Fraunhofer

Real-Time Digital Twin

Janina Knußmann, Robert H. Schmitt
DOI: 10.24406/h-442675

Extract from:
ICNAP Study Report 2022
International Center for Networked, Adaptive Production, Aachen
DOI: 10.24406/publica-1262

Real-Time Digital Twin

Janina Knußmann
Research Fellow
Production Metrology
Fraunhofer Institute for Production Technology IPT

Prof. Dr.-Ing. Robert H. Schmitt
Member of the board of directors of Fraunhofer IPT and holder of the chair for Production Metrology and Quality Management at the WZL of RWTH Aachen University

Introduction

Digital Twins (DT), the virtual representation of physical assets, can bring enormous value to the shopfloor. They are expected to reduce time and costs, as well as increase efficiency and knowledge about the product use. The possible use cases cover the whole value chain, including product planning, product development, production planning, production, and assembly, as well as logistics, maintenance and service during use, disassembly, and recycling. By mapping the real asset into the digital world, optimization, error-analyses, and even simulation of behavior can take place, enabling the usage of large-scale computing power, such as factory cloud systems.

The IT systems currently used for Digital Twins are Product Lifecycle Management (PLM), Enterprise Resource Planning (ERP), internal solutions, CAD systems, and the cloud. This is reflected in the data that is most often stored in Digital Twins: the largest portion is geometry data, followed by kinematics and behavior. In addition, Digital Twins store control, simulation, and environmental data. System models, however, only play a smaller part in current Digital Twins [1]43. To feed the Digital Twin with live data from the shopfloor, an increasingly integrated and connected production as well as a higher use of sensors and computing instances, such as edge and cloud devices, is needed. Protocols like Mqtt, LWM2M or OPC UA can be used to connect the sensors and actuators via Ethernet to the Digital Twin, allowing live monitoring of the production and subsequent optimizations.

To use the Digital Twin on the SCADA, PLC, and field level of a running production, the Digital Twin must meet industrial requirements, such as real-time capability. Most systems and devices at this level have hard real-time requirements to ensure precise and predictable behavior. Current Digital Twin solutions do not support these requirements and can therefore only be used for a retrospective analysis of the process, not for an optimization during the runtime. However, those optimizations have huge potential to reduce rejects and optimizing quality. This raises the question: can the Digital Twin also be used down to the field level? How can these systems benefit from a Digital Twin? What would be tangible use cases? And could they be implemented using state-of the art software frameworks? What are the specifics and challenges of a Digital Twin in a real-time context?

In the ICNAP study “Real-time Digital Twin” the goal is to explore the topic of Digital Twins in a real-time context and get answers to the previously asked questions. Starting with the state of the art of Digital Twins, the current understanding and use is described in the following report. The concept of real-time computation and communication is introduced to distinguish hard and soft real-time from information generated at runtime. Then, the term Real-time Digital Twin is defined. Further, the analysis of use cases, which have been identified in the literature as well as in the ICNAP community, show the potential of using Digital Twins in a real-time context. To provide a direct entry point into the use of the Real-Time Digital Twin with this study, various hard and software tools for realizing Digital Twins are covered, and the software frameworks for Digital Twins are evaluated. Finally, this report then touches on open research topics.

Digital Twins for industrial production

Digital Twins

Digital Twins are the representation of assets in the digital world. They consist of more than data that is assigned to a physical object: To make a Digital Twin a Digital Twin, it also must model the state of the physical twin supported by data [2]. The Digital Twin must also be associated with an existing physical twin, otherwise it would only be a model [3]. The ISO 23247-1 defines a Digital Twin as "a fit for purpose digital representation of an observable manufacturing element with synchronization between the element and its digital representation" [4]. The observable manufacturing elements can include processes and documents. Synchronization allows to update both the Digital Twin and the observable manufacturing element with the respective values. The synchronization can either be done continuously or if an event happens, which is realized via communication protocols.

