Conception and Implementation of a Digital Twin using an Enterprise Knowledge Graph Platform

David Jilg and Nico Stuckart

Business Information Systems II, University of Trier, 54286 Trier, Germany
{s4dajilg,s4nistuc}@uni-trier.de

Abstract. TODO (250)Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, diam sed nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, diam sed nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse

Keywords: Digital Twin · Knowledge Graph · Ontology · Industry 4.0.

1 Introduction

Production facilities are becoming more and more autonomous and are constantly developing through the integration of the physical and virtual space with enabling technologies like cyber-physical systems (CPS), Internet of Things (IoT) and cloud computing [38]. This revolution, which is particular visible in the manufacturing sector, has been coined by the term Industry 4.0 [24]. The Industry 4.0 offers possibilities to completely novel manufacturing models on the basis of smart manufacturing, meaning that physical processes are monitored and controlled by digital systems [9]. Smart manufacturing however requires deep integration between the digital and the physical space. One of the most promising enabling paradigms that have emerged for realizing smart manufacturing in the Industry 4.0 are digital twins [35].

There are many ways to model a digital twin. Besides formats such as AutomationML, UML [4] or classical CAD-based approaches, especially semantic information models are a promising way to formally represent the digital twin in a machine understandable way [15]. Using such semantic technologies, e.g. ontologies, it is possible to establish semantic links between different entities, like machines or data entities.

A technology that may be used to develop such semantic knowledge bases and additionally provide a lightweight access point to the information are knowledge graphs [15]. Using knowledge graphs provides advantages such as the integration of different data sources that may be queried via a unified interface as well as convenient user experiences [19]. Furthermore, using semantic search patterns in visual editors allows for user-friendly querying [19], without requiring the users to have in-depth knowledge about the underlying semantic technologies, such as RDF.

Considering this, knowledge graphs offer many benefits for hands-on usage of semantic technologies. In turn, using semantic technologies also provides a promising approach to establish reusable, machine understandable digital twins. Therefore, the aim of this work is to prototype a digital twin based on a knowledge graph in an experimental use case and to assess the resulting benefits. To do so, first a specific use case within the manufacturing context is selected and the resulting requirements towards the digital twin are derived. Thereafter, existing approaches and standards for modelling digital twins are considered in the conceptualization phase before the use case-specific digital twin that is subsequently implemented in a knowledge graph platform. Finally, the concept and the implementation are evaluated and a closing conclusion on the suitability of knowledge graph platforms for the implementation of digital twins is drawn.

2 Foundations and Related Works

In the following, the necessary foundations are introduced and related scientific works in the domain of semantic digital twins and their modelling are presented.

2.1 Digital Twins

The concept of digital twins was first introduced at the University of Michigan in the year 2003 [18]. To this day, there is no unique definition of this concept in the scientific literature [32]. Lee et al. define the digital twin as a dynamic digital representation of an asset that simulates its status based on real time data as well as other available knowledge [27]. digital twins can be used to collect data through different production stages and use that data to discover failure, causes, streamline supply chains, enhance the production efficiency [35] and predict required maintenance actions during a machine's lifetime [15]. Furthermore, digital twins may be used to aid the development of smart manufacturing systems which is especially important in the scope of Industry 4.0 since it is one of its main principles [25].

2.2 Knowledge Graphs

Although the ideas of the concept reach back to 1972 [34], the term was significantly coined by the introduction of knowledge graphs in the Google search in 2012 [16]. There are a variety of definitions and descriptions that have emerged for this term [11], but knowledge graphs tend to describe real world entities and their relations in a graph while defining possible classes and relations [33]. Knowledge graphs therefore can also be built in accordance to standard approaches such as the Resource Description Framework (RDF) [31], which can provide better opportunities to automatically process the data and share it with other stakeholders. Knowledge graphs can be built and accessed using knowledge graph platforms that provide user-friendly ways to query and modify the graph [19].

2.3 Asset Administration Shell

A standardization approach that may be used for the implementation of a digital twin is the Asset Administration Shell (AAS) [5]. The AAS aims at integrating several Industry 4.0 components and achieving cross-company interoperability, since all machines, products and devices, even non-intelligent products can be represented. By integrating the asset as an addressable object in the Industry 4.0 network, controlled access via a standardized interface is ensured [5]. In the context of this paper, it is especially of interest, that the AAS may be set up as an implementation form of a digital twin.

The AAS approach proposes that information about the assets are grouped into submodels which aggregate the available information. For each detail of an asset a separate submodel is defined in the AAS. Submodels may either refer to functionalities of the asset, e.g. drilling or milling of an industry machine, or general aspects such as condition monitoring, energy efficiency or identification [5]. These submodels again are a standardized way to express the information about the submodel domain with the aim that different components may implement the same submodel if they share the functionality or general aspect. If, for example, a submodel represents a functionality that different components provide, these components may implement this submodel which ensures comparability between the components in a standardized manner. Proposed data formats of the AAS are XML and JSON for serialization of the AAS and RDF to enable the advantages of semantic technologies [5].

