Digital Twins for Scaling up Hydrogen Electrolysis

Sharaf Alsharif[a], Nils Huxoll[a], Amin Raeiszadeh[a], Tobias Brandt[a], Michael Brand[a], and Sebastian Lehnhoff[a]
[a]OFFIS e.V., Oldenburg, Deutschland
{sharaf.aldin.alsharif, nils.huxoll, amin.raeiszadeh, tobias.brandt, michael.brand, sebastian.lehnhoff}@offis.de

Abstract

In the context of the energy transition and the development of a climate-neutral economy, there is a need to immensely increase hydrogen production. Producing green hydrogen with water electrolysis represents a great potential to improve the integration of renewable energies into the power grid. Scaling up hydrogen production capacity is a significant challenge on several levels. Hydrogen electrolysers need to be mass produced in an automated manner in order to meet the projected hydrogen demand in the medium to the long term. Digital twin technology offers a promising digitalisation solution to meet this challenge by facilitating the manufacturing and operation processes for hydrogen electrolysers. In this paper, we propose a software architecure for digital twins of electrolyzers as a future-proof method of implementing digitalization. The proposed software architecture utilizes data streams from hydrogen electrolyzers to analyze production and plant operations in both the long-term and near real-time, allowing for rapid reactions to unforeseen events. The aim of this approach is to optimize hydrogen production and improve plant operations.

1 Introduction

The shift to more carbon-neutral energy sources increases the need to build a comprehensive hydrogen economy as a pillar to replace fossil fuels. This gives rise to a number of challenges [1]. One of these challenges is the rapid upscaling of production and the evolution of manufacturing processes from many manual steps toward automated mass production [2]. At the same time, intensive research and development is taking place on the structure and materials used within the electrolyzers [2]. Long-term tests are difficult to perform or simulate in a short time span where most of the elctrolysers haven't reached end of life. Therefore, it is of high importance to make data from the plants in operation available for the development to be able to draw conclusions about the properties in operation and thus support future development. The use of digital twins as virtual representations of physical entities such as electrolyzers is a promising approach. Digital twins facilitate access to the operation data of electrolyzers throughout their entire life cycle and to enable analyses of performance characteristics [3].

Although there is no uniform definition of digital twins, they are in most cases more than virtual representations of physical entities. In this paper, a digital twin is qualified by enabling a live remote monitoring and control with an automated bi-directional data connection to the physical entity. Furthermore, it can be distinguished between a digital twin, shadow, and model as shown in **Figure 1** [4]. While a digital twin is featured with an automated bi-directional data connection to the physical entity, a digital shadow only has an automated data connection from the physical entity to the shadow (read direction) [4]. A digital model, however, is not featured with any automated data connection [4].

An important performance characteristic for electrolyzers is their degeneration and their predicted remaining useful life [2]. Based on an automated assessment of the state of health of electrolyzers over their life span so far, new insights in their degeneration can be gained. Hence, the development, production, and maintenance of electrolyzers can be improved [4]. However, a digital twin should be capable of enabling more than the analysis of degeneration, it should also enable different high level applications. A digital twin should also be able to add applications which might not even be thought of during the development of the digital twin later on. Therefore, a digital twin should be designed modular and flexible to be able to deal with different objectives.

In this paper, a modular and flexible architecture for digital shadows and twins is presented. The architecture also features the paradigm of an event-based processing that enables a live remote monitoring and control of the physical entity. The architeture not only enables the digital twin to foster the up-scaling of hydrogen electrolyzers but also abstracts from the concrete application and can be applied to other applications of live monitoring and control.

The remainder of this paper is structured as follows. Section 2 provides an overview of related work. Section 3 describes the non-functional requirements of the architecture. The proposed architecture is described in section 4 and the concept of containerization is explained in section 5. Future work is discussed in Section 6 and the paper is concluded in Section 7.