The ISO 23247 does not only define the Digital Twin and the synchronization, but it also presents a Digital Twin framework for manufacturing (see Figure 11): It consists of a user domain, a Digital Twin domain, and a device communication domain and then builds on the domain of the observable manufacturing elements. The device communication domain collects data from the observable manufacturing elements and controls and actuates them. The user domain analyzes the Digital Twin for humans, for example via a human machine interface. Alternatively, the user domain also provides access to the Digital Twin for Enterprise Resource Planning (ERP) or Manufacturing Execution Systems (MES). The Digital Twin domain holds the Digital Twin, takes care of updating it with the data from the device communication domain, and manages access to the Digital Twin by the user domain. Applications that operate on the Digital Twin, such as a simulation, are also part of the Digital Twin domain [5]. The Digital Twin should be implemented using appropriate tools and should include relevant data and models. It is also possible for the Digital Twin to only include a part of a system. [4]

Real-Time computation and communication

Important processes within automation and production technology are subject to strict time requirements, also called real-time. While the term real-time is often used synonymously for fast processes, real-time systems do not necessarily need to be fast. It is more important that a real-time system, including computation and communication, is predictable. Results or sent information do not only need to be correct and fast, but also be predictably available before a certain deadline [6]. A distinction can be made between hard and soft real-time, where the difference is how long after the deadline the value of a computation or communication is valid.

Hard real-time: The data loses its value directly after the deadline. If the system misses the deadline, there are consequences, e.g., the quality of the product will be affected, or someone could be in danger. Examples for this include control loops, safety systems, or the brake of a car.

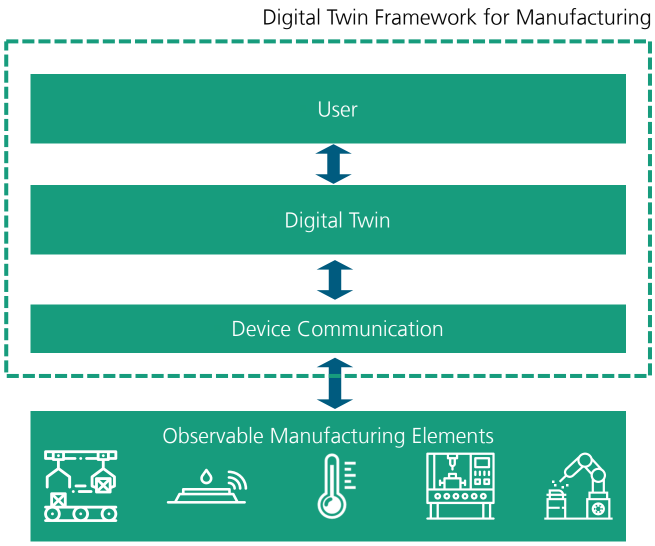

Figure 11: The Digital Twin framework for manufacturing [5]

Soft real-time: The data is also valid if the deadline is missed occasionally. If the system misses the deadline, the operation degrades. Examples for this include multimedia streaming, or weather stations.

Most Digital Twins today are either not a real-time system or only apply soft real-time.

Real-time computation: Real-time computation systems must do the respective data-processing according to the requirements of the connected outside process [7]. Therefore, they have to be synchronized with the external events and react directly. The availability of the result of the processing then must meet the deadline predetermined by the outside process. It is also required that the deadline is met predictably even in worst-case scenarios. The correctness of results of a real-time computation system are therefore not only defined by the correctness of the computed result but also by the ability to meet the predetermined deadline that is independent of the processing speed [8].

Real-time communication: To meet these requirements in industry, special physical fieldbus systems are used for time-critical sequences and processes in automation and production technology. These connect control and field devices and also enable cyclic and deterministic communication. However, the proprietary and vendor-dependent nature of the standards used to date makes it difficult or even impossible to combine them with networks or devices that use other communication standards, such as Cloud or Edge Systems, relying on IEEE 802.3 Ethernet. With Time-Sensitive Networking (TSN), Deterministic Networking (DetNet) and 5G that may change. All are compatible with Ethernet and offer real-time mechanisms for deterministic communication.

Definition of a Real-Time Digital Twin

The first application for Digital Twins in manufacturing listed by the ISO 23247-1 is real-time control of a manufacturing process [4]. As discussed in section 2.2, this process could be categorized as a hard real-time process.

Therefore, a Digital Twin in a real-time context is also a Digital Twin, but it has different requirements than the normal Digital Twin. To emphasize the use of a Digital Twin in a real-time context we can define the term "Real-Time Digital Twin". Therefore, a Real-Time Digital Twin is a Digital Twin on which hard or soft real-time timing requirements are placed.

Use Cases

Use case categories by the Trans Continuum Initiative

While in the literature the real-time Digital Twin is not yet a very defined term, in a whitepaper of the TransContinuum Initiative, the term Real-time Digital Twin refers to the use "for online prediction and optimization of highly dynamic industrial assets and processes" [8]. This is posing soft and hard real-time requirements on the Digital Twins and therefore matches with the definition of this ICNAP study.

