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. Smart manufacturing offers potential for new manufacturing models by using modern technologies. One paradigm, that helps realising smart manufacturing is the concept of digital twins, which are digital representations of physical assets such as production machines. By collecting data, analysing failure causes or predicting maintenance actions, the overall production process may be improved through the use of digital twins. Besides other modelling approaches, e.g. using AutomationML or CAD technologies, especially semantic information modelling ensures machine-readable connections between all involved components of the digital twin while avoiding data redundancy and enabling smart analysis. To model and access semantic information, knowledge graphs provide a promising option to fully enable the aforementioned benefits in a lightweight manner. To further elaborate on their suitability, the following paper aims at evaluating the potential and limitations of building a semantic digital twin in a managed knowledge graph platform.

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 [32]. This revolution, which is particular visible in the manufacturing sector, has been coined by the term *Industry 4.0* [18]. 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 [8]. Smart manufacturing however requires deep integration between the digital and the physical space. One of the most promising enabling paradigms that have emerged for realising smart manufacturing in the Industry 4.0 are *digital twins* [30].

There are many ways to model a digital twin. Besides formats such as AutomationML, UML [3] or classical CAD-based approaches, especially semantic information models are a promising way to formally represent the digital twin in a machine-understandable way [11]. 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 is the *knowledge graph* [11]. Using knowledge graphs provides advantages such as the integration of different data sources and semantically linking them. In order to build, modify and query knowledge graphs in a lightweight manner, managed *knowledge graph platforms* offer several functionalities with convenient user experiences [14]. Furthermore, their options to visually construct semantic searches and queries allow for user-friendly querying [14], without requiring the users to have in-depth knowledge about the underlying semantic technologies, such as the *Resource Description Framework (RDF)* [25]. Considering this, knowledge graphs offer many benefits for hands-on usage of semantic technologies.

Therefore, the aim of this work is to prototype a digital twin using a knowledge graph platform in an experimental use case and to assess the resulting benefits. To do so, firstly the required foundations are introduced and related research is discussed (section 2), before a specific use case within the manufacturing context is selected and the resulting requirements towards the digital twin are derived (section 3). Thereafter, existing approaches and standards for modelling digital twins are considered in the conceptualisation phase, before the use case-specific digital twin is subsequently implemented in a knowledge graph platform (section 4). Finally, the concept and the implementation are evaluated (section 5) and a closing summary on the suitability of knowledge graph platforms for the implementation of digital twins is drawn (section 6).

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 discussed.

2.1 Digital Twins

The concept of digital twins was first introduced at the University of Michigan in the year 2003 [13]. To this day, there is no unique definition of this concept in the scientific literature [26]. 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 [21]. Digital twins can be used to collect data through different production stages and use that data to discover failures, failure causes, streamline supply chains, enhance the production efficiency [30], and predict required maintenance actions during a machine's lifetime [11]. 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 [19].

2.2 Knowledge Graphs

Although the ideas of the concept reach back to 1972 [28], the term was significantly coined by the introduction of knowledge graphs in the Google search engine in 2012 [29]. There are a variety of definitions and descriptions that have emerged for this term [9], but knowledge graphs tend to describe real world entities and their relations in a graph while defining possible classes and relations [27]. Knowledge graphs therefore can also be built in accordance to standard approaches such as RDF, which can provide better opportunities to automatically process the data and share it with other stakeholders. Knowledge graphs can be built and accessed by using knowledge graph platforms that provide user-friendly ways to query and modify the graph [14].

2.3 Asset Administration Shell

A standardisation approach that may be used for the implementation of a digital twin is the *Asset Administration Shell (AAS)* [4]. 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 standardised interface is ensured [4]. 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 [4]. These submodels are a standardised 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 a 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 standardised manner. Proposed data formats for the AAS are XML and JSON for serialisation of the AAS and RDF to enable the advantages of semantic technologies [4].

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* [16] 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 [17] or video-based monitoring of manufacturing processes [24]. 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 colour. 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* [16].