2 Related Work

The term digital twin is used in different ways in the literature. In the following we will give an overview of the main usages and how we use the term in our work. The early formulation of the digital twin concept was presented by

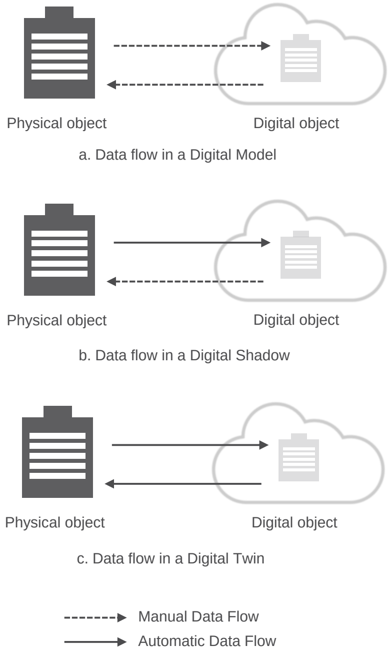

Figure 1 Differences between a digital model, a digital shadow and a digital twin, reproduced and adapted from [4].

Grieves in 2003 [5]. The definition of a digital twin consists of three main components of a digital twin, namely the physical component in real space, the virtual component in virtual space, and the connections of data and information that ties the virtual and real components together [6]. Tao et al. [7] proposed an extended concept with the additional components “data” and “services”.

The aforementioned definitions describe the components as part of a digital twin and their relations, but they do not describe how the digital twin is build up on an architectural level. In the literature, digital twins have been designed with many different architectural patterns, such as layered, service-oriented, component-based, microservices and so on, with the layered pattern as the most recurring pattern followed by service oriented [8].

Different architecture patterns support different requirements of the digital twin such as performance efficiency, compatibility, usability, reliability, maintainability, and scalability [8]. The requirements for a digital twin architecture pattern differ per application. The work in [8] also address these requirements and shows that layered and service-oriented architecture patterns are used the most with the goal to achieve the mentioned requirements.

Another perspective for the definition of a digital twin is the data flow as described in the study in [4]. The authors classify digital twins into three subcategories according to their level of data integration: Digital Model, Digital Shadow, and Digital Twin as shown in Figure 1. According to [4], the Digital Model is a representation of the object of interest without any form of automated data exchange with the physical object, however, manual data flow is possible in this case. The Digital Shadow term is used when an automated data flow from the physical object to the digital object exists but not vice versa [4]. In a digital twin, an automated data flow exists in both directions between the physical and the digital object, and the state of the digital object affects the state of the physical object [4].

A definition that summarizes the characteristics of a digital twin with a strong focus on real-time data and adaptability of the models used in a digital twin was given by Semararo et al. [3]:

“A set of adaptive models that emulate the behaviour of a physical system in a virtual system getting real time data to update itself along its life cycle. The digital twin replicates the physical system to predict failures and opportunities for changing, to prescribe real time actions for optimizing and/or mitigating unexpected events observing and evaluating the operating profile system.”

The use of digital twin technology in the field of electrolysis can be found in the literature, though not ubiquitous. The work in [9] applies the digital twin concept for prognostics and health management of fuel cells. The authors construct a data-driven digital twin to integrate the physical knowledge of the system into a deep transfer learning model. The authors implement the digital twin in a layered architecture comprising of a physical, a connection and a digital layer. The study focuses on the accuracy of the suggested models but not on the architecture of the digital twin.

The authors of [10] highlight the potential of the Functional Mockup Interface (FMI) standard in the development of the digital twin. The authors discuss a strategy of building digital twins from individual Functional Mockup Units (FMUs) to enable the exchange of sub-models in digital twins without the adjustments of the interfaces. An example for the proposed process is given by the composed simulation model of a hydrogen generation process based on wind energy. The work demonstrates the advantage of the FMI standard to provide interoperability in a digital twin. However, the work does not address the scalability of digital twins or real time requirements.

The work in [11] presents a short summary of the architecture design which represents the seed idea for this work and hence further development, design and requirements are elaborated in this work.

Furthermore, in this work we focus on designing the architecture for digital twins that is suitable for use cases such as hydrogen electrolysis, where a huge amount of data is expected to be exchanged with the digital twin in real time. More details on the requirements for such use cases are discussed in the following sections.

3 Requirements

The digital twin in this work aims to enable a live monitoring and control of a remote physical entity. Thus, the core functionalities of the digital twin are, first, to provide an automated bi-directional live data connection with the remote physical entity, second, to calculate or estimate the state of the physical entity, and third, to enable remote control.

Services, such as a human-machine-interface, concrete control algorithms, remaining useful life, or predictive maintenance are enabled by these core functionalities. However, services themselves are not part of the core of a digital twin. Instead, they are part of a service ecosystem that makes use of the state of the physical entity provided by the digital twin, historical data and any other available data. Such a modular design, i.e. with a thin digital twin core and an extensible service ecosystem, ensures flexibility, scalability, and maintainability. A comprehensive overview of the identified requirements is provided in this section.

The following requirements have been considered for the use case in hydrogen electrolysis, however, they also allow the use of such a digital twin for other applications such as live monitoring and control of energy systems.

- **Scalablity**: The ability to handle a growing amount of data, services and communication between digital twin elements.
- **Modularity**: Different components of the digital twin and services act as decoupled single functioning modules that can act together to achieve higher level tasks.
- **Efficiency**: Achieving the requirement of a service with minimum response time and utilize the available resources optimally.
- **Flexibility**: The ability to change parts of the architecture.
- **Extensibility**: The ability to add new services to the digital twin.
- **Maintainability**: The ability to modify the digital twin to "correct faults, improve performance or other attributes" [12].
- **Adaptability**: The ability to adapt the digital twin for research and possible industry users.
- **Self contained services**: Services can operate independently in their own containers or runtime environments.
- **Interoperability**: The ability to integrate models and services implemented in various programming languages.
- **Near real-time**: The data as well as control commands between the physical and the digital twin need to flow as quick as possible, so that the digital twin reflects the current state of the electrolyser. Additionally, this shall enable the digital twin to take proper actions in timely manner.

4 Digital Twin Architecture

The concrete proposed architecture is shown in **Figure 2** and the individual components are explained in the following sections. At the center of Figure 2 is the message orchestrator, which is described in Section 4.1. One source and target at the same time of the message orchestrator is the connection framework (Section 4.2). The connection framework's task is to establish the bi-directional data connection with the physical entity in (nearly) real-time. On the right side in Figure 2 is the state estimation component, responsible for proving the state of the physical entity based on the live measurements (Section 4.3). A data management system, which may consists of several different databases, and a service framework, which may also consists of several different services, are described in Section 4.4 and Section 4.5, respectively.

When considering an electrolyzer, there are multiple sensors directly measuring the parameters of the plant as well as possibly some further sensors inside the factory/environment that monitor external factors. Those sensors might have different polling rates as well as data formats. Thus, during the pre-processing step in the data stream system, the problem is handled by converting the data to a predefined format. With this data, the state estimation can run and estimates the parameters that are not directly measurable or even missing. This is done in near real-time using stream processing. As the plant runs, internal parameters of the plant might change due to the operation, degradation or other influences. Hence, the state estimation model needs to be adapted. The parameter drift can be estimated offline and an updated model for the state estimation can be used by the stream process.

Next, the measurement as well as the state data has to be made available for the services as well as being saved in a time-series database. Considering the availability for other services, the data is written onto a publish-subscribe system, where it can be read from any subscribing service. Such services can access the live data as well as the stored data to provide their functionality. The Service Framework can include services such as the aforementioned provision of flexibility for power systems, degradation prognosis for control actions or predictive maintenance.

The results of the services are again made available on the publish-subscribe system, as other services might rely on those results. In addition, the data will be pulled by the Data Management System and stored in the appropriate database.

As new services are developed or old ones become obsolete, the service-based architecture, with a strong focus on containerization can handle the addition, removal or update of services easily. The service must only be able to access the publish-subscribe system and read or write data in a suitable format.

4.1 Message Orchestrator

A central component in the digital twin architecture as shown in Figure 2 is the message orchestrator. The exchange of data between different elements of the digital twin is realized with a publish-subscribe system. Message sources (publishers) send messages to a message broker, which then distributes them to message sinks (subscribers) that have opted to receive messages on a specific topic. A publish-subscribe environment is essential to provide real-time streaming data and event-driven applications within a

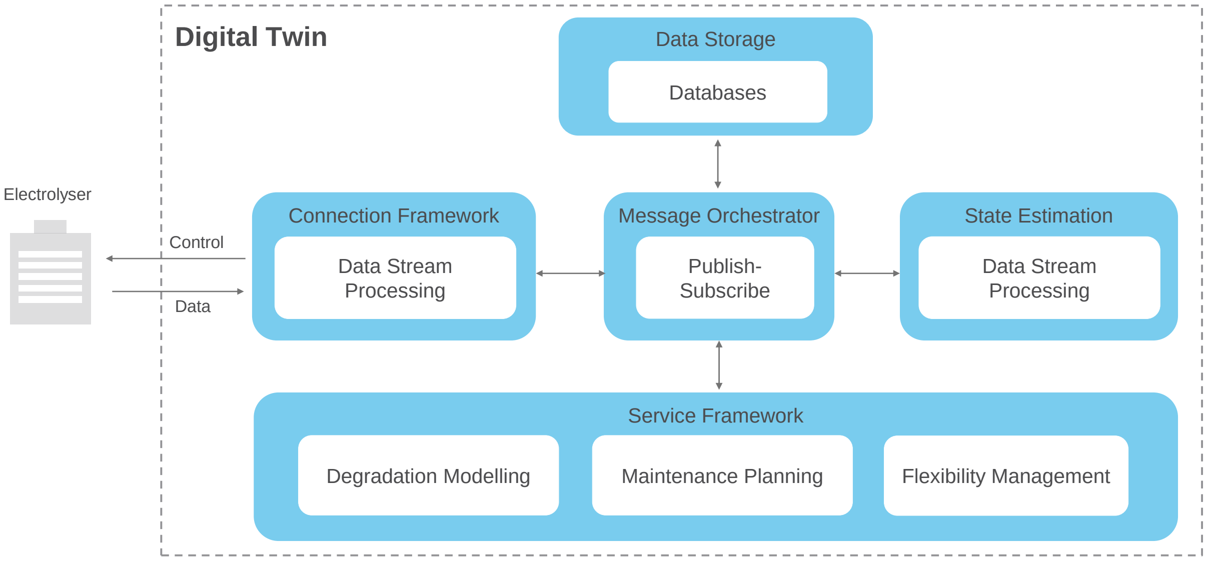

Figure 2 Overview of the digital twin architecture with connection to the physical entity.

digital twin. Furthermore, the publish-subscribe environment enables a decoupled communication between many services. One service may post sensor data from the electrolyzer, for instance, while another might subscribe to that data and use it to send an alarm when a measurement goes beyond a certain limit. This kind of event-driven methodology enables more responsive and dynamic behavior by enabling various services to respond to changes in real-time. Additionally, avoiding direct communication between services reduces the complexity and avoids strong coupling of the communication and ensures extensibility and exchangeability. In addition, publish subscribe systems such as Apache Kafka [13] can offer a high level of reliability and scalability for the datastreams.

4.2 Connection Framework

The Connection Framework that enables the bi-directional live communication with the physical entity has mainly three requirements. First, it needs to be able to deal with external active data sources outside of the control of the digital twin, i.e. data is pushed from the physical entity to the digital twin. Second, it should be able to deal with different communication and data protocols, hence, it should be an extensible framework that is able to translate arbitrary protocols into an internal data format. Third, scalability regarding the amount of external data sources is important. The physical entity may be equipped with multiple sensors providing measurements via different communication channels.

To fulfill these requirements, a data stream management system, namely Odysseus [14], is proposed. Odysseus is a modular framework to create individual data stream management systems. Data stream management systems combine an event-based processing (push and pull) with management features known from database management systems [15]. Odysseus is equipped with a so-called access framework, consisting of adapters for various communication and data protocols. Since it is also designed to be extended, it fits the needs of the connection framework for this digital twin architecture.

4.3 State Estimation

The goal of the state estimation component is to estimate the current state of the physical entity in an event-driven manner, i.e., whenever new measurements are available. Typically, it is only an estimation because of the following reasons. First, the values defining a state are often not directly measured and need to be either calculated or estimated based on the available measurements. For complex non-linear systems it is often an approximation. Second, single measurements may be incorrect due to deviations of the metering devices, errors or even cyber attacks. A challenge in this context is to provide the estimated state event-driven and in near real-time. Typically, a state estimation requires measurements from all available sensors and, thus, is implemented as a batch process.

In the context of hydrogen electrolyzers, an essential part of the state is the so-called state of health, which is the foundation for other services such as the calculation of the degradation and predictive maintenance.

For the state estimation algorithms, different kinds of models can be applied as shown in **Table 1**. Each kind of model has its requirements, advantages and drawbacks. First-principle models, on the one hand, work with mathematical equations (e.g., modeling the physico-chemical processes of an electrolyzer) and require, therefore, all parameters (measurements) required for the model as well as expert knowledge for building the model. They are qualified by a high accuracy but can be slow compared to other models for complex non-linear systems. Data-driven models, on the other hand, make use of machine learning. Hence, ex-

Model	Requirements	Advantages	Drawbacks
1^{st} principle	• Knowledge • Sufficient parameters	Accuracy [16]	Speed [16]
Data-driven	Historical data	Speed [16]	Accuracy [16]
Combined	See above	• Faster than 1^{st} principle [17] • More accurate as data-driven [17]	• Slower than Data-driven [17] • Less accurate as 1^{st} principle

Table 1 Different possible models for a state estimation.

pert knowledge about the physical system is not required but historical data in the scale of big data. Once a data-driven model is learned, its application of the live data is comparatively fast. Its drawback is, compared to first-principle models, a lack of accuracy.

Which kind of model suits best depends on the requirements of the concrete application. However, another possibility is to combine both kinds of models. Unavailable parameters or very slow pars of a first-principle model can be derived from a data-driven model. Such a combination is then most probably in between the other two options regarding accuracy and speed or is only the enabler for first-principle models in case of missing required parameters.

However, a digital twin should support the parallel use of both or all three kinds of algorithms and let the respective services decide, which solution fits best to their requirements.

4.4 Data Storage

The data management stores the data for long-term use cases such as the subsequent analysis of the performance of a plant in relation to the manufacturing conditions. Mainly a time series database such as InfluxDB [18] is used here.

However, structured databases as well as non-relational databases are expected to be part of the data management of the digital twin. In short, structured databases store data in tables with rows and columns include relational databases [19]. These databases support advanced Structured Query Language (SQL) querying capabilities and are well suited for storing and querying structured data, such as design and engineering data. NoSQL databases, also referred to as non-relational databases, are used to store data that is challenging to model in a conventional, tabular structure and are designed to handle unstructured or semi-structured data such as documents or human generated reports.

4.5 Service Framework

As mentioned in the requirements, the digital twin should be extensible. Hence, depending on the use case, necessary services can be inserted and work with the existing data infrastructure. Examples here are degradation modeling and offering flexibility services . The exchange among the additional services also works via the publish-subscribe environment of the digital twin. An exemplary use case for collaboration among multiple services is flexibility offering based on degradation estimates of individual control actions. The flexibility management service, using the degradation estimation service, checks to what extent an action to offer flexibility (e.g., turning off the plant for a certain period of time) ages the plant. If the remuneration of the flexibility is in a favorable relation to the aging, it can be offered. If the offer is accepted, the digital twin is able to change the operation mode of the physical entity.

Another exemplary use case for extended service is performing planning and operation simulation. Different control scenarios of the electrolyser can be simulated to find out the best operation scenario in terms of aging for example. Such simulations allow the digital twin to experiment with scenarios before performing them in reality.

5 Containerization

Digital twin technology and service-based software architecture can benefit from the concept of containerization, e.g with Docker [20] in a number of ways. One advantage of containerization using Docker is the simplicity with which lightweight, isolated environments known as "containers" that contain all the required code and dependencies for a particular service can be easily created and deployed [20]. This facilitates consistent and repeatable service development and testing. In a digital twin scenario where the workload on a specific service may change over time, containers' ease of scaling up or down to meet changing demand can be helpful.

The simplicity with which services can be deployed and managed across numerous servers or cloud environments is another advantage of using containers. This can be especially helpful in the case of a digital twin where services may need to be distributed across a network of devices. Using containers, it is simple to deploy and manage the same service across various environments, ensuring that every instance is using the same code and is configured correctly. A digital twin system's complexity and maintenance requirements may be reduced as a result.

6 Future Work

The current implementation of the digital twin is being tested with artificially generated data. A connection to data from real plants is to be made together with the project partners. In addition, the majority of the implementation tasks are still pending in the project. While the data acceptance, the publish-subscribe system, the data management as well as a simple visualization are already implemented for first tests, the implementation of the state estimation and further services is still pending. In the case of state estimation, there is still the particular challenge of integrating different models via a common interface for all models, as well as the automated selection of the appropriate model for different use cases. In order to simplify prototyping in the future and thus to accelerate development on the services and the models, a simulation environment is also to be set up in which data generation can also be tested with feedback from the digital twin, for example for control commands in the flexibility example.

7 Conclusion

The impact of the climate change becomes more severe and forces the industry and nations to focus stronger on decarbonisation of the industry. A big contender to accelerate this decarbonisation is hydrogen generated from renewable energies. Hydrogen can be utilized in various ways to support decarbonization. It can be used in processes to create green replacements for oil products in the primary industry or to produce fuel replacements as well as for high temperature processes such as steel production. For this, electrolyzer production needs to shift from manual production to series production as well as improvement of operation. In order to optimize the production and operation of the electrolysers, data from the electrolysers is collected and analyzed. The concept of the digital twin is used here to make this possible. Engineers can utilize the data from begin of life until end of life of an electrolyzer in conjunction with maintenance and operational data to improve the longevity and efficiency of electrolzyers. Furthermore, the calculation of the current state in the state estimation as close to real-time as possible is of high importance to ensure safe and reliable operation. This requires the use of different models that can be integrated in the digital twin and used for different services. This also enables more advanced use cases, such as the integration of electrolysers into a flexibility market. In general, the reliability, availability, and performance of electrolyzers can be improved with the use of digital twins, which will ultimately result in higher productivity and profitability. For this purpose, this work presents a service based digital twin architecture. The requirements such as scalability, modularity and extensibility are discussed. The concrete design of such an architecture is presented and explained in detail, highlighting the advantages and possible use cases in the context of hydrogen electrolysis.

Funding

This work is part of the projects DERIEL (De-risking Elektrolyseur) funded by the BMBF under reference numbers: 03HY122F.

Acronyms

FMI Functional Mockup Interface. 2

FMU Functional Mockup Unit. 2

SQL Structured Query Language. 5

References

[1] Bundesministerium für Wirtschaft und Energie (BMWi), “Die nationale wasserstoffstrategie 2020,” 2020. [Online]. Available: https://www.bmwk.de/Redaktion/DE/Publikationen/Energie/die-nationale-wasserstoffstrategie.html

[2] Bundesministerium für Bildung und Forschung, “Wasserstoff leitprojekte: H2giga: Serienfertigung.” [Online]. Available: https://www.wasserstoff-leitprojekte.de/leitprojekte/h2giga

[3] C. Semeraro, M. Lezoche, H. Panetto, and M. Dassisti, “Digital twin paradigm: A systematic literature review,” *Computers in Industry*, vol. 130, p. 103469, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361521000762

[4] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, “Digital twin in manufacturing: A categorical literature review and classification,” *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896318316021

[5] M. Grieves, “Digital twin: Manufacturing excellence through virtual factory replication,” *White paper*, 03 2014.

[6] ——, “How a ‘digital twin’ can warrant products are built as designed,” 2015. [Online]. Available: https://www.manufacturing.net/operations/article/13240878

[7] F. Tao, M. Zhang, Y. Liu, and A. Nee, “Digital twin driven prognostics and health management for complex equipment,” *CIRP Annals*, vol. 67, no. 1, pp. 169–172, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0007850618300799

[8] E. Ferko, A. Bucaioni, and M. Behnam, “Architecting digital twins,” *IEEE Access*, vol. 10, pp. 50 335–50 350, 2022.

[9] S. Meraghni, L. S. Terrissa, M. Yue, J. Ma, S. Jemei, and N. Zerhouni, “A data-driven digital-twin prognostics method for proton exchange membrane fuel cell remaining useful life prediction,” *International Journal of Hydrogen Energy*, vol. 46,

no. 2, pp. 2555–2564, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360319920339252

[10] M. Wiens, T. Meyer, and P. Thomas, "The potential of fmi for the development of digital twins for large modular multi-domain systems," *Proceedings of 14th Modelica Conference*, vol. 46, no. 2, pp. 2555–2564, 2021. [Online]. Available: https://ecp.ep.liu.se/index.php/modelica/article/view/200

[11] N. Huxoll, S. Alsharif, J. C. Diettrich, T. Brandt, and M. Brand, "Digital twin architecture and technologies for hydrogen electrolyser applications," *Abstracts of the 11th DACH+ Conference on Energy Informatics*, 2022. [Online]. Available: https://doi.org/10.1186/s42162-022-00215-6

[12] I. Board, "Ieee std 610.12-1990 standard glossary of software engineering terminology," *New York Inst. Electr. Electron. Ehgineers*, 1990.

[13] Apache Software Foundation. (2022) Open-source distributed event streaming platform. [Online]. Available: https://kafka.apache.org/

[14] H.-J. Appelrath, D. Geesen, M. Grawunder, T. Michelsen, and D. Nicklas, "Odysseus: A highly customizable framework for creating efficient event stream management systems," in *Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems*, ser. DEBS '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 367–368. [Online]. Available: https://doi.org/10.1145/2335484.2335525

[15] S. Geisler, "Data Stream Management Systems," in *Data Exchange, Integration, and Streams*, ser. Dagstuhl Follow-Ups, P. G. Kolaitis, M. Lenzerini, and N. Schweikardt, Eds. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2013, vol. 5, pp. 275–304. [Online]. Available: http://drops.dagstuhl.de/opus/volltexte/2013/4297

[16] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "Integrating scientific knowledge with machine learning for engineering and environmental systems," 2020. [Online]. Available: https://arxiv.org/abs/2003.04919

[17] O. Erge and E. van Oort, "Combining physics-based and data-driven modeling for pressure prediction in well construction," in *Proceedings of the 19th Python in Science Conference*, 01 2020, pp. 125–131.

[18] InfluxData, "Influxdb time series database." [Online]. Available: https://www.influxdata.com/products/influxdb-overview

[19] E. F. Codd, "A relational model of data for large shared data banks," *Commun. ACM*, vol. 13, no. 6, p. 377–387, jun 1970. [Online]. Available: https://doi.org/10.1145/362384.362685

[20] D. Merkel, "Docker: lightweight linux containers for consistent development and deployment," *Linux journal*, vol. 2014, no. 239, p. 2, 2014.