In the whitepaper, use case categories including sample use cases are defined.

Milling with robots

In a milling process, the accuracy depends on the stiffness of the milling machine. In this use case, the vibration that occurs during the milling process is predicted during the process. The results are then used to compensate the vibrations occurring in a robot during the milling process, and this way the accuracy of the milling of the product is enhanced. This allows us to use robots for industrial milling tasks. [8] This use case falls into the category of performance optimization.

Increasing uptime of motors

The temperature inside the electrical drive is a limiting factor for operation of the motors. In this use case the Digital Twin is used as virtual sensor to sense this temperature. It can be obtained using a model-based simulation. With this information, the availability of the motor can be increased because the motor can be used again earlier. [8] This use case is part of the virtual sensing category.

Dynamic structural health monitoring for an unmanned aerial vehicle

Using a probabilistic graphical model, [9] abstract the "asset-twin system" as a coupled system and use it for scalable structural health monitoring of an unmanned aerial vehicle (UAV). Uncertainty quantification is done with a Bayesian inference framework. One of the listed limitations is the limitation of computational resources which could be solved e.g., via reduced order-modelling. The project has a corresponding Github repository [10] where the Digital Twin is done with ROS [11]. This use case falls into the performance prediction category.

Software for implementation of a Real-Time Digital Twin

Available Digital Twin Solutions

Asset Administration Shell
The Asset Administration Shell (AAS) is the Digital Twin for Industry 4.0. It is curated by the Industrial Digital Twin Association (IDTA). The IDTA was founded by VDMA, ZVEI and Bitkom and twenty companies in 2021 [12]. Today, the number has more than doubled [13]. The AAS in combination with an asset is called an Industry 4.0 component. Assets can be machines or material, but also documents or contracts. It is everything that needs to be integrated into the virtual world for an Industry 4.0 solution [14].

OPC UA
OPC UA, which stands for Open Platform Communications Unified Architecture, is a standard developed by the OPC foundation, which is an industry consortium consisting of more than 850 members, including major players in manufacturing technologies [15]. OPC UA describes itself as a platform independent, service-oriented architecture. It is independent of hardware platforms and an operating system and may run on microcontrollers as well as the cloud. Therefore, it can provide interoperability [16].

OPC UA provides a base information model which is extended by more detailed specifications by the OPC Foundation. Companion specifications are built on top of that, by industry partners agreeing on a common information model for a particular type of asset. For example, the VDMA, Verband Deutscher Maschinen und Anlagenbauer, has working groups developing such companion specs, e.g. for robotics or drive technology [17]. Vendor specific, proprietary information models can then be implemented on top of the companion specification.

Eclipse Ditto™
Eclipse Ditto™ [18] is an open-source software framework to implement Digital Twins. It locates itself as part of the Internet of Things (IoT) and provides a (software independent) IoT middleware that connects and abstracts the devices to and from the applications. It can be used with various communication technologies such as MQTT. It provides access to the Digital Twin via an API and provides authentication among other features. The assets are modelled in Eclipse Ditto™ as Things and can be physical assets or "anything else - if it can be modeled and managed appropriately by the supported concepts/capabilities" [19].

ROS

The Robot Operating System ROS2 [11] is an open-source meta operating system for robotics. ROS can be divided into middleware, algorithms, and developer tools. These developer tools contain, amongst others, tools for configuration, visualization and simulation. The algorithms are suited for robotic applications such as SLAM [20]. The middleware is based on publish / subscribe communication and the ROS graph. This middleware is interesting for building Digital Twins: [9] have used ROS to implement their simulated asset and the corresponding Digital Twin.

MQTT+ Sparkplug

MQTT implements Publish Subscriber (PubSub) communication. It is an OASIS Standard and can be seen as a de facto standard for Internet of Things (IoT) [21]. Originally, MQTT was developed for message transport of SCADA systems and because of this, information about the state of the MQTT devices are built-in. MQTT does not specify what the topics the publishers and subscribers use to communicate should look like, nor does it specify the content (payload) of the messages. This allows for maximum flexibility but is problematic if interoperability between different sectors is to be achieved. For that reason, MQTT Sparkplug was developed (see website [22] or GitHub [23]). It provides a specification of topics and payloads of the messages.

