Digital Twins in Industrial Applications – Requirements to a Comprehensive Data Model

Authors:
Kilian Bächle
Account Manager IoT
CONTACT Software GmbH
kilian.baechle@contact-software.com

Stefan Gregorzik
Manager Corporate Development
CONTACT Software GmbH
stefan.gregorzik@contact-software.com

INTRODUCTION

The concept of the digital twin is central to the Industrial Internet of Things (IIoT). In order to make its application effective, various organizations are undertaking standardization efforts—for example in the form of the Industry 4.0 Initiative[1] and, of course, the Industrial Internet Consortium (IIC)[2]. In our opinion, it is crucial that such standardized digital twin concepts meet the following requirements:

- A common understanding of the granularity and significance of the contents of the digital twin is necessary in order to enable the joint monitoring of different assets.
- To create the digital twin of an end product, it must be possible to consolidate the various submodels from the supply chain and the different product components.
- Digital twins must be portable between systems in order to guarantee the independence of the operating companies—as is the case, for example, in various engineering fields thanks to the use of the STEP format.

Corresponding data models are needed, and some standardization initiatives have already resulted in meta-models for digital twins designed to address these requirements. However, in order to use these meta-models in practice, we believe it is necessary to further concretize them by defining relevant content, granularity und correlations between them. In our opinion, this is only possible with reference to the use cases for digital twins. Abstracting these use cases would then permit the generic matching of “use case types” and data model characteristics.

Furthermore, suitable transport media (protocols) must be selected and implemented. Instead of vendor-specific approaches (of the sort already available in the market), we strongly support the idea of an open-source solution. We regard this as the approach most likely to avoid any kind of “vendor lock-in.”

The contents and correlations presented in this paper are based on business cases and were obtained using our IIoT platform *CONTACT Elements for IoT*. Our assessments regarding relevance and feasibility are based on the experiences we have gained.

Outline of this article

This paper proposes content characteristics for digital twin data models that can be implemented in existing concepts. To support our proposals, we refer to various real-world examples, requirements and use cases.

We consequently take existing data models for digital twins as the basis for a further-reaching discussion in section 2. In section 3, we then describe the business cases that

[1] Plattform Industrie 4.0, ZVEI (eds.): Details of Asset Administration Shell. Specification Part 1 – The exchange of information between partners in the value chain of industry 4.0 (version 1.0). Berlin Nov. 2018

[2] Industrial Internet Consortium, Plattform Industrie 4.0 (eds.): Architecture Alignment and Interoperability. White paper. 2017

form the practical foundation for our proposals. Sections 4 and 5 present our conclusions in terms of the contents and behavior of digital twin models. These give rise to suggestions about how to realize such digital twin models, which we present in section 6, and we conclude with an examination of the outlook for the future (section 7).

DATA MODELS FOR DIGITAL TWINS

Probably one of the best-known data models for digital twins is the “Asset Administration Shell” as formulated by *Plattform Industrie 4.0*. In short, this model is structured around the core concept of an “asset” and describes the following aspects:

- Identification and description of an asset
- Submodels related to this asset
- Property definitions to specify data from operations
- Access Control

At least one crucial use case for the digital twin data model mentioned above takes the form of information exchange between partners in the supply chain, e.g. between “suppliers” and “integrators.”