2.5 Related Works

Despite the large number of publications in the context of digital twins, no uniform modelling concept has yet achieved acceptance [30]. Especially in the context of semantic digital twins, the number of publications dealing with this domain is still rather small. Boschert et al. [6] present an approach trying to utilise knowledge graphs for comprehensive sharing of information along all life cycle phases of a digital twin. Another work in this domain is [11], 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, [12] proposes an approach to semantically model Industry 4.0 components in an AAS. This approach examines the advantages of semantics, such as interoperability achieved by using RDF, using uniform resource identifiers (URI) for global identification or using SPARQL[1] to query the data comprised in a semantic AAS [12]. However, their exemplary translation of a standard for electrotechnical components into a semantic class hierarchy is not applicable to the use case within the Fischertechnik environment since the available components and data are of different structure. Another approach that highlights the idea of semantic AAS is [2], where a mapping from the AAS XML serialisation to an RDF serialisation 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 section introduces some general application domains for digital twins in manufacturing to frame the context of a specific use case for the pro-

[1] https://www.w3.org/TR/rdf-sparql-query/

totype that is defined afterwards. 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 [30]. 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 standardisation approaches such as the AAS. Next, it needs to be specified, in which way the digital twin may utilise and extend existing knowledge structures in the chosen implementation platform, before the final implementation is realised in an exemplary knowledge graph platform.

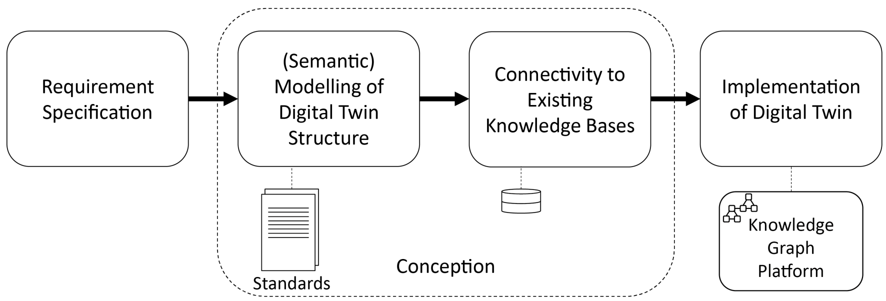

Fig. 1. Methodology for modelling and implementing a semantic digital twin

3.1 General Use Cases

Literature proposes several use cases for digital twins. Since the use cases are so wide-ranging, only use cases within the maintenance domain are considered below. Within their literature review, Errandonea et al. [10] 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. Summarising potential functionalities that a digital twin may include, Tao et al. propose a classification of nine different service categories [31]. 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.
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.

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.

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 standardisation 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). This submodel with its respective wear information can then be used for state monitoring of the asset, forecasting of the wear status if further workpieces are processed or maintenance actions can be derived based on the wear status. Summarising, the use case of a maintenance digital twin is implemented for two component categories, ovens and sorting machines, for which a digital twin 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 stakeholders 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, which is regarded in the following requirements. However, as the concept of 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 [23]. 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 ISO 29148 [1] using the terms "shall" for mandatory requirements and "should" for non-mandatory but desired requirements and serve as a guideline in the following implementation. The digital twin ...

R1 ... shall provide interoperability and interconnectivity [23], e.g. connecting different data sources.

R2 ... shall connect to existing knowledge representations [23], such as the FT-Onto [16].
R3 ... shall enable ontology and knowledge base updates [23], so that the digital twin is able to dynamically update attribute values in the underlying knowledge base.
R4 ... shall be enriched with semantic descriptions [23] to ensure comprehensibility of object, relation and attribute types.
R5 ... shall model the mechanical wear of the oven and the sorting machine by using a reusable, standardised approach.
R6 ... shall enable querying information about the mechanical wear of the component without requiring specific domain knowledge about the modelling of mechanical wear.
R7 ... shall allow users to modify and extend the digital twin without requiring RDF knowledge.
R8 ... should be able to predict the status of mechanical wear of the machine after a specified amount of workpieces is produced.

4 Semantic Digital Twin based on a Knowledge Graph

In this section, the approach of building a semantic digital twin based on a knowledge graph is illustrated by firstly explaining the general framework and subsequently describing a concept for the structure of the digital twin based on this framework. Additionally, the prototypical implementation of this concept is described. The framework and in particular the structure of the digital twin is specifically developed to meet the requirements that are described in the previous section.