Evaluation Method

This [5] standardizes the Digital Twin framework for manufacturing, pictured below, for the observable manufacturing elements. The Digital Twin Framework as a concept has been introduced in section 2.1 and will now serve, in a more detailed version, as base for the software evaluation. The Digital Twin framework consists of the device communication entity, the Digital Twin entity, the user entity, and a cross system entity.

The observable manufacturing elements communicate with the Digital Twin via the device communication entity which has sub entities for data collection and device control. The information is then processed by the Digital Twin entity. It has the sub entities "operation and management", "application and service" and "resource access and interchange". The Digital Twin entity is the virtual representation of the observable manufacturing elements.

The user entity is managing the access of the Digital Twin entity for user applications such as Human Machine Interfaces, Manufacturing Execution Systems and Enterprise Resource Planning Systems as well as for other Digital Twins.

The evaluation concentrates on a subset of functional entities: for the Digital Twin itself, the Digital Twin entity of course is most important. In the real-time context also the device communication entity is relevant. Therefore, the evaluation focuses on functional entities that would be directly included into a closed loop setup.

The explanations for the individual scores can be found in the corresponding guidebook of the real-time Digital Twin study.

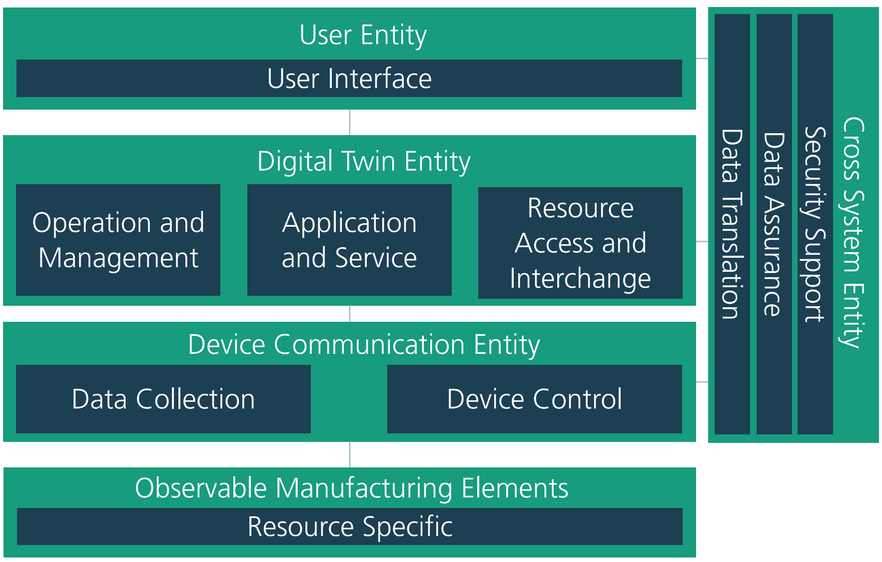

Figure 12: Digital Twin Framework of ISO 23247 [5]

Summary

Summarizing the software evaluation, each of the presented software frameworks has advantages and disadvantages, which will be discussed below: The Asset Administration Shell provides standardized information modelling and is supported by an impressive number of companies, but still lacks maturity and easy deployment. OPC UA is widely adopted in the industry, the information modelling is worked on in various groups and with OPC UAFX, it takes important steps towards real-time capability. Eclipse Ditto connects to the Thing descriptions prevalent in the IoT world and offers APIs for easy communication with the Digital Twin. ROS2, although not intended for use as a Digital Twin, does provide a very promising framework, but it lacks standardized data / information modelling. With Sparkplug, MQTT is starting to move from a communication protocol mostly used in IoT to also advance in the industrial sector and incorporate Digital Twin functionality.

It is not possible to cover every function of the Digital Twin within a single framework. Therefore, just employing one software is not sufficient. Building a Digital Twin requires building the architecture, spanning the whole framework of communication with the assets, and building connections to further applications, such as visualizations, or to other Digital Twins.

The presented frameworks can all play a part in building such an architecture. Which software is most suitable highly depends on the requirements that are placed on the software. It is important to keep in mind the standardization and interoperability, because one of the interesting features of Digital Twins is the data provision and data exchange with various partners.

The Digital Twin functions that can be fulfilled in real-time are very limited so far. For soft real-time systems, all frameworks presented might be sufficient, but this should be carefully evaluated on a use case basis before the implementation. The evaluation at hand can be used as a starting point and additional information on the frameworks and their fulfillment of the functions is available in the guidebook.