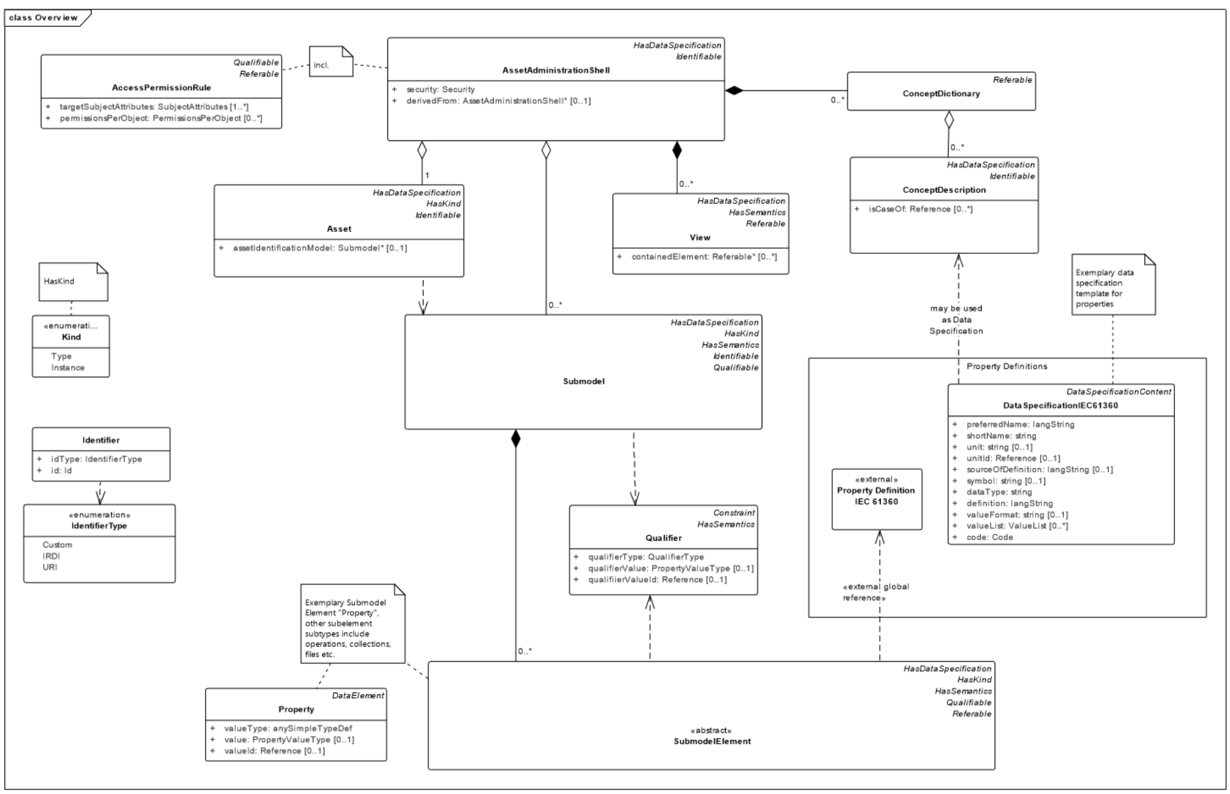

Figure 1: Meta-model of Asset Administration Shell (Plattform Industrie 4.0)

Gartner[3] defines the minimum requirements for a digital twin data model as:

- data structure and meta-data
- a system model of functional elements and critical measures
- extended by subsystems of the asset (optional)

In the following section, we focus on these concepts for the formulation of data model characteristics.

EXAMPLE SCENARIOS FOR IIOT

A study by the Fraunhofer IPK research institute, CONTACT Software and VDI (Association of German Engineers) analyzed the central applications for IIoT.[4] It identified *service optimization*, *customer process optimization* and *product optimization* as the three most important processes, followed by processes with a much narrower focus, such as *complaints management*, *process analysis* and *production optimization* (see Figure 2).