4.1 Conceptual Framework

Although there is an agreement in the literature about the basic potential of digital twins, concrete standards and frameworks for the implementation have not yet achieved establishment [30] due to missing harmonisation and integration of different approaches [20]. 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 visualises 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 be 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 can use and query the digital twin in a lightweight manner to satisfy R6. The proposed data infrastructure 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

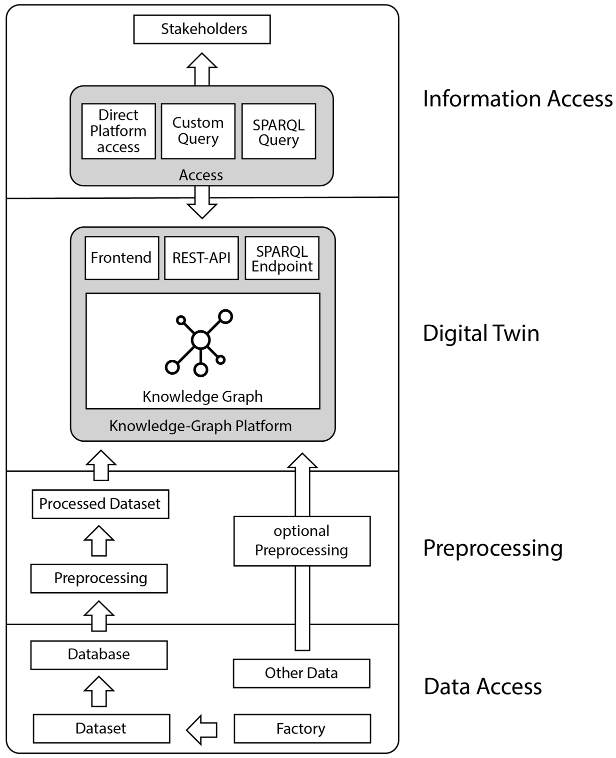

Fig. 2. Digital twin framework

that go beyond the capabilities of knowledge graph platforms. To the best of the authors' knowledge, the general features of knowledge graph platforms, such as *i-views*[2] or *Metaphactory* [14], are rather limited regarding more complex preprocessing, e.g. calculating advanced attributes from imported data.

The preprocessing step can be used to filter the data or calculate required indicators, which are then fed into a database afterwards. The database, in turn, represents the data source to the digital twin, on which the knowledge graph operates. 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.

[2] https://i-views.com/

4.2 Digital Twin Structure

The structure of the digital twin that is conceptualised in accordance with the previously described approach is illustrated in figure 3. The digital twin consists