2.4 Experimental Environment

For the implementation, an experimental production environment is used to gather the required data for a specific asset and simulate an Industry 4.0 scenario.

The Fischertechnik factory model [22] is a simulation factory that emulates a manufacturing environment. With its different workstations and modules, e.g. ovens, milling machines, sorting machines or a high-bay warehouse, it is used to receive confidential data within a manufacturing scenario for research purposes. Furthermore, the model is equipped with components such as light barriers, switches, compressors, valves and multiple sensors which allow the factory to be used for various research applications, such as predictive maintenance [23] or video-based monitoring of manufacturing processes [30]. Simplified, within each simulated process cycle, workpieces are passed to the factory and are processed by an oven and a milling machine, before they pass a sorting machine, which sorts them by color. After that, they are transported and stored in a warehouse. Building on the Fischertechnik factory model, an ontology was developed which semantically models the factory, called "FTOnto" [22].

2.5 Related Works

Despite the large number of publications in the context of digital twins, no uniform modelling concept has yet achieved acceptance [35]. Especially in the context of semantic digital twins, the number of publications dealing with this domain is still rather small. Boschert et al. present an approach trying to utilize knowledge graphs for comprehensive sharing of information along all life cycle phases of a digital twin [7]. Another work in this domain is [15], where knowledge graphs are used for the storage of data from different workstations which is then used by the digital twin. Furthermore, this work proposes a five-step methodology for developing a digital twin based on knowledge graphs. However, this methodology does not primarily target the semantic modelling of the digital twin.

Within the domain of semantic AAS [17] proposes an approach to semantically model Industry 4.0 components in a AAS. Their approach examines the advantages of semantics, such as interoperability achieved by using RDF, using URI for global identification or using SPARQL to query the data comprised in a semantic AAS [17]. However, the exemplary translation of a standard for electrotechnical components into a semantic class hierarchy is not applicable to the use case with the Fischertechnik since the available components and data are of different structure. Another approach that highlights the idea of semantic AAS is [3], where a mapping from the AAS XML serialization to an RDF serialization is presented. However, to the best of the authors' knowledge, no prior research specifically deals with using a managed knowledge graph platform for the implementation of a digital twin or an AAS. To address this research gap and to examine the potentials and limitations, in the following a custom methodology is introduced to semantically model and implement a digital twin as an AAS in a knowledge graph platform. This approach is then applied and evaluated in a prototypical implementation thereafter.

3 Use Case and Requirements

The following chapter introduces some general application domains for digital twins in manufacturing to frame the context of a specific use case for the prototype that is defined thereafter. Based on this use case, a set of requirements is specified that is placed towards the digital twin. As previously indicated, no common framework or methodology yet exists to model and develop (semantic) digital twins [35]. Therefore, in the following, a custom approach is applied, which is described in figure 1. First, use-case specific requirements are defined towards the digital twin, before the digital twin is semantically modelled considering standardization approaches such as the AAS. Next, it needs to be specified, in which way the digital twin may utilize and extend existing knowledge structures in the chosen implementation platform, before the final implementation is realized in an exemplary knowledge graph platform.

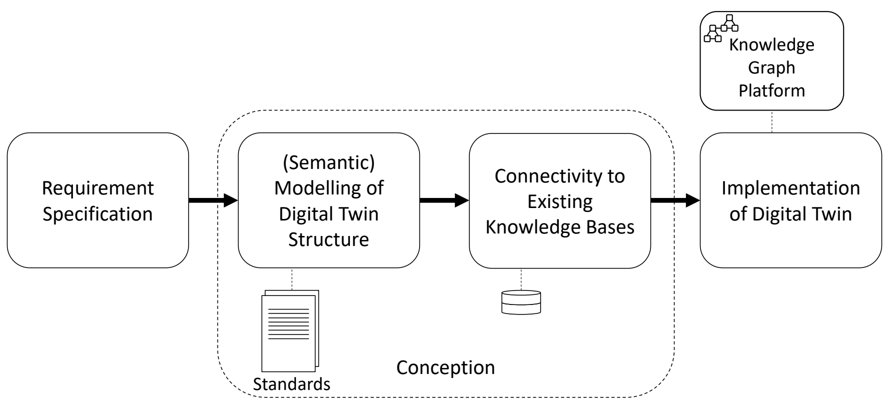

Fig. 1: Methodology for Modelling a Semantic Digital Twin

3.1 General Use Cases

Literature proposes several use cases for digital twins. Some of them are introduced below, before one specific use case is selected for the prototypical implementation in this paper. Since the use cases are so wide-ranging, we will limit ourselves to the domain of maintenance and describe some relevant approaches from it.