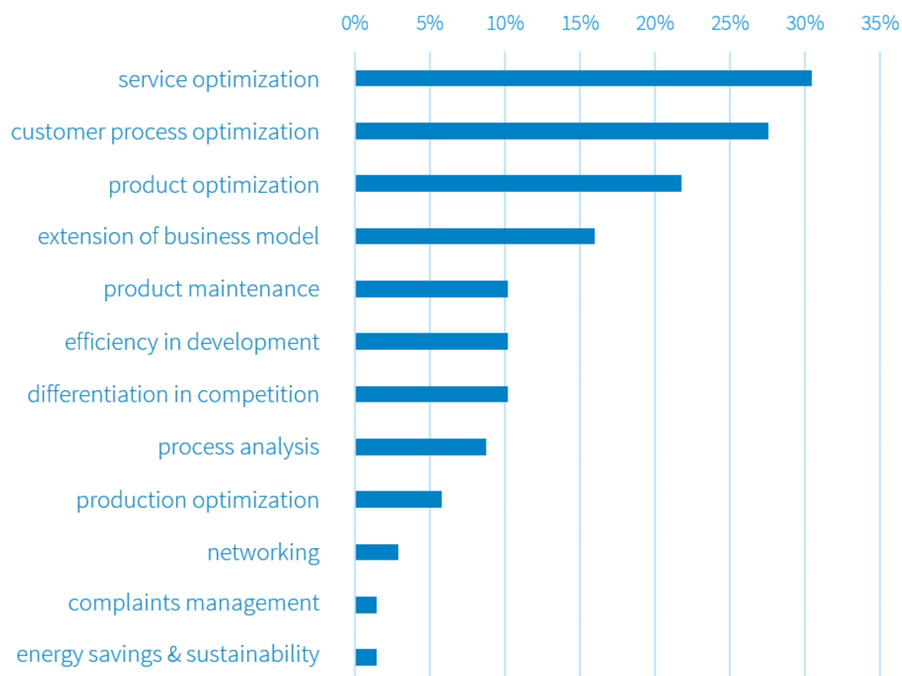

Figure 2: Applications for IIoT (x = % of matches from a cluster analysis of free text answers) (Smart Industrial Products, pg. 111)

[3] Gartner Inc.: Innovation Insight for Digital Twins — Driving Better IoT-Fueled Decisions. 22 March 2017.

[4] Müller, P.; Lindow, K.; Gregorzik, S.; Stark, R. (eds.): Smart Industrial Products. Berlin/Bremen/Düsseldorf 2019.

The study analyzed data collected from 183 experts representing different branches of industry, such as the automotive sector, machine engineering and plant construction. 75% of these experts exercised development or product management functions.

The results of this study provide a good general impression of the goals companies associate with IIoT: benefits are envisaged primarily at the level of broader processes like customer service optimization, whereas, according to the respondents to the study, specific applications have less potential.

Consequently, the following real-world scenarios are selected from our customers' focus on broader business cases and various interrelations within the supply chain. By means of this selection, we try to cover a range of typical industrial applications in order to give us a representative basis for the proposals that follow.

Case 1: Customer support and design loop

The first example is provided by an equipment manufacturer that sells batch reactors for mixing processes in the chemical industry. The manufacturer's customers usually install these reactors in machine lines, with various upstream and downstream processes being integrated around the mixers. The end products are chemicals that are very sensitive to production parameters such as temperature, pressure or humidity. Insufficient control of such parameters may lead to the scrapping of entire product batches and require the conduct of maintenance checks at the mixers. However, since the batch processes running on the machine lines are something of a black box, it is difficult to localize the actual faults and thus prevent the rejection of subsequent batches, thereby resulting in considerable costs due to product nonconformities and machine downtimes.

This situation led the manufacturer to decide to provide (digital) services for its mixers. In doing so, it had two aims in mind: first, to provide supplemental services (such as maintenance and spare part sales) for its mixing machines in order to better meet customer demands and strengthen the customer relationship. Second, to gain more accurate insights from field operations to enable it to improve its next generation of mixers by means of "closed loop engineering" (feedback round trip from operations to design).

The business-critical requirements for IIoT in this case are to:

- Track control parameters (temperature, speed, etc.)
- Consider environmental influences (humidity, temperature, etc.)
- Identify production jobs (e.g. for quality feedback)
- Identify spare parts

Case 2: Overall fleet monitoring and predictive maintenance

The second example comes from a large-scale logistics supplier that operates various sites for transshipping freight from global to local transport routes. The company operates a fleet of different-sized container carriers, fork lifts and cranes to address the different reloading situations. The critical goal when reloading is to minimize the downtime of ships, trucks and trains. The logistics supplier's performance is measured on the basis of the transfer time, and penalties are incurred if deadlines are missed. Therefore, the availability of the corresponding logistics equipment is a business-critical factor and requires sophisticated condition monitoring and maintenance control.

