their connection, yet. Thus, further research in integrating data from operating twins or the envisioned operation context is still relevant.

5.6. Digital Twin implementation techniques (RQ-3.1) vs. lifecycle phase (RQ-2.4)

The orthogonal analysis relating implementation techniques to the lifecycle phases the Digital Twin addresses (cf. Fig. 24) aims to uncover which techniques are best suited for the twinning a system as-designed, as-manufactured, or as-operated. Consequently, it also might identify gaps in research in form of technologies not applied to specific lifecycle phases.

Overall, the 356 publications of our corpus contribute research to a total of 678 combinations of implementation techniques and lifecycle phases. This occurs as many publications combine multiple implementation techniques and consider several purposes for Digital Twins presented in their research.

Generally, research on Digital Twins as-operated produced two times as many contributions than research on Digital Twins as-designed and four times as many on Digital Twins as-manufactured. The high number of publications not making their implementation explicit (225, 33.18 %) suggests that much research on Digital Twins actually focuses on conceptual research that cannot be translated into Digital Twins without further information.

Hence, out of the five most popular facet combinations, two belong to the “N/A” column, i.e., where the technique of implementation is unspecified. This suggests that there are many publications reporting conceptual contributions to Digital Twin research. In our corpus, these most often are high-level reference.
models on the similar conceptual abstraction than RAMI 4.0 (Han-
kel, 2015) that suggest how to organize architectures of Digital
Twins without implementation (Mukherjee and DebRoy, 2019;
Renzi et al., 2017; Block and Kuhlenkötter, 2019). The other three
most popular facet combinations belong to research on Digital
Twins as-operated while using CAD models, General-Purpose Pro-
gramming Language (GPL) code, or mathematical models, which
suggests that purely data-driven approaches, MDE models and
simulation models are less relevant implementation techniques
for Digital Twins.

For Digital Twins as-designed, CAD and 3D models as well
as mathematical models are the primary implementation tech-
niques, which make up 55 (52.38 %) of the contributions to
corresponding Digital Twin research. While for Digital Twins as-
manufactured, the applied implementation techniques are dis-
tributed almost evenly, the overall numbers of contributions to
such Digital Twins is too small to generalize.

Overall, the data suggests that CAD and 3D models are over-
proportional important for developing Digital Twins as-designed,
where they account for 36 (25.17 %) of the overall as-designed
contributions. In contrast, for Digital Twins as-operated, they only
make up 45 (12.89 %) of the overall as-operated contributions.
Similar observations hold for simulation implementations, which
seem to be more important for Digital Twins as-designed (16,
11.19 %) than for Digital Twins as-operated (31, 8.88 %).

The different prominence of implementation techniques for
Digital Twin research focusing on different lifecycle phases of
the twinned system might suggest a technological gap between
Digital Twins used to twin systems as-designed and Digital Twins
used to twin systems as-operated. This also could explain the
low number of Digital Twin research addressing more than one
lifecycle phase of the twinned system.


While reading the included publications, we noted and syn-
thesized a collection of concerns that need to be considered
when engineering and operating different digital twins. We have
clustered and arranged these in the feature models presented in
the following. Note that each intermediate feature refers to the
research question its subfeatures where extracted from. Overall,
we have identified four dimensions of digital twin engineering
and operations:

1. The requirements dimension comprises concerns that define
the capabilities of the Digital Twin under development. Design
choices within this dimension include identifying the
Digital Twin’s counterpart, defining whether there can
be one or multiple Digital Twins of the same system, and
fixing the phase of the twinned system that the Digital
Twin shall represent. Decisions made along this dimension
govern what the Digital Twin under development will be
capable of.
2. The **realization dimension** comprises concerns about implementation techniques, tools, and methods applied to engineering Digital Twins. Design choices within this dimension include selecting modeling and programming languages, a development process, and suitable quality assurance techniques. Decisions made along this dimension essentially govern how the Digital Twin will be developed.

3. The **deployment dimension** is about bringing the Digital Twin to life and includes concerns about hosting and connecting the Digital Twin. Design choices include deploying the Digital Twin locally, in the cloud, or in-between, installing it on a simulator or in a virtual environment, and selecting appropriate means to connect it to its counterpart(s). This dimension governs where the Digital Twin will exist.