Within their literature review, Errandonea et al. distinguish maintenance digital twins regarding the maintenance strategy they support: starting from the basic reactive maintenance, ranging via preventive, condition-based and predictive maintenance to a highly intelligent prescriptive maintenance approach. Some of the functionalities that a digital twin can provide in such a maintenance scenario are, for example, predicting the asset condition in order to predict the maintenance plan or, within preventive maintenance, to reduce the number of maintenance activities needed [14]. Summarizing potential functionalities that a digital twin may include, Tao et al. propose a classification of nine different service categories [36]. Especially the following three are in the closer scope of the chosen top level domain of (physical) maintenance:

1. **real-time state monitoring**
   Sensors and communication between the physical and the virtual asset allow for real-time updates of the status. The data being updated may contain, for example, information about energy consumption, setting data or wear status of different parts [36].
2. **product failure analysis and prediction**
   By providing different information, the digital twin can also be linked to failure prediction analyses. These results again can be included in the digital twin to make this information available for the stakeholders [36].
3. **product maintenance strategy**
   The digital twin may use the information from failure analyses to detect faulty parts and based on this provide maintenance information, e.g. which parts need to be replaced [36].

Building on these nine service categories, Ciminio et al. further summarize the functionalities of digital twins into two categories:

1. **"DT that offer services connected to state monitoring, energy consumption, maintenance support and data analysis for optimization"** and
2. "DT whose main services are visualization of the behavior of the system or offering user guidelines"[10].

Regarding this binary classification, the maintenance use case selected for this paper may be mapped to the first category, as it combines services such as state monitoring and maintenance support.

3.2 Specific Use Case

Due to the external preliminaries, which are taken as given, the use case within the maintenance domain is constructed and described in the following. The Fischertechnik simulation model and its components provide several options that may be used for building a digital twin for maintenance purposes. As this paper is intended to show a proof-of-concept implementation, particularly two machine categories of the Fischertechnik model are selected for which the implementation is carried out: ovens (OV_1, OV_2) and sorting machines (SM_1, SM_2). Moreover, since the goal is to use reusable standardization approaches, the digital twin shall especially model aspects of the selected components, that both have in common. Therefore, the mechanical wear of the components is to be modelled and implemented as a "condition monitoring" submodel (according to the AAS) in this paper. Summarizing, the use case of a maintenance digital twin is constructed as follows: the Fischertechnik factory model is used as an exemplary production environment, from which the components, two ovens and two sorting machines, are chosen. For all of these components a maintenance scenario is created, which especially shall provide information about the mechanical wear of these components.

3.3 Requirements

Within such a production environment, there are potential stakeholders that have different views and requirements towards a digital twin. Such stakeholder are, for example, manufacturers of the asset, operators or maintenance providers during ongoing operations. All of these might be interested in the wear status of a machine for different purposes. These requirements are considered in the following list of requirements. However, as semantic digital twins in the context of knowledge graphs is a fairly novel research direction, literature from related domains, such as semantic technologies, is considered to define some more general requirements that the digital twin should meet. Within the domain of Industry 4.0 Malburg et al. suggest multiple requirements, that a service oriented architecture (SOA) should meet in order to be suited for AI-based research in the Industry 4.0 context [29]. Considering both, aforementioned literature on semantic technologies, as well as use-case specific stakeholder requirements that represent their information interests, the following compilation of requirements is placed towards the digital twin. The requirements are specified according to the ISO 29148 [1] using the terms "shall" for mandatory requirements and "should" for non-mandatory, but desired requirements.

R1 The digital twin shall provide interoperability and interconnectivity [29].
R2 The digital twin shall connect to existing knowledge representations [29].
R3 The digital twin shall enable ontology and knowledge base updates [29].
R4 The digital twin shall be enriched with semantic descriptions [29].
R5 The digital twin shall model relationships between services and manufacturing components [29].
R6 The digital twin shall model the mechanical wear of the oven and the sorting machine by using a reusable, standardized approach.
R7 The digital twin shall enable querying information about the mechanical wear of the component without requiring specific domain knowledge about the modelling of mechanical wear.
R8 The digital twin should be able to predict the status of mechanical wear of the machine after a specified amount of workpieces is produced.
R9 The implementation shall allow users to modify and extend the digital twin without requiring RDF knowledge.

These defined requirements do not only serve as a guideline in the following conceptualization and implementation phase, but also provide a means for a final evaluation of the proof-of-concept implementation afterwards.

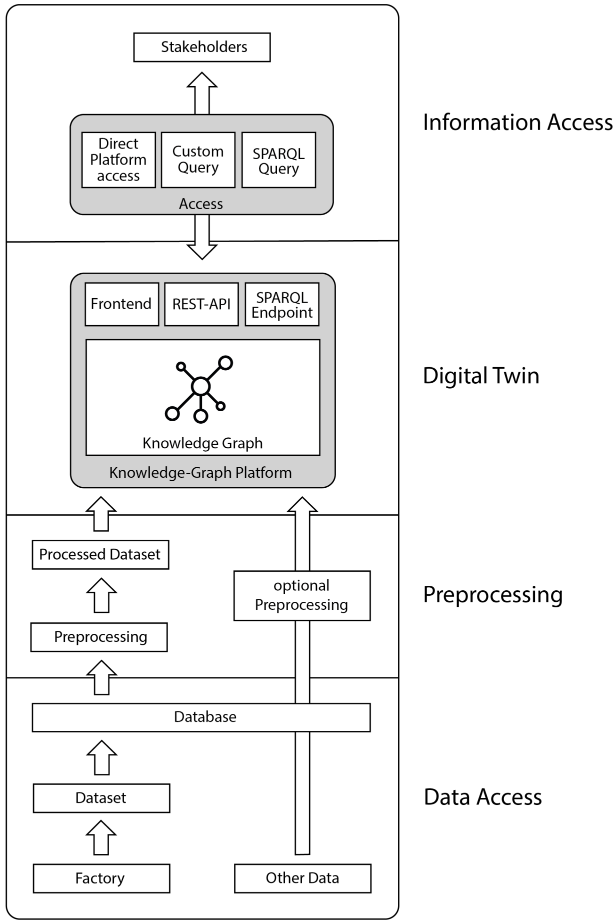

Fig. 2: Digital Twin Infrastructure

4 Semantic Digital Twin based on a Knowledge Graph

In this chapter, the approach of building a semantic digital twin based on a knowledge graph is explained. Furthermore, the prototypical implementation of the digital twin is described.

4.1 Digital Twin Concept

Although there is agreement in the literature about the basic potential of digital twins, concrete standards and frameworks for the implementation have not yet achieved establishment [35] due to missing harmonization and integration of different approaches [26]. Therefore, in the following a general, conceptual framework is proposed that describes the information access and the data sources of the digital twin.

Figure 2 visualizes the conceptual idea of the digital twin and its interfaces. The core of the digital twin is the implementation which is set up in a knowledge graph platform with two directions of data flow: first, it needs to specified how the data gathered from a data source needs to be prepared to feed the knowledge graph platform and second, it needs to be defined in which way stakeholders may use and query the digital twin in a lightweight manner. The proposed methodology illustrated in figure 2 shows a layer for data preprocessing, that prepares the raw data produced by the data source, e.g. a production facility like the Fischertechnik factory.

The preprocessing addresses use cases where aggregated metrics need to be calculated from the raw data that go beyond the capabilities of knowledge graph platforms. To the best of the author's knowledge, the general features of knowledge graph platforms to perform more complex preprocessing, such as calculating advanced attributes from imported data, are rather limited (i-views [13], Metaphactory [19], Amazon-Neptun [2]).

The preprocessing step may be used to filter the data or calculate required indicators, which are then fed into a database afterwards. The database, in turn, represents the direct interface to the digital twin, which the knowledge graph operates on. The digital twin itself, i.e. the implementation within the knowledge graph platform offers different access points to the end users. It can either be used directly via the frontend, the graph may be queried via a SPARQL endpoint or the knowledge graph may be used via a Rest-API. By these options different needs of different stakeholders are met: the system may be integrated into existing systems by using the Rest-API or the knowledge graph platform is used directly, both options offering possibilities to query predefined submodels and key figures.

Fischertechnik Data Set The data set that is used for the prototypical implementation of the digital twin is generated by running the Fischertechnik factory model for two hours. By using a fault simulation for the chosen components (ovens and sorting machines) different degradation types of the wear may be simulated to showcase a condition monitoring submodel reflecting the wear of the components in the digital twin. During the two hours of running the wear of the two ovens and the two sorting-machines is simulated by slowly reducing the speed of the motors of these machines. In case of the sorting machines the speed of the motors that drive the conveyor belts is lowered after every workpiece that is processed. For the ovens the speed of the motors that transport the workpieces inside the oven for processing and outside of the oven after processing are reduced. In order to simulate different kind of degradation processes the reduction of speed of the motors is altered between OV_1 and OV_2 as well as between SM_1 and SM_2. For OV_1 and SM_1 a linear degradation is simulated by lowering the motor speed by a fixed value for every workpiece that is processed. In contrast to that, the motor speed of the OV_2 and SM_2 is lowered with an exponential function to simulate accelerated wear.

4.2 Implementation

For the evaluation of the previously described concept a prototypical implementation is used. Due to time constraints in the corresponding research project the prototype is limited to the two ovens and two sorting machines of the Fischertechnik factory model. The necessary data is provided by the previously described data set which contains information to approximate the simulated wear of the machines. To implement the digital twin the "Empolis Intelligent Views Platform"(i-views) [13] in the version 5.3 is used, which is provided by the Empolis Information Management GmbH [12] for the research project. I-views is an enterprise knowledge graph platform that allows the users to import data from different sources and link the information therefore creating a knowledge graph [13].

Data Preprocessing The preprocessing in our adapted approach is done by multiple Python-scripts which iterate through the data set for each machine and calculate new attributes. The table 1 shows the attributes that are calculated during the preprocessing of the data concerning the first oven of the factory model.

Attribute	**Description**
combined transport time	The combined transport time of a workpiece that is calculated by the determine the sum of the inbound and outbound transport time.
average combined transport time	The average combined transport time of the last five work pieces.
number of parts	The number of parts a machine has processed since the last maintenance.
wear type	The "type" of wear. This attribute can be either "stable" or "increasing". If the attribute is "stable" the wear increases by the same amount with every work piece (linear degradation). If the attribute is "increasing" the wear every work piece causes is increasing (e. g. exponential degradation).
wear rate/factor	If the wear type is "stable" this attribute contains the amount of wear that occurs everytime an workpiece is processed. In the case of the oven this is the amount of milliseconds that the transport time increases. If the wear type is "increasing" this attribute contains the factor by which the wear is increasing.
remaining parts	This attribute contains an estimation of how many part the machine can still process before the wear reaches a critical level.

Table 1: Calculated Attributes for OV_1

During preprocessing the first attribute that is calculated for every precessed workpiece is the "transport time". This time is used to approximate the speed of the motor of every machine since the motor speed is not captured in the dataset directly. In the case of the two ovens this time is calculated by combining the time it takes to transport the workpiece into the oven for processing and back out again. We use the calculated transport time to determine the new attribute "wear type". The wear type shows if the degradation is stable (e.g. linear degradation) or accelerating (e.g. exponential degradation). Additionally, we use the wear type and the change in transport times to calculate the remaining life time of the machine by simulating how the degradation would progress if processing additional workpieces.

Digital Twin Structure The first step in setting up the digital twin is to import the already existing FTOnto ontology [22] that semantically describes the Fischertechnik factory model into i-views. In addition to that the the mason [28] and the SOSA [20] ontologies have to be imported since the FTOnto ontology relies on some of their classes and relations. I-views provides an RDF-import feature to do this, however i-views does not support all namespaces and constructs used in these ontologies, which is why again some preprocessing is required (e.g. transforming the "Web Ontology Language 2" (OWL 2) construct "owl:NamedIndividual" to "rdf:Description, since i-views does not support OWL 2 constructs). The imported ontologies are used to provide extending information about the assets that are modeled in the digital twin.
After importing the ontoglogies, the needed relations, attributes and object types are created. Object types in i-views are like classes in ontology's. They can be used to create instances of these types with specific property's. Theese propertys can for example be the allowed relationships and attributed that a instance can have. For the prototype five main object types with additional sub types are created. The created types are "digital twin", "Asset Administration Shell", "Data", "Metadata" and lastly "Submodel". The type "digital twin" is used to represent the top level object for the entire digital twin and each top level object for the sub digital twin of every machine. Since each digital twin for a machine includes a asset administration shell, a object type is needed to represent this. The object type "Data" is used to store the imported dataset entry's while the "Metadata" type is used to provide metadata for different components of the digital twin. The last type created for the digital twin is the "Submodel" type which is used as a container for a sub model of an asset administration shell. To connect the instances of these object types multiple types of relationships are defined. For example the relationship type "has AAS" is used to connect the object instance representing a digital twin for a machine with the corresponding asset administration shell instance. To prevent possible mistakes in modelling the digital twin i-views restriction features are used to restrict the use of theese relationship types. For example the previously mention "has AAS" relationship type can only be used to connect instances of the object types "digital twin" and "Asset Administration shell". In addition to that the relationship type can only be used once on every single object assuring that no digital twin can acciden-

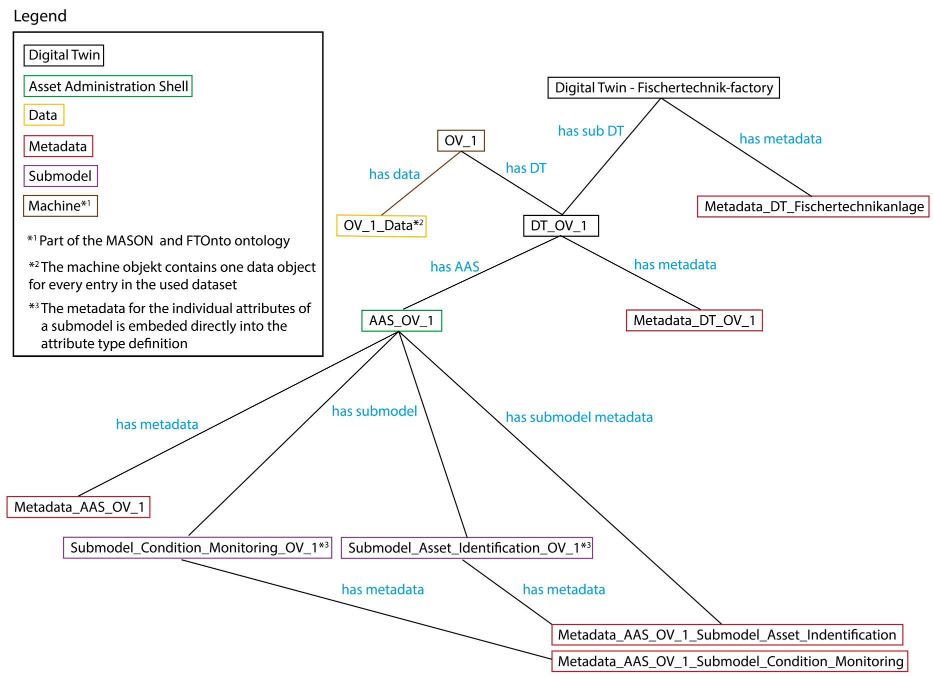

Fig. 3: Overview of the prototype structure

tally have multiple asset administration shells. Restrictions are also used in the defined attribute types. The attribute types define the attributes that instances of a specific object types are allowed to have.
The next step after setting up all object, relationship and attribute types and their properties and restrictions is to create the actual digital twin structure. Figure 3 shows an overview of the digital twin structure. Due to the size of the prototype the overview is limited to the first oven of the factory model ("OV_1") and all relationships are only shown in one direction. As can be seen in the figure the digital twin uses a hierachical structure with a top level instance of the "digital twin" type as a representetive. Every asset of the digital twin has a "sub-digital twin" which is represented by another instance of the "digital twin" object type ("OV_1" is the digital twin instance for the first oven). This instance is linked to the machine instance that represents the asset in the FTOnto onotlogy. The data that is imported is also linked to that instance.

As previously mentioned the structure of the developed concept is heavily based on the AAS [5]. The prototype is modeled by refering to the recommendations of the discussion paper "Verwaltungsschale in der Praxis"[8]. Every digital twin instance of an asset therefore contains a Asset Administration shell instance and a metadata instance. The Asset Administration shell instance contains the submodels used to describe the asset and its current state. The submodels share the same structure regardless of the asset they are describing. This prorotype only contains two submodels. The "Asset Indetification" submodel can be used to uniquly identifiy the asset and the "Condition Monitoring" submodel can be used to asess the wear of the asset.

For the sake of querying the information of a condition monitoring or asset identification submodel there are predefined queries. The potential stakeholders could access these predefined queries either by getting direct access to the i-views backend and using the GUI to run the query or by using the Rest-API. When using a knowledge graph platform with a SPARQL-Endpoint [37] the stakeholder could also use a SPARQL-Query to search for information in the knowledge graph. Aside from the predefined querys it is also possible to define custom queries by either using the GUI or the Rest-API.

To encourage further research on semantic digital twins using knowledge graph plattforms, the structure of the digital twin was exported in RDF format and made available in a Github repository [21].

5 Evaluation

In the following section, the previously defined requirements (section 3) are used to evaluate the implemented digital twin. The conceptualized and implemented digital twin provides interoperability and interconnectivity (R1) between different ontologies as well as between potential different digital twins due to the usage of the standardized AAS. Furthermore, by its design the digital twin is extending the existing FTOnto [22] ontology (R2). It is also possible to connect the digital twin to a database from which the digital twin, i.e. the i-views platform, is importing data and based on that updates its values (R3). Due to the fact that the digital twin is built upon RDF each component is enriched with metadata and semantically annotated information that describes the different values (R4). The underlying model of the Fischertechnik factory is semantically described in the FTOnto [22], which provides and models multiple services and components available in the factory. The digital twin was built on this basis and extends the ontology, again modelling the relationships between the different components of itself as well as to the related components of the FTOnto (R5). In the prototypical conception and implementation the mechanical wear was modelled as a submodel, which is implemented by the ovens and the sorting machine, while using the standardized modelling approach of the AAS (R6). This wear may be queried either using the i-views platform or, for example, its Rest-API to access predefined queries or by using custom queries.

The results show that the different stakeholders might pose their query about the wear of the machines without needing any deeper domain knowledge about the modelling of the wear (R7). Furthermore, regarding a more simulation-oriented perspective, the digital twin is able to forecast when a critical failure of the machine is likely to occur due to its wear (R8). This capability is used to calculate how many workpieces a given machine can still process before maintenance is required. This information can be retrieved by the stakeholder by using the predefined query for asset wear information. Moreover, due to the choice of a knowledge graph platform users can modify and extend the knowledge graph via the i-views frontend without requiring knowledge about the underlying technology R9.

6 Summary and Outlook

In summary, the introduced approach of using knowledge graph platforms for the digital twin implementation offers many benefits. Knowledge graph platforms provide a lightweight access to semantic technologies without requiring deep technological knowledge [19]. Furthermore, the resulting knowledge graph is machine-readable and the semantic annotations support comprehensibility. Due to the platform features it is also possible to use different permissions and views to differentiate with regard to different stakeholders. Combining these characteristics with standardized frameworks such as the AAS, reusable components and submodels may be created that together avoid redundant data while enhancing the aforementioned benefits even more.

However, always depending on the specific use case, the knowledge graph platform should be suited for representing the specific needs. If, for example, the use case relies on time-series data, the selected tool should offer proper methods to include and represent such data. Potential future work for the chosen use case could be to switch from offline sensor data to online sensor data by using the proposed Rest-API. Contrary to the approach used in the prototypical implementation, where the data is stored in the knowledge graph itself, in a real-life setting it is more suitable to store the data in an external database. This keeps the knowledge graph slim while at the same time being able to retrieve the same knowledge. Moreover, the preprocessed calculation of the wear indicators, which is chosen because of limited features of the software, might be disadvantageous as stakeholders can not verify or modify the calculation within the knowledge graph itself.Furthermore, within the prototypical implementation only two submodels (asset identification and condition monitoring) were developed, while there are still many other potential use cases for submodels. An exemplary, related submodel that could build on the defined condition monitoring submodel could be in the domain of fault diagnostics. Such a fault diagnostics submodel could diagnose fault patterns and propose solutions, for example by connecting a case based reasoning system that stores previously collected error cases and solutions [6]. Additionally, the calculation that is currently used to determine the wear rate could be extended by using simulation methodologies to predict future conditions of the asset.

In conclusion, to further evaluate on the proposed approach it could be implemented in a more complex IoT scenario, for example by extending it to the whole Fischertechnik factory.

References

1. ISO/IEC/IEEE Draft Standard for software and systems engineering – Life cycle processes – Requirements engineering. ISO/IEC/IEEE P29148 First Edition,

August 2011 (2011). https://doi.org/10.1109/IEEESTD.2011.5966382
2. Amazon Web Services, Inc.: Amazon Neptune. https://docs.aws.amazon.com/neptune/latest/userguide/intro.html, [Online; accessed 29-August-2021]
3. Bader, S.R., Maleshkova, M.: The semantic asset administration shell. In: Acosta, M., Cudré-Mauroux, P., Maleshkova, M., Pellegrini, T., Sack, H., Sure-Vetter, Y. (eds.) Semantic Systems. The Power of AI and Knowledge Graphs. pp. 159–174. Springer International Publishing, Cham (2019)
4. Bao, Q., Zhao, G., Yu, Y., Dai, S., Wang, W.: Ontology-based modeling of part digital twin oriented to assembly. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture **0**(0), 0954405420941160 (0). https://doi.org/10.1177/0954405420941160, https://doi.org/10.1177/0954405420941160
5. Barnstedt, E., Bedenbender, H., Billmann, M., Boss, B., Clauer, E., Fritsche, M., Garrels, K., Hankel, M., Hillermeier, O., Hoffmeister, M., Jochem, M., Koziolek, H., Legat, C., Mendes, M., Neidig, J., Sauer, M., Schier, M., Schmitt, M., Schröder, T., Ziesche, C.: Details of the asset administration shell. part 1 – the exchange of information between partners in the value chain of industrie 4.0 (version 1.0) (11 2018)
6. Bergmann, R., Grumbach, L., Malburg, L., Zeyen, C.: Procake: A process-oriented case-based reasoning framework (09 2019)
7. Boschert, S., Heinrich, C., Rosen, R.: Next generation digital twin (05 2018)
8. Bundesministerium für Wirtschaft und Energie: Verwaltungsschale in der Praxis. https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/2020-verwaltungsschale-in-der-praxis.html, [Online; accessed 15-August-2021]
9. Chen, G., Wang, P., Feng, B., Li, Y., Liu, D.: The framework design of smart factory in discrete manufacturing industry based on cyber-physical system. International Journal of Computer Integrated Manufacturing **33**, 1–23 (12 2019). https://doi.org/10.1080/0951192X.2019.1699254
10. Cimino, C., Negri, E., Fumagalli, L.: Review of digital twin applications in manufacturing. Computers in Industry **113**, 103130 (12 2019). https://doi.org/10.1016/j.compind.2019.103130
11. Ehrlinger, L., Wöß, W.: Towards a definition of knowledge graphs. In: SEMANTiCS (Posters, Demos, SuCCESS) (2016)
12. Empolis Information Management GmbH: Empolis Information Management GmbH. https://www.empolis.com/en/, [Online; accessed 15-August-2021]
13. Empolis Information Management GmbH: Empolis Intelligent Views Platform®. https://i-views.com/en/empolis-intelligent-views-platform/, [Online; accessed 15-August-2021]
14. Errandonea, I., Beltrán, S., Arrizabalaga, S.: Digital twin for maintenance: A literature review. Computers in Industry **123**, 103316 (12 2020). https://doi.org/10.1016/j.compind.2020.103316
15. Gómez-Berbís, J.M., de Amescua-Seco, A.: Sedit: Semantic digital twin based on industrial iot data management and knowledge graphs. In: Valencia-García, R., Alcaraz-Mármol, G., Del Cioppo-Morstadt, J., Vera-Lucio, N., Bucaram-Leverone, M. (eds.) Technologies and Innovation. pp. 178–188. Springer International Publishing, Cham (2019)
16. (Google), A.S.: Introducing the Knowledge Graph: things, not strings. https://blog.google/products/search/introducing-knowledge-graph-things-not/ (2012), [Online; accessed 19-April-2021]

17. Grangel-González, I., Halilaj, L., Coskun, G., Auer, S., Collarana, D., Hofmeister, M.: Towards a semantic administrative shell for industry 4.0 components (02 2016). https://doi.org/10.1109/ICSC.2016.58
18. Grieves, M.: Digital twin: Manufacturing excellence through virtual factory replication (03 2015)
19. Haase, P., Herzig, D., Kozlov, A., Nikolov, A., Trame, J.: metaphactory: A platform for knowledge graph management. Semantic Web **10**, 1–17 (06 2019). https://doi.org/10.3233/SW-190360
20. Janowicz, K., Haller, A., Cox, S.J., Le Phuoc, D., Lefrançois, M.: Sosa: A lightweight ontology for sensors, observations, samples, and actuators. Journal of Web Semantics **56**, 1–10 (2019). https://doi.org/10.1016/j.websem.2018.06.003, https://www.sciencedirect.com/science/article/pii/S1570826818300295
21. Jilg, D., Stuckart, N.: Digital twin using a knowlege graph. https://github.com/DavidJilg/Digital-Twin-using-a-Knowlege-Graph/tree/main/Digital%20Twin%20-%20Rdf%20Export (2021)
22. Klein, P., Malburg, L., Bergmann, R.: FTOnto: A Domain Ontology for a Fischertechnik Simulation Production Factory by Reusing Existing Ontologies (10 2019)
23. Klein, P., Weingarz, N., Bergmann, R.: Enhancing Siamese Neural Networks through Expert Knowledge for Predictive Maintenance. In: IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning. Communications in Computer and Information Science, vol. 1325, pp. 1–16. Springer International Publishing. (2020). https://doi.org/10.1007/978-3-030-66770-2_6, http://www.wi2.uni-trier.de/shared/publications/2020_ECML-IoTStreams_SiameseNeuralNetwork-for-PredictiveMaintenance_Preprint.pdf, the original publication is available at www.springerlink.com
24. Koh, L., Orzes, G., Jia, F.: The fourth industrial revolution (industry 4.0) : technologies' disruption on operations and supply chain management. International Journal of Operations and Production Management **39** (11 2019). https://doi.org/10.1108/IJOPM-08-2019-788
25. Lasi, H., Fettke, P., Kemper, H.G., Feld, T., Hoffmann, M.: Industrie 4.0. WIRTSCHAFTSINFORMATIK **56**, 261–264 (08 2014). https://doi.org/10.1007/s11576-014-0424-4
26. Lattanzi, L., Raffaeli, R., Peruzzini, M., Pellicciari, M.: Digital twin for smart manufacturing: a review of concepts towards a practical industrial implementation. International Journal of Computer Integrated Manufacturing (April 2021). https://doi.org/10.1080/0951192X.2021.1911003
27. Lee, J., Lapira, E., Bagheri, B., an Kao, H.: Recent advances and trends in predictive manufacturing systems in big data environment. Manufacturing Letters **1**(1), 38–41 (2013). https://doi.org/10.1016/j.mfglet.2013.09.005, https://www.sciencedirect.com/science/article/pii/S2213846313000114
28. Lemaignan, S., Siadat, A., Dantan, J.Y., Semenenko, A.: Mason: A proposal for an ontology of manufacturing domain. In: IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06). pp. 195–200 (2006). https://doi.org/10.1109/DIS.2006.48
29. Malburg, L., Klein, P., Bergmann, R.: Semantic web services for ai-research with physical factory simulation models in industry 4.0 (11 2020). https://doi.org/10.5220/0010135900320043
30. Malburg, L., Rieder, M.P., Seiger, R., Klein, P., Bergmann, R.: Object detection for smart factory processes by machine learning. Procedia Computer Science

184, 581–588 (2021). https://doi.org/https://doi.org/10.1016/j.procs.2021.04.009, https://www.sciencedirect.com/science/article/pii/S1877050921007821, the 12th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 4th International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops

31. Miller, E.: An introduction to the resource description framework. Bulletin of the American Society for Information Science and Technology **25**(1), 15–19 (1998). https://doi.org/10.1002/bult.105, https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/bult.105
32. Negri, E., Fumagalli, L., Macchi, M.: A review of the roles of digital twin in cps-based production systems. Procedia Manufacturing **11**, 939–948 (2017). https://doi.org/10.1016/j.promfg.2017.07.198, https://www.sciencedirect.com/science/article/pii/S2351978917304067, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy
33. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic Web **8**, 489–508 (2017)
34. Schneider, E.W.: Course modularization applied: The interface system and its implications for sequence control and data analysis. In: Association for the Development of Instructional Systems (ADIS. Chicago, Illinois
35. Tao, F., Zhang, H., Liu, A., Nee, A.Y.C.: Digital twin in industry: State-of-the-art. IEEE Transactions on Industrial Informatics **15**(4), 2405–2415 (2019). https://doi.org/10.1109/TII.2018.2873186
36. Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., Sui, F.: Digital twin-driven product design, manufacturing and service with big data. The International Journal of Advanced Manufacturing Technology **94** (02 2018). https://doi.org/10.1007/s00170-017-0233-1
37. World Wide Web Consortium: SPARQL Query Language for RDF. https://www.w3.org/TR/rdf-sparql-query/, [Online; accessed 15-August-2021]
38. Xu, L.D., Xu, E.L., Li, L.: Industry 4.0: state of the art and future trends. International Journal of Production Research **56**(8), 2941–2962 (2018). https://doi.org/10.1080/00207543.2018.1444806, https://doi.org/10.1080/00207543.2018.1444806