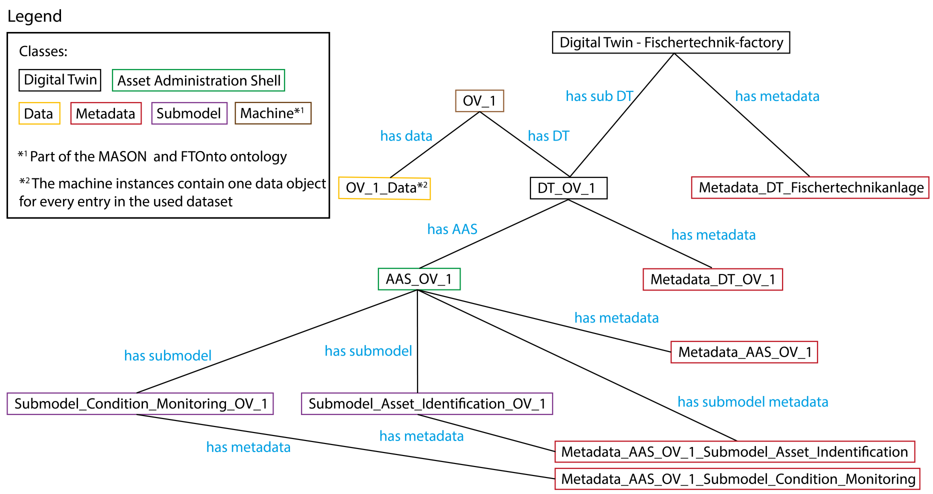

Fig. 3. Overview of the digital twin structure

of a hierarchical structure with various classes being linked by different relations. For example, the class "Digital Twin" is used as a top level class for the entire digital twin and the sub digital twins of every asset. Since R2 requires the digital twin to use existing knowledge representations, the digital twin is connected to the FTOnto and MASON ontology by using the class "Machine", which is part of these ontologies. The ontologies are not modified but only linked to the digital twin to potentially allow dynamic updates of these ontologies, which helps to satisfy R3. The structure of the individual sub digital twins for the different assets is modelled by referring to the standardised approach of the AAS [4] and specifically to the discussion paper "Verwaltungsschale in der Praxis" [7]. Each sub digital twin therefore contains an instance of the AAS class, which contains different submodels that describe specific parts of an asset. The submodel "Asset Identification" can be used to uniquely identify an asset. For example, this submodel can contain information like the *Global Location Number*[3] of the assets' manufacturer, which provides interoperability with regard to R1. The condition monitoring submodel is used to model the current wear of an asset as required by R5, which states that the mechanical wear shall be modelled in a standardised approach. Since the submodels in the AAS approach are modelled in the same way for every asset, no asset-specific domain knowledge is required to interpret the submodels (R6). Using the standardised AAS approach to define the structure of the digital also aids the satisfaction of R1 by improving the interoperability of the digital twin. In addition to that, the metadata added in

[3] https://www.gs1.org/standards/id-keys/gln

accordance with the AAS approach satisfies R4, which requires the digital twin to be enriched with semantic descriptions, such as a URI.

4.3 Implementation

In the following, the previously described approach and structure for a semantic digital twin is applied in a prototypical implementation. To implement the digital twin, the enterprise knowledge graph platform Intelligent Views (i-views)[4] in version 5.3 is used, which is provided by the Empolis Information Management GmbH[5] for this research project. Using a knowledge graph platform like i-views allows users to modify and extend the digital twin without requiring RDF knowledge, which is the aim of R7. In accordance with R5 the prototype is limited to only include the ovens and sorting machines of the Fischertechnik factory model.

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), linear and exponential wear is simulated to showcase a condition monitoring submodel reflecting the wear of the components in the digital twin. The wear is simulated by reducing the motor speeds of the transport components, which results in higher transport times.

Data Preprocessing In order to generate necessary attributes, that are non-existent in the initial data, a Python-based, upstream data preprocessing calculates and adds new attributes. Table 1 shows the attributes that are calculated during the preprocessing of the data concerning the ovens of the factory model. These attributes are necessary to model the wear of the components (R5) as well as enable querying of the wear without deep domain knowledge (R6). The calculated transport time is used to determine a new attribute "wear type", which shows if the degradation rate is stable (e.g. linear degradation) or accelerating (e.g. exponential degradation). Additionally, the wear type and the change in transport times are used to calculate the remaining useful life of the machine. This allows to predict the point where the wear of the machine reaches a critical level with regard to R8.

Building the Digital Twin Structure The first step in setting up the digital twin is to import the already existing FTOnto [16] into i-views. In addition to that, the MASON [22] and the SOSA [15] ontologies are imported since the FTOnto relies on some of their classes and relations. The imported ontologies are used as the basis for the semantic descriptions of the factory and its assets. While importing the ontologies into i-views several compatibility problems arise because i-views does not support all namespaces and constructs that are used in these ontologies. Therefore some preprocessing is required. Subsequently the needed object and relationship classes are created. These classes are used to

[4] https://i-views.com/
[5] https://www.empolis.com/en/

Attribute	**Description**
transport time	The transport time of a particular workpiece.
average combined transport time	The average combined transport time of the last five workpieces.
number of parts	The number of parts a machine has processed since the last maintenance.
wear type	The "type" of wear. This attribute can either be "stable" or "increasing". If the attribute is "stable", the wear increases by the same amount with every workpiece (linear degradation). If the attribute is "increasing", the wear caused by each workpiece is increasing (e.g. exponential degradation).
wear rate/factor	If the wear type is "stable" this attribute contains the amount of wear that each workpiece causes. In case of the ovens this is the amount of milliseconds by which 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 parts the machine can still process before the wear reaches a critical level.

Table 1. Calculated attributes for the two ovens

create the structure described in section 4.2 by creating instances of them and linking these with instances of the relationship classes. Furthermore, several attribute classes are created to define the possible attributes that these instances can have. The final RDF-structure of the digital twin is made available via a Github-repository[6] in order to foster further research on semantic digital twins using knowledge graph platforms.

Restrictions Since one of the requirements for the digital twin states that it shall allow users to modify and extend the digital twin without requiring RDF knowledge (R7), some restrictions are integrated into the digital twin to prevent possible mistakes when working with the digital twin. The defined relationship classes are limited to be used between instances of specific classes and the amount of relations of a specific type are limited as well. It would also be advantageous to force certain class instances to have specific relationships but i-views does not support such restrictions. Furthermore, the defined attribute classes have an assigned data type (e.g. integer) which prevents users from entering potential false data. It would be advantageous to further limit the attribute definitions (e.g. limit integer attributes to positive numbers only) but i-views' features for doing that are rather limited.