As these vehicles are provided by different suppliers, there is no common fleet-wide monitoring logic or even any common monitoring mechanism, and the logistics supplier has had to implement fleet control on its own. Of course, weather conditions play a significant role here. The objective of such a fleet control system is to monitor the different (groups of) vehicles and consolidate the data for the joint reporting of general readiness information. Furthermore, a predictive maintenance system based on historical data from the vehicles' internal control systems is intended to proactively trigger measures to prevent malfunctions and breakdowns. Automated spare parts provision might be the next logical step.

This case gives rise to the following IIoT requirements:

- Consolidate key measures of system behavior from different suppliers/assets
- Consider environmental influences
- Define limit values for parameters and related measures
- Provide spare parts orders

Case 3: Maintenance and modification of assets

The third case involves a manufacturer of large-scale plant systems for handling food products. Based on customer requirements, the manufacturer combines a number of modules and functional units to create an integrated system (configure-to-order). This system is then interlinked with the customer's production environment.

The manufacturer offers a full range of services from the design and manufacturing of the equipment through to its installation, maintenance and system overhauls. In most cases, tasks are divided, with the manufacturer and customer working together to commission the system while standard-maintenance and small extensions are performed by the customer, and more extensive or complex modifications are undertaken by the manufacturer.

At this point, a potential conflict arises when the customer makes changes to the system without informing the manufacturer and then subsequently requests a modification or complains about a system failure. The first challenge then facing the manufacturer is to identify the up-to-date as-maintained state of the system in order to have a reliable baseline for any further activity.

The key abilities required of an IIoT system in this context are to:

- Track control parameters (pressure, speed, etc.)
- Describe the as-maintained-state in terms of the components and assembly situation
- Integrate a description of third party components (modifications by customer)
- Describe the interaction with other surrounding production assets

Case 4: Monitoring of overall systems and predictive maintenance

The fourth case relates to train operation. Since trains are operated for many hours every day, often in tough environments, the requirements in terms of material, component and condition monitoring are extremely high. A carriage breakdown causes major difficulties. Furthermore, the withdrawal of a train from service has serious timetabling implications since replacing it may take hours. During operation, the repair options for trains are limited as the railway depot is often the only site where proper maintenance and repair can be performed.

Carriage doors, in particular, are a crucial component of a train. Any door failure can extend the stopping time in stations and cause delays. Furthermore, the safety buffer for evacuating trains in the event of an emergency is reduced if an exit is blocked; and if there are too many such exits, the train may not be allowed to operate at all. At the same time, doors are sensitive to contamination, and, as regular moving parts, mechanical and drive elements suffer from high levels of wear.

As trains have a long product lifecycle, the replacement of parts and the refurbishment of subsystems or even complete carriages is a regular activity. In many cases, such modifications to trains, regarded as a system in themselves, may influence the behavior of the doors. To predict their functionality or the risk of breakdowns based on sensor data, it is therefore necessary to possess an overall up-to-date as-maintained-state of the carriage.

Based on this use case, the most critical IIoT requirements are to:

- Consolidate key measures of system behavior
- Consider environmental influences
- Describe the as-maintained-state in terms of the actual components installed, including third party spare parts
- Describe the interaction with surrounding components of the overall system

Content of a Digital Twin Data Model

Regarding the requirements placed on digital twin data models, we need to start from a general view. In the survey mentioned prior ("Smart Industrial Products"), the three most important elements of a digital twin identified by the industry experts were: 1) the *software version* installed on the asset, 2) the asset configuration in terms of *product-specific parameters* and 3) the *unique ID* of the asset. Additional requirements just slightly outside of the top three were 4) views of *components* from the perspective of different disciplines, 5) related *documentation* and *audit information* and 6) *3D visualization* (see Figure 3).