4. The **operations dimension** is about the Digital Twins runtime behavior. It includes concerns about stimuli the Digital Twin reacts to, interaction with other systems (such as enterprise information systems), and decision making techniques influencing its behavior. Hence, this dimension governs what the Digital Twin will do.

Developing a Digital Twin involves making choices for each characteristic along these dimensions. To support this, the feature models presented in the following make these dimensions and their concerns to Digital Twins explicit and guide Digital Twin engineers and users. Therefore, we considered all research questions that apply to these dimensions and have categorized them accordingly.

6.1. **Requirements dimension**

The requirements dimension covers the conceptual foundation for Digital Twins. These cover the basic constituents and characteristics a Digital Twin must or can have to fulfill its purpose. **Fig. 25** provides a general overview of conceptual features: To realize a Digital Twin, there must be some kind of real-world entity in the first place that acts as its counterpart. However, this requirement does not contradict the actual usage phase of the observed entity. Thus, a Digital Twin may exist before its physical counterpart. Overall, we investigated three types of counterparts, the first one of which is a living being, considering an individual. Furthermore, the physical twin can be a dedicated system, for instance, a production machine in a factory. Finally, a Digital Twin can observe a composed system (i.e., a system of systems), where multiple sub-systems are included. This situation focuses more on an overall goal than on supervising individual components. Digital Twins for systems of systems often prove to be very mature.

The second requirement on Digital Twins deals with the question of whether a real-world entity may feature multiple twins. This topic is highly controversial, and there are different approaches. On the one hand, some argue that by the nature of Digital Twins, there can only be one twin for a physical counterpart, managing all tasks for fulfilling its purpose. On the other hand, a Digital Twin might have a specialized view on a distinct part of a system, thus allowing the coexistence of multiple Digital Twins for a single observed entity. While there are pros and cons to both views, twin developers should consider this issue from the start to avoid potential conflicts later on.

A Digital Twin may exist at different stages in the lifecycle of a system. During design time, it supports the development and during runtime the system's operation. There may exist twins that cover both. Furthermore, independent of its stage of existence, a Digital Twin can also represent different lifetime phases of the observed entity. Therefore, Digital Twins can represent an entity as designed, as manufactured, or as operated. Again, multiple selections are possible if the twins should cover more than one specific phase.

As there are different concepts on Digital Twins, the approaches also differentiate regarding the interaction with their real-world counterparts. Some propose that the nature of a Digital Twin always includes direct interaction between the twin and its asset, while others are content with a pure indirect approach. As the kind of interaction (or if any exists at all) strongly depends on the Digital Twin's purpose, different realizations, including a combination of both attempts, are quite possible here.

Furthermore, a Digital Twin can be part of an optimization process. Our study revealed two main possibilities. First, a twin could optimize itself, e.g., to improve its own analyses or give a more accurate representation of the counterpart's state. Second, the observed system can be optimized directly by automatically taking measures for specific situations. Generally, also a combination of both approaches or no optimization at all might be feasible, depending on the goal.

Finally, a Digital Twin must consist of some conceptual constituents. There are multiple different approaches, including hardware and software components, data, models, or again other Digital Twins. Often, a combined effort of different approaches is used. While some findings, such as the reliance on some hardware or software, are expected, there are also further interesting building blocks. For instance, the use of models indicates an increasing notion towards model-driven approaches (cf. Section 4.9). Another interesting aspect is the involvement of other Digital Twins. The possibility of composing different twins to cover distinct sub-tasks comes with new possibilities but also challenges and shows growing sophistication in the development of Digital Twins.

6.2. **Realization dimension**

From an engineering perspective, it is important to know how Digital Twins are implemented, which tools are used for their implementation, and which process is used for the Digital Twin development. For the realization of Digital Twins, we propose a feature model as described in **Fig. 26**. We describe the properties of these engineering aspects in the following.
Every Digital Twin has an implementation, which defines how it achieves its purposes. In general, we identified that Digital Twins describe a counterpart’s geometry, (software) systems, behavior, and general information about the counterpart. To describe a Digital Twin’s geometry, the CAD/3D Model feature provides a modeling implementation for geometry description and design. Moreover, the data feature describes an implementation to handle information about the Digital Twins counterpart. We also noted that behavior descriptions are often implemented as mathematical (including stochastic models) or physical, as well as simulations. Finally, we identified that the implementation of concrete systems is often also realized as handcrafted GPL code and MDE models.