Querying Submodel Information In order to query the information of the submodels and therefore satisfy R6 some queries are predefined within i-views. The potential stakeholders could access these predefined queries or create their own 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

[6] https://github.com/DavidJilg/Digital-Twin-using-a-Knowlege-Graph

with a SPARQL-Endpoint the stakeholder could also use a SPARQL-Query to search for information in the knowledge graph.

5 Evaluation

In the following section, the implemented approach is evaluated in an exemplary use case scenario. Reconsidering the purpose of the implemented condition monitoring submodel (section 3.2), the modelled wear information can be used in various situations.

The evaluation scenario assumes a production employee who wants to get insights into the existing wear information of the two ovens based on the information of the digital twin. For this purpose, he first queries the entire condition monitoring submodel of both ovens. Based on this information, he then wants to make a decision on which of the two ovens the next batch of parts should be processed. To query the submodels, the employee can use the implemented query illustrated in figure 4, which takes a machine as parameter and returns the corresponding condition monitoring submodel. Therefore, the employee specifies the parameters with "OV_1" and "OV_2" and receives the output displayed in figure 5. The production employee can then interpret the results and decide

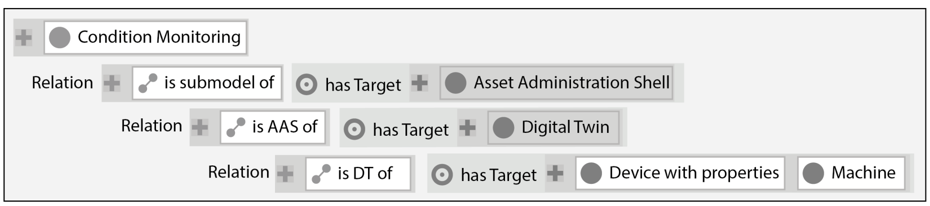

Fig. 4. Query to get condition monitoring submodel

which oven to use. Figure 5 shows that the wear factor is "stable" in OV_1 and "increasing" in OV_2. Furthermore, OV_1 has 34 parts remaining before a critical wear level is reached, compared to 12 remaining parts in OV_2. Also the transport time of OV_1 is with ∼8000ms lower than ∼9080ms in OV_2. Combining this retrieved information, the production employee therefore decides to use OV_1 for the next batch of pieces. Besides this, the employee could even use the information to plan the next maintenance action or predict when a failure is likely to occur, if no maintenance is scheduled. The implemented query can either be used through the i-views user interface directly, it can be used through the Rest-API embedded in another application or even by an automated system since the data is implemented in a machine-readable way.

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

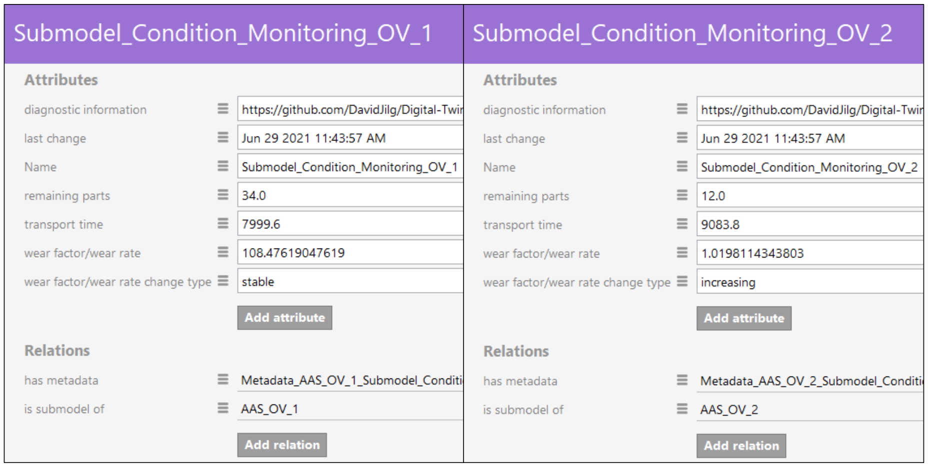

Fig. 5. Queried submodels of OV_1 and OV_2

deep technological knowledge [14]. 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 several stakeholders. Combining these characteristics with standardised frameworks such as the AAS, reusable components and submodels may be created that together avoid redundant data while enhancing the aforementioned benefits.