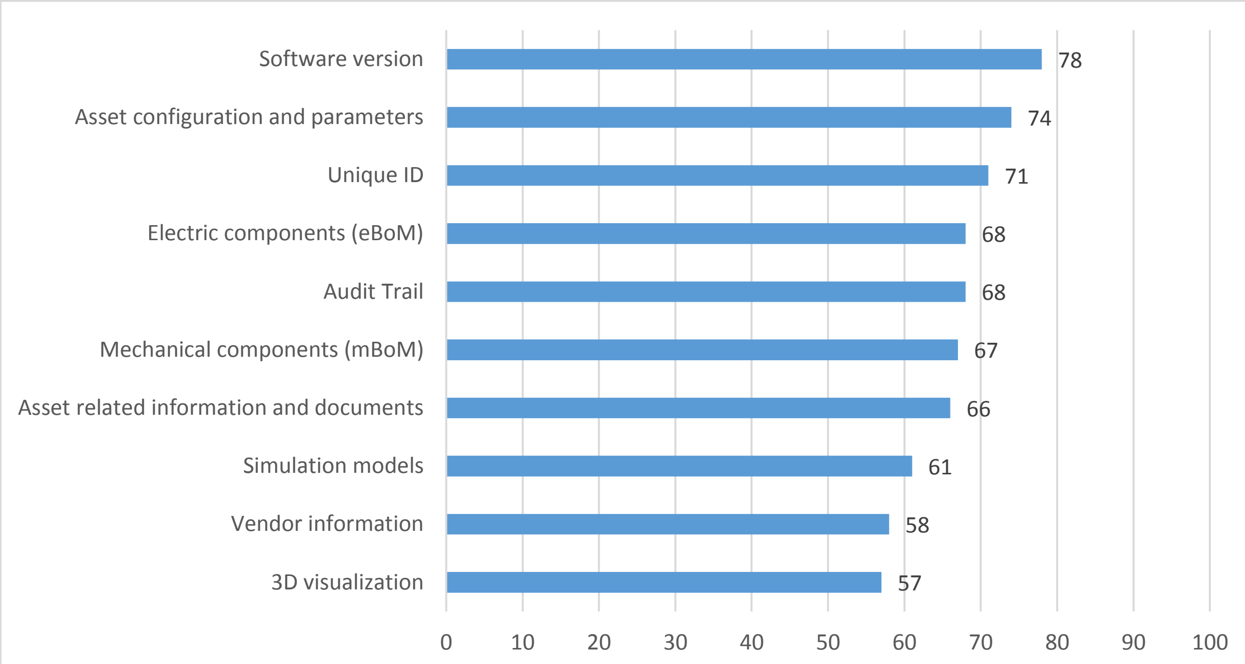

Figure 3: Requirements for contents of Digital Twins (x = % of positive answers, multiple selections possible) (Smart Industrial Products, pg. 68)

In the use cases mentioned in the previous section, we find a recurring pattern in IIoT applications: the mapping of *field data* to a *reference* in order to extract insights is the key mechanism for generating benefits in the business processes. Here, the digital twin is the essential basis for any monitoring, analysis or prediction relating not only to the identification of assets, but also to the provision of the circumstances and state of the asset at any point in time in relation to the data points. There is obviously a direct correlation between the precision of such data and the possible insights that may be gained.

The digital twin data model must not only contain the meta-description of the data, but also the rules, conditions and algorithms that define how to interpret the data in the context of the asset in question (e.g. border conditions, transformations from K to °C). As this interpretation or usage of data in general can vary from case to case, such *instructions for use* are of great relevance.

Taking the example use cases described above together with the survey results, we suggest the following general *reference characteristics* for a digital twin data model:

Assets

The asset is the key object in a digital twin data model. It uniquely identifies and represents a physical *thing* (the *classical* concept of the digital twin).

Components (including history)

An asset may consist of components (mechanical, electrical or electronic sub-systems or functional units) which have to be described insofar as they influence the asset's behavior. As the analysis of asset behavior is mostly based on time series data, the configuration history of an asset's component structure is of great relevance (e.g. for audit purposes).