In addition to the implementation, we also identified several tools that may be used for the implementation. For this, we identified A.I. tools that use the information from the counterpart’s data to make predictions and provide services. To handle the required data, data management tools may be used to engineer Digital Twins. Furthermore, CAx and 3D Tools may be used to process and provide geometric models and visualization tools for their visualization. For other simulation purposes, we found out that also general simulation tools are usable in this context. As Digital Twins are often embedded in a production environment, process management and PLM tools may also be helpful for Digital Twin engineering. Finally, communication tools and GPLs may also be used.

For Process, we differentiate between different kinds of development processes. Each product and Digital Twin is either developed jointly or isolated, with the development of the Digital Twin either frontloading the development of the product (explorative engineering) or following its development (subsequent engineering). While the development of the Digital Twin with explorative engineering is not bound to restrictions by already existing systems, the subsequent engineering of the Digital Twin has to take the constraints given by already existing systems into account but may also reuse elements and knowledge from the development of these prior systems. In contrast to these approaches, joint engineering of Digital Twin and counterpart enables to incorporate joint design decisions. Another aspect we considered under the topic of Process is quality assurance. We identified mainly three kinds of quality assurance that can be either used alone or together to assure a high quality of Digital Twins. First, consistency checking can be used to validate the information the Digital Twin uses or produces. Moreover, simulations can be used as a verification technique. Finally, also, testing as a verification and validation technique is a good method to assure a Digital Twin’s quality. Apart from quality assurance, we also identified requirements specific to Digital Twins in our feature model. We identified that consistency requirement, which requires that the Digital Twin’s behavior matches the behavior of their real-world counterpart, are typical requirements of Digital Twins. Moreover, real-time capabilities of Digital Twins are often required when the Digital Twin may serve a specific purpose concerning its real-time counterpart, and therefore the Digital Twin must react to events in real-time. Finally, reuse is an own Digital Twin requirement, as a new development of a Digital Twin for each physical counterpart is often unnecessary if the Digital Twin or parts of the Digital Twin are reusable.

6.3. Deployment dimension

The deployment dimension supports the design decision to bring the Digital Twin into action and is characterized by its features shown in Fig. 27. To this end, this dimension is concerned with two closely related topics, hosting the Digital Twin in the real world and appropriate means for connecting it to its counterpart. Hosting is furthermore concerned with where the host is located, which could be the twinned system itself, a local server, or a cloud system. But hosting is also concerned with the kind of the host, as this may either be a data management system, a simulation, or even a virtual environment. When deciding about deploying a Digital Twin in the real world, it is also relevant how the Digital Twin is connected to its counterpart. While decisions about the connection of a Digital Twin are subject to its host location, various design options still exist. As such, a Digital Twin could be connected through a BUS, some other kind of local network, or even deploy Internet technology, such as respective protocols.

Decisions about the hosts’ location are alternatives. A Digital Twin does mostly not live simultaneously on a twinned system and a cloud platform. Decisions here may also be subject to the type of Digital Twin to be employed and its real-world counterpart. A Digital Twin that governs a whole factory is probably not located on a local machine; vice versa, a Digital Twin that controls and monitors a single machine, may rather be deployed on the machine itself than on a cloud platform. Design decisions here should be made with the purpose of the Digital Twin in mind. In contrast, Digital Twins can support multiple host types. For example, a Digital Twin can be part of a database management
system that also incorporates simulations for value updates or provides simulations as an alternative service. Furthermore, a Digital Twin deployed in a virtual environment could also function as a simulation of that twin. Finally, the connection domain provides multiple selectable options. A Digital Twin can be both connected to its counterpart through a bus and employed in a local area network.

While design decisions about the Digital Twin’s deployment are important, they are mostly subject to other concerns and the available environment. The purpose of the twin affects the host location, which then limits the available connection options.

6.4. Operation dimension

The operation dimension classifies the Digital Twin behavior while the Digital Twin is running. It specifically focuses on interaction with other systems, how the Digital Twin decides on next actions, and which kind of information it exchanges with peripheral systems. All characteristics in this dimension are optional, which means that they are not necessarily covered by all Digital Twins (see Fig. 28).