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, the prototype might be extended by further submodels besides the two described in this paper (condition monitoring and asset identification). An exemplary, related submodel that could build on and extend 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 [5] that stores previously collected failure cases and solutions. Additionally, the calculation that is currently used to determine the wear rate could be extended by using advanced simulation methodologies to predict future conditions of the asset.

In conclusion, to further evaluate on the proposed approach it should be extended to a more complex IoT scenario and further submodels could be added.

References

1. ISO/IEC/IEEE Draft Standard for software and systems engineering – Life cycle processes – Requirements engineering. ISO/IEC/IEEE P29148 First Edition (2011)
2. Bader, S., Maleshkova, M.: The Semantic Asset Administration Shell. In: Semantic Systems. The Power of AI and Knowledge Graphs. pp. 159–174. Springer International Publishing (2019)
3. 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
4. Barnstedt, E., et al.: 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) (2018)
5. Bergmann, R., Grumbach, L., Malburg, L., Zeyen, C.: ProCAKE: A Process-Oriented Case-Based Reasoning Framework. In: Workshops Proceedings for the Twenty-seventh International Conf. on Case-Based Reasoning co-located with the Twenty-seventh International Conf. on Case-Based Reasoning (ICCBR 2019). CEUR Workshop Proceedings, vol. 2567, pp. 156–161. CEUR-WS.org (2019)
6. Boschert, S., Heinrich, C., Rosen, R.: Next Generation Digital Twin. In: Proceedings of TMCE 2018 (2018)
7. 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]
8. 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 (2019)
9. Ehrlinger, L., Wöß, W.: Towards a Definition of Knowledge Graphs. In: SEMANTiCS (Posters, Demos, SuCCESS) (2016)
10. Errandonea, I., Beltrán, S., Arrizabalaga, S.: Digital twin for maintenance: A literature review. Computers in Industry **123**, 103316 (2020)
11. Gómez-Berbís, J.M., de Amescua-Seco, A.: SEDIT: Semantic Digital Twin Based on Industrial IoT Data Management and Knowledge Graphs. In: Technologies and Innovation. pp. 178–188. Springer International Publishing (2019)
12. Grangel-González, I., Halilaj, L., Coskun, G., Auer, S., Collarana, D., Hofmeister, M.: Towards a Semantic Administrative Shell for Industry 4.0 Components. In: 2016 IEEE Tenth International Conf. on Semantic Computing (ICSC). pp. 230–237 (2016)
13. Grieves, M.: Digital Twin: Manufacturing Excellence through Virtual Factory Replication (2015)
14. Haase, P., Herzig, D., Kozlov, A., Nikolov, A., Trame, J.: metaphactory: A platform for knowledge graph management. Semantic Web **10**, 1109–1125 (2019)
15. 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)
16. Klein, P., Malburg, L., Bergmann, R.: FTOnto: A Domain Ontology for a Fischertechnik Simulation Production Factory by Reusing Existing Ontologies. In:

Proceedings of the Conf. on "Lernen, Wissen, Daten, Analysen". CEUR Workshop Proceedings, vol. 2454, pp. 253–264. CEUR-WS.org (2019)
17. 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)
18. 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** (2019)
19. Lasi, H., Fettke, P., Kemper, H.G., Feld, T., Hoffmann, M.: Industry 4.0. Business & Information Systems Engineering **6**, 239–242 (2014)
20. 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 (2021)
21. 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)
22. 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)
23. Malburg, L., Klein, P., Bergmann, R.: Semantic Web Services for AI-Research with Physical Factory Simulation Models in Industry 4.0. In: Proceedings of the International Conf. on Innovative Intelligent Industrial Production and Logistics, IN4PL. pp. 32–43 (2020)
24. 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)
25. Miller, E.: An Introduction to the Resource Description Framework. Bulletin of the American Society for Information Science and Technology **25**(1), 15–19 (1998)
26. 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)
27. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic Web **8**, 489–508 (2017)
28. 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)
29. Singhal, A.: Introducing the Knowledge Graph: things, not strings. https://blog.google/products/search/introducing-knowledge-graph-things-not/ (2012), [Online; accessed 19-April-2021]
30. 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)
31. 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** (2018)
32. 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)