Environment

Assets are influenced by their environment in terms of surrounding systems (e.g. carriages), installation sites (such as drilling platforms) and environmental conditions (such as hot or cold weather, presence of dust, etc.). Thus, the proper recording of such environmental data is a relevant aspect.

Models and descriptions

A simple verbal description coupled with the component structure may not suffice for all the data analytics requirements relating to an asset. Additional aspects such as geometry or the operating or maintenance situation (e.g. motor within a vehicle) are an essential part of the digital twin data model.

Control Parameters (including history)

To define the operating profile of an asset, the software programs and parameters used (such as settings for motor speed, pump pressures) must be recorded and stored. Once more, the data history is of great relevance because changing these parameters modifies the behavior of the asset.

In our opinion, the definition of a digital twin data model is not complete without these reference characteristics. As a smart asset delivers data in an individual format, the description of these data streams (not the field data in terms of "payload" itself but the meta-information) must be part of the data model if a fully-fledged basis for IIoT applications is to be achieved.

Behavioral Data

Data describing the behavior and operation of assets takes the form of values from sensors (such as temperatures, flows) and messages from controllers (e.g. error warnings, ready states). Both types of data may appear in different technical protocols and may have significantly different characteristics: Sensor data usually takes the form of a continuous stream of values while messages are discrete and intermittent.

Environmental Data

Data describing the situation and the environment in which the asset operates (e.g. temperature, humidity) may originate from sensors in the asset itself or its environment or can be provided by web services. In light of this variety of possible data sources, it is obvious that different technical protocols may be used to transfer this data (possibly even in the same way as for the behavior data). It therefore seems reasonable to consider this data separately in a digital twin data model.

Finally, another type of data needs to be included within a digital twin data model:

Connectivity Parameters

To be able to receive data from and send data to an asset, information on how to address the asset in the (inter)net is required. This means unique addresses such as IP's or MAC declarations are required together with a description of the authorization method.

As mentioned above, data can appear in different protocols. Another aspect of connectivity parameters therefore relates to the definition of these protocols and the formats to be interpreted in the IIoT system.

MAINTAINING A DIGITAL TWIN DATA MODEL

The major challenge associated with all the references and meta-descriptions mentioned in Section 4 relates to how to generate them and keep them up to date. The following requirements apply to the maintenance process:

Consolidation

As we have seen in the use cases (Section 3), a digital twin may relate to a product containing components that originate from multiple sources. Industrial products typically consist of a number of third party components assembled in deep supply chains with tier-1, tier-2 or other suppliers. To make it possible to create a digital twin of a final product, these sources have to be connected and consolidated into a common model to reflect the complete, fully updated *as-delivered*-state.

Hence, the concerns and challenges involved in keeping the digital twin up to date also apply to the OEMs in the supply chain. The OEM will require up-to-date twin descriptions of all the components in a way that permits the (automated) compiling of partial models from the different suppliers in the supply chain (or a number of pre-consolidated models from the tier-1 suppliers). It is clear that models have to be interoperable across system boundaries since OEMs and suppliers may not work with the same software.

Therefore, one initial requirement for maintaining digital twin model data is the ability to consolidate partial models from different parts of the supply chain into a single model that is associated with the final asset.

Maintenance and extensions

While the tasks described above are already complex, the main challenge actually arises during operation over the lifecycle of the asset. Maintenance and refurbishment can substantively change the way an asset behaves, and new or replacement components will be installed in it—including from suppliers that were not part of the original OEM supply chain. Third party maintenance providers may modify assets without reference to the OEM if they possess corresponding service-level agreements with the operating company.

This leads to the obligation on the part of the owner or operator of the asset to keep the digital twin up-to-date so that it describes the asset's as-maintained-state.

A second requirement for digital twin models therefore refers to the possibility of maintaining and extending them—even if the operator is not part of the manufacturing supply chain. This also includes the frictionless handover of the digital twin data from the manufacturing process to the operating process owner.