Horizontal communication encapsulates all Digital Twin communication with the main focus on information exchange where none of the involved partners can instruct another one to behave or change in a certain way. We distinguish between information exchange with PLM systems, which was frequently mentioned during our study, information exchange with the physical counterpart sharing, e.g., its current state, and even interaction with other Digital Twins. The feature decision-making specifies how the Digital Twin determines its next actions or the data that it exchanges. Machine Learning covers all Digital Twins that make predictions or decisions without being explicitly programmed only by evaluating provided data. The data mining feature characterizes Digital Twins that evaluate data sets and try to detect patterns. When Digital Twins imitate the physical world to decide on the best action, they have the simulation feature. For reasoning, we further classified Digital Twins in symbolic reasoning, numeric reasoning, and case-based reasoning. Digital Twins react to different input data and sources. These are covered by the Inputs and Events feature. Machine data can cover error logs or notifications of machines, sensor data, other IoT data, and general data about the environment in which the Digital Twin operates, e.g., temperature values in a production location. User specifications can either be given as direct control commands that are specified via a user interface, but some Digital Twins also evaluate human movements and gestures. Models and simulation results are also options for gaining knowledge about the operating context, the intended behavior, or the physical entities. The output feature describes Digital Twin outputs on a content level, so this feature does not characterize output formats or communication channels. Some Digital Twin reflect the current system state (the physical entity as it is), some Digital Twins plan how the physical entity should act in the future (the physical entity as it should be), and some Digital Twins predict the future behavior of the physical entity but do not influence it (the physical entity as it can be). Often, Digital Twins combine several of the described features to fulfill an information need or specifically optimize the underlying physical entity.

7. Threats to Validity

Our study is subject to threats to validity. In the following, we analyze and classify these according to Wohlin et al. (2012) as construct, internal, external, and reliability validity. Construct validity directly refers to the study’s overall design, such as search query or evaluation criteria. External threats restrict the generalizability of a study, while internal validity refers to the specificity, i.e., factors that influence the conclusions drawn from the readers. Reliability describes the trustworthiness of the study’s results.

Regarding threats to the construction of this mapping study, there are plenty of similar yet distinct terms for describing Digital Twins, Digital Shadows, Virtual Twins, etc. While some publications extensively distinguish between these terms, others use them interchangeably. To ensure an accurate mapping in our study, we have considered these terms as separate concepts per default. However, if an investigated publication switched the wording while clearly referring to the Digital Twin, we followed the paper’s intellectual roadmap and considered these as synonyms. Overall, this yields an accurate analysis result of the included papers. In contrast, publications using different terms only (e.g., constantly mentioning virtual copies without including the term of a Digital Twin) could not be recognized in this study, as it is impossible to extract whether the authors refer to the Digital Twin concept or explicitly distinguish from it. This topic could be addressed in a future study that explicitly includes all potential synonyms, thus covering a larger yet less precise scope.

A further threat to construct validity arises from our exclusion criteria during the initial screening of the papers, as it only considers title, keywords, and abstracts. This procedure could mistakenly exclude potentially relevant publications. To minimize this effect, we generally included papers for which we were uncertain and only excluded these in the classification phase when they turned out to be not relevant for our mapping study.

Another threat resulting from the design of our study is based on the classification of publications. In general, the categorization for several research questions is not disjoint, as a publication could be related to multiple dimensions. For instance, Digital Twins can use combinations of different techniques for decision making. This causes difficulties in evaluating the results since some dimensions may be highly interdependent. We designed the classification without overlapping and used existing classification schemas to minimize the threat and only allow multiple assignments if necessary.

Since our work is based on a literature study, it is inevitably subject to publication bias. Principally more successes and positive reports on a topic are published. This complicates assessing the areas that are not positively affected by Digital Twins or which concepts and methods for constructing them are not applicable. Furthermore, there may be research and material outside of common research distribution channels, i.e., grey literature, which must be handled specifically (Qi et al., 2021b). Further work on the analysis of the current status of Digital Twin research could focus on grey literature.