For hard real-time use cases, none of the software frameworks can currently be used to implement a Digital Twin off the shelf.

Most promising are ROS2 and OPC UAFX. ROS2 offers a real-time framework intended for robotics with potential for implementing Digital Twins, although this possibility should be explored further. The real-time communication that is already possible via OPC UAFX could be further advanced.

Selected Research Areas

Usability

As of today, the model creation and deployment are mostly done manually, which is not scalable. New service-oriented architectures and solutions for the exchange of data, simulation models, and process knowledge between different stakeholders are missing. Also, the effort of implementing a Digital Twin needs to be reduced. DevOps, the conjunction of development and operation mostly used in software development and deployment, should be applied for Digital Twins also.

Additionally, the usability of the Digital Twin has to be improved. Here, mixed reality techniques as well as low code approaches are promising to make interacting with a Digital Twin easier for humans [8].

Models, simulation, and computation

In addition to being able to exchange models between stakeholders, creating and (re)calibrating models in real time is still a challenge. The fields of scientific machine learning and physics informed neuronal networks could help advance model creation toward a joint approach based on data, AI, and models [8]. The models must be sufficiently accurate, but the cost of computation for high accuracy models is very steep. For techniques like Finite element modelling or computational fluid dynamics, updating them after several hours instead of every second could be sufficient. Other possibilities would be to only simulate local parts instead of the whole system, or use surrogate models, which approximate the complex models to make them easier computable. Another approach is model order reduction [3].

Trust

Especially for (hard) real-time Digital Twins, where the predictions of the Digital Twin are used in a safety or quality critical process, it is important to be able to trust those predictions. This trust must encompass the data, the model, and the procedure.

Trust in data can, to a degree, be enhanced using metadata, by for example including a timestamp, the last calibration, or the type of sensor used. For this, an agreed upon standard on metadata is needed. Furthermore, it needs to be specified what data is really needed for which Digital Twin. Using a wide set of sensors, big data volumes can be used, but finding the relevant data through data reduction techniques is an active field of research.

Trust in the model can be increased by using uncertainty quantification and a validation strategy that encompasses not only one model, but also chains of interdependent models [3].

Conclusion

The topic of real-time Digital Twins promises efficiency gains for mass customization, process optimization, and quality. The use case categories show the different application areas ranging from the estimation of virtual sensor data to performance optimization and virtual testing possibilities. The community use cases show the potential for improving quality and efficiency in production.

This ICNAP study focused on the infrastructure part for real-time Digital Twins, finding that possibilities for real-time communication are available and information modelling has been and still is getting a lot of attention. It helps to address the need for high trust in the data of the Digital Twin.

The presented software comes from different backgrounds, from Industry 4.0 activities, existing communication solutions, standards and from the Internet of Things (IoT) movement. In the end, they all share the same vision of a virtual world that can interact with the physical assets that they are modelling and help support humans in the optimization of processes in networked and adaptive production. On the infrastructure side of things, a lot of research and development is still needed to allow for easy to use, standardized, and scalable solutions. Currently, there is no one framework for building real-time Digital Twins that is able to fulfill all requirements for all use cases. With all presented frameworks building Digital Twins is mostly possible in a soft real-time context. Solutions are available for using predictions in hard real-time contexts, not with a dedicated Digital Twin framework but using models and control theory. But as of right now, challenges for building real-time Digital Twins do not stop with scalable frameworks: Knowledge about processes, uncertainty quantification, and questions about what data to include are still challenges present in the literature. An important issue is also the inclusion of simulations and artificial intelligence in hard real-time environments, such as closed loop control loops.

Real-Time Digital Twins have a lot of potential for industrial production. While a lot of research is still needed, most fields are under active development, and using specific models and carefully selected software, some implementations are already possible today.

References

[1] T. Riedelsheimer, P. Lünnemann, S. Wehking, and L. Dorfhuber, "Digital Twin Readiness Assessment: Eine Studie zum Digitalen Zwilling in der fertigenden Industrie," 2020. https://www.ipk.fraunhofer.de/de/medien/markt-trendstudien/digital-twin-readiness-assessment.html Accessed: Mar. 3, 2023.

[2] T. Bergs, S. Gierlings, T. Auerbach, A. Klink, D. Schraknepper, and T. Augspurger, "The Concept of Digital Twin and Digital Shadow in Manufacturing," Procedia CIRP, vol. 101, pp. 81–84, 2021, doi: 10.1016/j.procir.2021.02.010.