Granularity

The task of consolidating and maintaining a digital twin's data during operation raises the question of the level of detail the partial models and the overall model should possess. From the point of view of the final asset, a detail in a given component (e.g. a resistor in a motor) might not be of interest to the operator, whereas it could be a critical part for the manufacturer of the component (motor).

The way in which digital twins are handled throughout the supply chain must therefore include methods that make it possible to select and filter contents from one stage to another. As far as the top-level view of the final asset is concerned, only components that have an influence on the asset's behavior or functioning might be relevant in terms of monitoring and analytics. However, in terms of a closed loopback to product design or spare parts management, other components may play a role in the asset's business processes.

Obviously the required level of detail is a question of product design and the business case associated with the asset. As components are often used in several different products, the OEM's data requirements for a specific component in no way constitute a generally applicable component model—another OEM using the same component might have a totally different usage scenario and therefore a different view of the component.

Digital twin data models therefore need to provide options for controlling the granularity of the specific models that are processed.

Distribution and porting

It is obvious that unproblematic data transfer must be possible if the exchange of digital twins between stakeholders is planned. This means that it is necessary to be independent of transfer mechanisms, such as dedicated services or infrastructures (e.g. cloud platforms). Stakeholders must be able to choose their own way to communicate with each other.

Another aspect of independence concerns the portability of a digital twin: one of the players involved in the overall lifecycle of the asset may want to replace the existing IIoT system with a new one. It is therefore critical to be able to transfer the complete digital twin data model(s), e.g. including data dictionary, connectivity etc.

CONCEPTS TO REALIZE A DIGITAL TWIN DATA MODEL

When considering the requirements placed on the reference data of the digital twin model and its maintenance, we need to address the question of the complex data structures that have to be described along the supply chain. This question relates to both the data sources for the twin data model and the format used to describe and transport it.

Data Sources of Digital Twins

In our view, the only reasonable source for the essential structural data is to be found in Product Lifecycle Management (PLM), where consolidation capabilities, references to the product structure and, in particular, change information (including versions of structures) are available. We therefore regard PLM as the mechanism capable of establishing digital twins with solid foundations and consistent contents[5].

PLM deals with product information in a very detailed and granular manner due to its design focus ("Engineering-BOM"). Applying this information to a digital twin makes downsizing (quantitative reduction) necessary: as described above, not every screw or resistor is relevant for the monitoring and analysis of operations. In fact, quite the opposite is true: too many details would cause friction and overhead in the processes.

While reducing the structures involved in digital twins is highly beneficial, it is crucial not to lose the link to the design perspective. This is vital for channeling feedback from the field into the engineering processes. To implement such a closed-loop engineering process, it is necessary to take a retrospective view from the downsized structure of the digital twin to the engineering-BOM.[6]

Up to this point, the data provision process is very similar to that found in collaboration scenarios in engineering, e.g. in the case of joint ventures or consortia. In engineering, this is the point at which we regard the "abstract" product in terms of a design. However, if we move away from these typical PLM-based approaches, we have to consider the "thing" in terms of serialization, together with the resulting digital twin. This digital twin then reflects real-world data and makes it possible to utilize the data upstream in the process in order to optimize design and progress ("closed-loop engineering").

Describing and transporting Digital Twins

The next aspect is to describe a technical means that makes it possible to capture and distribute digital twins.

As an analogy, one could say that a *STEP-model[7] for digital twins* is required, i.e. a format allowing for the exchange of digital twins over system boundaries in high process quality. IIoT systems are mostly and—as far as we can see—will generally remain proprietary, at least in their specific customizations, due to the business cases they have to cover: we need only consider the above-mentioned different scenarios and system requirements that apply to OEM and component suppliers as an example. If we bear these differences in mind, it seems normal to assume that there will be friction and data loss (e.g. due to improper matching, incompatible categorization, etc.) at the interfaces between the source and target systems (based on a comparison of the quality of the internal data and the transferred data). By making the transfer format more comprehensive, it will be possible to reduce these losses and achieve high process quality in

[5] See also: Malakuti, S.; Schlake, J.; Grüner, S.; Schulz, D.; Gitzel, R.; Schmitt, J.; Platenius-Mohr, M.; Vorst, P.: Digital twin – a key software component of Industry 4.0. ABB Review 12/2018.

[6] Dickopf, T.; Apostolov, H.; Müller, P.; Göbel, J.; Forte, S.: A Holistic System Lifecycle Engineering Approach. 29th CIRP Design 2019.

[7] Wikipedia: ISO 10303 (https://en.wikipedia.org/wiki/ISO_10303), visited 6 September 2019

terms of low waste or low levels of manual interventions.

A suitable candidate for implementing an environment for digital twin data models[8] is the Eclipse Ditto project. As members of the Eclipse IoT working group ourselves, we consider the Eclipse foundation to have a strong reputation and a substantial working community that will be able to ensure the lasting and reliable introduction of a comprehensive digital twin model into the IIoT community.

Ditto[9] is an implementation of a fundamental data model and consists of an API that supports connectivity via protocols like MQTT. The central object in the data model is a *thing* that can be described in terms of *attributes* and extended by a number of *features* describing data and functionality related to the asset (see Figure 4). The implementation also includes revision information on changes to the digital twin.

CONTACT Software will work on example interfaces and payload examples for Ditto in order to describe our requirements for digital twin data models.


```json
{
  "attributes": {
    "manufacturer": "ACME",
    "VIN": "0815666337"
  },
  "features": {
    "transmission": {
      "properties": {
        "automatic": true,
        "mode": "eco",
        "cur_speed": 90,
        "gear": 5
      }
    },
    "environment-scanner": {
      "properties": {
        "temperature": 20.8,
        "humidity": 73,
        "barometricPressure": 970.7,
        "location": {
          "longitude": 47.682170,
          "latitude": 9.386372
        },
        "altitude": 399
      }
    }
  }
}
```


Figure 4: Example of a JSON representation of the Eclipse Ditto data model (Ditto release notes)

OUTLOOK: ARRANGING MODEL CONTENTS BY USE CASE

Establishing a technical option for exchanging digital twins is only a first step towards a comprehensive data model: a technical format does not answer questions about which model contents are appropriate, in particular in terms of references, data and granularity. To make it easier to distribute digital twin data models within the supply chain, it is

[8] Roest, M.: An open source platform for digital twins? (https://www.linkedin.com/pulse/open-source-platform-digital-twins-mark-roest), visited 10 September 2019

[9] Eclipse Ditto: Release Notes Version 0.9.0., released on July 10., 2019 (https://www.eclipse.org/ditto/release_notes_090.html)

necessary to possess a framework for determining the contents of partial models.

Process models in the fields of project management or quality control, for example, often permit customizing. For example, the development of a variant does not require the same full-blown development process as is needed in the case of new products.

Consequently, the framework for model contents would have to establish a relation between different use cases in the supply chain and their related requirements, on the one hand, and the digital twin data model on the other. When requesting data from suppliers, it would only be necessary to quote a use case taken from a standardized set. System models (such as SysML) could also be created to permit a machine-interpretable description of use cases.

CONCLUSION

To permit the practical application of existing digital twin models, it is necessary to enrich existing models with proposals relating to their content and structure. If it is possible to link these content elements with generic use cases, then this will support the effective applicability and interoperability of digital twins, for example between the stakeholders in a supply-chain. In this paper, we have used example cases to show how such derivations are possible. For implementation, we recommend the close coupling with PLM systems and an open protocol in the form of the Eclipse Ditto project.

➢ Return to IIC Journal of Innovation landing page for more articles and past editions

The views expressed in the *IIC Journal of Innovation* are the contributing authors' views and do not necessarily represent the views of their respective employers nor those of the Industrial Internet Consortium.

© 2019 The Industrial Internet Consortium logo is a registered trademark of Object Management Group®. Other logos, products and company names referenced in this publication are property of their respective companies.