Our study is also affected by external validity in terms of generalizability. We selected a rather general search query to obtain a large corpus of publications. Including only online-available,
peer-reviewed, English publications (excluding short papers) reduces the corpus. This slightly affects generalizability, but at the same time, guarantees the accessibility of our results and the reliability of the study. As the investigated publications cover different domains and produce various findings, we cannot generally conclude that results from one problem domain apply to another. Therefore, our study elaborates on the relationships between the individual clusters to identify similarities as well as differences in a generalizable fashion.

We have used the search engines of leading scientific databases and libraries, such as SpringerLink, IEEE Xplor, ACM, WoS, and Scopus, for searching the literature. We intentionally excluded google scholar as a search engine, as it contains vast amounts of non-peer-reviewed publications (which are excluded during screening in any case). Furthermore, google scholar does not store any publications such that generally, most relevant publications are found as long as the related libraries are considered. Although this may negatively affect the external validity, it increases the reliability of the search results.

Regarding internal validity, the publications differ significantly in the level of detail in which they explain Digital Twins and their constituents. Authors often do not specify the exact system boundary of the Digital Twin, which impedes a precise mapping regarding relevant technologies. For instance, it is often obscure whether a cloud system is an integral part of the Digital Twin or whether the Digital Twin merely uses it for communication. To obtain an unambiguous mapping, we generally decided these cases in favor of the Digital Twin, attributing these properties and technologies to its realization. Although this may negatively affect the external validity, it increases the reliability of the search results.

A further threat to internal validity is the readers’ different previous knowledge, which may lead to classification discrepancies, e.g., through experience, more details can be anticipated. To minimize this effect as much as possible, we have collectively read the first 60 publications to synchronize our mapping.

The conclusions drawn from analyzing the included publications can influence the reproducibility and, thus, the study’s reliability. As mentioned for internal validity, we analyzed the publications in favor of the Digital Twin to ensure an unbiased evaluation of the different sources. Another research group might draw slightly different conclusions in particular circumstances. To add transparency and to ensure a reproducible study, we explained the research method and corresponding design decisions in detail (cf. Section 3).

8. Conclusion

Our survey has shown that Digital Twins are researched in many domains, including agriculture, construction, education, mining, transportation, and for a variety of purposes. Yet, the large majority of research on Digital Twins investigates individual (cyber–physical) systems in manufacturing. We could not detect a trend that research on Digital Twins is catching up in other domains, at least in terms of the number of publications. However, advanced Digital Twins are already being presented for domains beyond manufacturing. Often, research on Digital Twins focuses on monitoring the twinned system, as well as optimizing or predicting its behavior. Where research focuses on optimizing the twinned system, the Digital Twin often acts as an outer control loop that adapts the twinned systems behavior, i.e., both systems, the twin and the twinned system, form a larger, self-adaptive system point of view. Such often emit actions, commands, or plans that directly or indirectly (e.g., via another CPS management system) control the CPS’s behavior. Consequently, research on Digital Twins as-designed, as-operated, or Digital Twins addressing multiple lifecycle phases, is less common. Furthermore, current research also focuses on Digital Twins that are developed after the twinned system. Rarely, the Digital Twin and the twinned system are engineered together.

We also found relatively few research on combining AI methods with Digital Twins. Instead, to engineer and operate Digital Twins, a large variety of tools, e.g., for simulation, CAX, process management, visualization, data management, and model-driven development, are used. The produced Digital Twins consist of models, complex subsystems (e.g., databases or dashboards), plain GPL code, and sometimes even (mostly for augmented reality components) hardware parts.

Through our survey, we also have identified and organized central design decisions common to engineering Digital Twins. These include (i) requirements on the number of twinned counterparts, when the Digital Twins should be used and which lifecycle stage of the twinned system it should represent; (ii) realization decisions regarding implementation technologies, tools, and process; and (iii) deployment decisions on the Digital Twins hosting location and its connections to the twinned system. The feature models detailing these represent the state-of-the-art decisions to consider when engineering Digital Twins. We expect future Digital Twins development to contribute further decisions to the presented feature models. Yet, they can serve researchers and practitioners as a guidance when considering Digital Twins.

Based on our observations, we identified seven challenges for the future of Digital Twin research:

1. **Domain-specific Digital Twins (RQ-1.1).** The large body of Digital Twin research focuses on a single domain, primarily manufacturing, yet other domains employ technologies that can serve as an excellent foundation for further Digital Twin research.