[3] L. Wright and S. Davidson, "How to tell the difference between a model and a digital twin," Adv. Model. and Simul. in Eng. Sci., vol. 7, no. 1, 2020, Art. no. 13, doi: 10.1186/s40323-020-00147-4.

[4] Automation systems and integration — Digital twin framework for manufacturing — Part 1: Overview and general principles, ISO 23247-1:2021, ISO, 2021.

[5] Automation systems and integration — Digital twin framework for manufacturing — Part 2: Reference architecture, ISO 23247-2:2021, ISO, 2021.

[6] J. A. Stankovic, "Misconceptions About Real-Time Computing: A Serious Problem for Next-Generation Systems," Computer, vol. 21, no. 10, pp. 10–19, 1988, doi: 10.1109/2.7053.

[7] Information technology — Vocabulary, ISO/IEC 2382:2015, 2015. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:ed-1:v2:en

[8] R. Gumzei and W. A. Halang, Real-time Systems' Quality of Service: Introducing Quality of Service Considerations in the Life Cycle of Real-time Systems. London: Springer, 2010. doi: 10.1007/978-1-84882-848-3.

[9] D. Hartmann, "Real-Time Digital Twins: A TransContinuum Initiative Use Case," 2021, doi: 10.5281/zenodo.5470478.

[10] M. G. Kapteyn, J. V. R. Pretorius, and K. E. Willcox, "A probabilistic graphical model foundation for enabling predictive digital twins at scale," Nat Comput Sci, vol. 1, no. 5, pp. 337–347, 2021, doi: 10.1038/s43588-021-00069-0.

[11] M. Kapteyn. "UAV-Digital-Twin." https://github.com/michaelkapteyn/UAV-Digital-Twin.git (accessed Mar. 3, 2023).

[12] Open Robotics. "ROS - Robot Operating System." https://www.ros.org/ (accessed Mar. 3, 2023).

[13] Industrial Digital Twin Association e.V. "Gemeinsam mit Plattform Industrie 4.0 die Digitalisierung gestalten." https://industrialdigitaltwin.org/news-termine/gemeinsam-mit-plattform-industrie-4-0-die-digitalisierung-gestalten-218 (accessed Mar. 3, 2023).

[14] Industrial Digital Twin Association e.V. "Who's who – IDTA members." https://industrialdigitaltwin.org/en/about-idta/members-idta (accessed Mar. 3, 2023).

[15] "Plattform Industrie 4.0: Die Verwaltungsschale im Detail von der Idee zum implementierbaren Konzept," Plattform Industrie 4.0, n.d. Accessed: Mar. 3, 2023. [Online]. Available: https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/verwaltungsschale-im-detail-pr%C3%A4sentation.pdf?__blob=publicationFile&v=14

[16] OPC Foundation. "Members." https://opcfoundation.org/members (accessed Mar. 3, 2023).

[17] OPC Foundation. "Unified Architecture." https://opcfoundation.org/about/opc-technologies/opc-ua/ (accessed Mar. 3, 2023).

[18] VDMA. "Welche OPC UA Arbeitskreise gibt es im VDMA?" https://www.vdma.org/viewer/-/v2article/render/1248676 (accessed Mar. 3, 2023).

[19] Eclipse Ditto™. "ditto: … where IoT devices and their digital twins get together." https://www.eclipse.org/ditto/index.html (accessed Mar. 3, 2023).

[20] Eclipse Ditto™. "Thing." https://www.eclipse.org/ditto/basic-thing.html (accessed Mar. 3, 2023).

[21] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot Operating System 2: Design, architecture, and uses in the wild," Science robotics, vol. 7, no. 66, 2022, Art. no. eabm6074, doi: 10.1126/scirobotics.abm6074.

[22] A. Helmbrecht-Schaar. "Sparkplug – eine auf MQTT basierende Spezifikation für das IIoT." https://www.informatik-aktuell.de/betrieb/netzwerke/sparkplug-eine-auf-mqtt-basierende-spezifikation-fuer-das-iiot.html (accessed Mar. 3, 2023).

[23] Eclipse Foundation. "MQTT + Sparkplug = ‚Plug & Play' IIoT: The Eclipse Sparkplug Working Group ensures open and interoperable IIoT." https://sparkplug.eclipse.org/ (accessed Mar. 3, 2023).