2. **Composable Digital Twins (RQ-2.8).** Most Digital Twins found in our survey are build from scratch. The reliable combination and composition of Digital Twins is essential for their effective reuse. Different methods to support these processes are required. For instance, integrating the Digital Twin of a motor into the Digital Twin of a car...
may require another composition method than integrating the Digital Twin of a manufacturing device into the Digital Twin of a factory. For instance, building information modeling based on IFC (ISO 16739) in architecture and construction supports the integration of various concerns of Digital Twins and can be employed for many of the purposes found in our study.

(3) **Standardization (RQ-3.1).** Literature yields a wide continuum of systems considered Digital Twins by the authors from various domains. These range from high-fidelity simulation models to model-less software systems operating on the twinned systems and various combinations in-between. A future, in which Digital Twins (e.g., using Digital Twins as contract parts between OEMs and suppliers) can be exchanged, combined, and integrated, requires a common understanding of the concept. Currently, there is an ISO standard for Digital Twins in manufacturing\(^5\) in development that might at least harmonize the understanding of Digital Twins in that domain. Whether this standard will be compatible with the understanding in other domains needs to be evaluated and technological implementations on, e.g., exchange interfaces for Digital Twins, need to follow then.

(4) **Tool support (RQ-3.2).** While we have identified a large variety of tools employed to engineer and operate Digital Twins, we found very few tools specifically tailored to Digital Twins. While there are some tools mentioned in literature, such as Amazon Greengrass\(^6\), Eclipse Vorto\(^7\), or Microsoft’s Digital Twin Definition Language\(^8\), these largely focus on data structure modeling and data exchange for Digital Twins but do not cover the full spectrum of modeling concerns.

(5) **Modeling support (RQ-3.2).** Abstraction is the key to understanding and improving CPSS. Consequently, models are essential to Digital Twins. This is not limited to software engineering models, but includes CAD models, mathematical models, physical models, simulation models, and many more. However, modeling methods developed by software engineers are also used by professionals without formal software engineering training. Therefore, software engineering must provide methods to integrate, analyze, and transform models used in research and practice so that they can be used without software engineering background.

(6) **Quality assurance and requirements (RQ-3.4).** Digital Twins are subject to common expectation, such as to high-fidelity representation of the twinned system. Yet, we found few research on quality assurance and requirements for Digital Twins. Hence, it currently is hardly possible for a Digital Twin to fail in fulfilling requirements on it. For instance, it is left to investigate how much the fidelity of a Digital Twin may degrade before its not a (useful) Digital Twin anymore. While the feature models presented in this paper can be a starting point for exploring such requirements, these also need to build on a common understanding of the concept of Digital Twins in general.

(7) **Tool selection support (RQ-3.2).** An incredible variety of methods and technologies are used in the development of Digital Twins. Identifying which methods and technologies are suitable for which challenges, requirements, and Digital Twins goals would facilitate advancing the state-of-the-art in Digital Twins. To this end, the employed methods and tools used in engineering Digital Twins need to be cross-tabulated against the purposes these Digital Twins. Such research could result in a design catalog of technologies to achieve certain effects with Digital Twins.

To improve our insights into the software engineering for and use of digital twins, future studies on the topic should consider the evolution of concerns, tools, and methods across time. Moreover, with Digital Twins increasingly being deployed in various industries, considering including patents or gray literature from industry might yield valuable insights as well.

Overall, the study presented in this paper sheds light on the state-of-the-art in Digital Twins and on the concerns related to engineering and operating these for future research to build upon our results and for practitioners to guide their work.

**Declaration of competing interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgments**

This work has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC-2023 Internet of Production – 390621612, by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development (CDG), and by the German Federal Ministry of Economic Affairs and Climate Action (BMWK, Bundesministerium für Wirtschaft und Klimaschutz) under grant no. 19S21002L.

**Primary studies**


Block, C., Kuhlenkötter, B., 2019. Digital factory implementation approach starting from the macroscopic perspective with an example for holistic planning in assembly systems. In: Advances in Production Research.


Schirmann, M., Collette, M., Gose, J., 2018. Ship motion and fatigue damage
Ruohomäki, T., Airaksinen, E., Huuska, P., Kesäniemi, O., Martikka, M.,
Ríos, J., Morate, J., Roßmann, J., 2019. Industry, 4.0 beyond the factory:


