5th Conference on Production Systems and Logistics

Conceptualizing A Digital Twin Based On The Asset Administration Shell For The Implementation Of Use Case Specific Digital Services

Henry Himmelstoss[1], Roland Hall[1], Bernd Vojanec[2], Paul Thieme[1], Thomas Bauernhansl[1,3]

[1] *Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Nobelstraße 12, 70569 Stuttgart, Germany*
[2] *Wittenstein SE, Walter-Wittenstein-Str. 1, 97999 Igersheim, Germany*
[3] *Institute of Industrial Manufacturing and Management, University of Stuttgart, Allmandring 35, 70569 Stuttgart, Germany*

Abstract

In the context of ongoing digitalization, manufacturing companies face new challenges and the need to expand their portfolios to include new digital services. Enriching their portfolio with digital services can be an opportunity for manufacturing companies to position themselves in emerging collaborative production networks and thus make their business model fit for the future. The paper ties in with the activities from the Product Lifecycle Enrichment as a Service (PLCEaaS) research project. Within this context, use cases for digital services have already been derived, modelled, and documented in various workshops. The respective digital services directly address external customers and internal stakeholders from development. The central enabler for the new digital services is the digital twin based on the asset administration shell, which makes the necessary data available and thus supports interoperability. The asset administration shell is enriched with use case-specific submodels. The procedure for structuring these submodels is shown in this paper using the research project as an example. This includes modelling the digital services with a standardized modelling language based on the semi-structured use cases described so far. As a result, we obtain an asset administration shell enriched with several submodels - some of which may be based on standardization activities already underway or represent proprietary submodels. Likewise, it is considered whether more submodels are required to implement the domain-specific use cases that are currently not yet addressed in standardization activities. The paper ends with an outlook on the further research activities that are necessary to prototype the planned project and describes which criteria can be used to evaluate the defined submodels in the later course of the project.

Keywords

Digital Twin; Asset Administration Shell; Smart Service; Digital Manufacturing; Industry 4.0

1. Introduction

The ongoing digital transformation is also changing the engineering industry and should be understood as a process that operates at different levels through the use of various technologies and concepts. New possibilities are now available for gathering product data along the life cycle of a product. The digital twin (DT) [1] supports this because it can integrate the physical and virtual data throughout the product lifecycle. The DT is increasingly emphasized by both academia and industry. The lifecycle data can be used for various types of analytical methods, for example, to optimize the performance of physical products or processes [2]. The DT can be used to tackle the challenges of information islands and duplicate data between different

DOI: https://doi.org/10.15488/15285
ISSN: 2701-6277

phases of the product lifecycle. Therefore it has high potential in product design, manufacturing, and product service [3]. Early definitions characterize DT as a set of virtual constructs that describe a product – actual or virtual – in different levels of detail. It includes the use of operational states from sensor data or from predicting models for a variety of purposes [4]. The DT therefore ties information of physical and virtual products together [5]. Since the project described in this paper relies on a digital twin based on the asset administration shell (AAS), the digital twin is defined as a digital representation sufficient to meet the requirements of a set of use cases. The AAS, in turn, is the standardized digital representation of an asset that enables interoperability between applications and the management of manufacturing systems. Assets represented by it are uniquely identifiable, contain various digital models – called submodels – and describe the functionality of the assets [6]. It characterizes the technological attributes of an asset, central to the Reference Architectural Model for Industry 4.0, often referred to as RAMI 4.0 [7].

This paper builds on a functional service model created without a standard modelling method. To obtain this functional model, we built on an abstract idea for servitising an existing business model of a component manufacturer. This abstract idea arose from a business opportunity addressed as part of a research project, which in turn, is exploring new manifestations of everything-as-a-service approaches. To sharpen the abstract idea of services, the component manufacturer's product life cycle was examined. Subsequently, user stories were created for the respective stakeholders, who could be external customers or internal employees, depending on the service. For the user stories, word templates were used for systematic elaboration. To prioritize the user needs expressed by the user stories and the service functionalities, the quality function deployment (QFD) method was applied. For this purpose, the user stories were first weighted in a pairwise comparison. The services' functionalities were defined as precisely as possible by various domain experts. Next, the correlation of the customer's voice in the form of the weighted user story was correlated with functionality by asking, for each comparison, what effect does the fulfilment of service functionality X have on user need Y? The result is the prioritization of service functionalities for further elaboration. Afunctional model was developed with an online tool that enables a collaborative work mode with the simultaneous participation of each team member.

In this paper, we present a procedure from the defined service idea to the actual data model of the digital twin. This does not include its validation or implementation in the service, as this represents a later step in the research project. We thus address the research question: How to model the data model for the digital twin based on the AAS for a use case-specific service?

2. Need for a procedure for creating a digital twin

According to the aforementioned definition of the DT, meeting the requirements of use cases is a central point determining its overall quality and meaningfulness. One can proceed by deriving the DT data model based on an already existing use case or by first deriving a data model and then validating it against the use case requirements. In this research project, we decided to first model the use cases in digital services in using an iterative and systematic approach. The Industrial Digital Twin Association e. V. has published a guideline for creating a submodel template. One of the topics addressed is that different stakeholders and experts should be involved in creating a submodel template. In this project, research team also consists of experts from different domains. However, the guideline does not address how to build the AAS with the necessary submodels for specific use cases - this includes the creation of proprietary submodels or the selection of existing submodels [8]. Liu et al. also describe the problem that despite the long existence of the DT terminology, in addition to the lack of a unified definition, there is also no unified creation and deployment process [9]. Regarding the DT product-service systems, Bertoni indicates that research projects often focus on the development of standalone applications. Also, the focus is often on the development of frameworks and methods rather than on models, tools, or algorithms. Bertoni explicitly poses the research question of how virtual product models and the DT should be developed as a link between the real and virtual worlds to support decision-making in the early design stages [10]. There is already work on the automatic generation of AAS by using neural language models. However, this approach requires that there are already information sources whose contents can be mapped into standardized submodels of the AAS [11]. Also described is that there remains a poor understanding of what data should be collected from the usage phase of product-service systems (PSS) that can be used to design better PSS [12]. Lack of knowledge about the data to be collected inevitably leads to lack of knowledge about the data structure of a DT and the properties to be contained in it.

Acatech, the German Academy of Engineering Sciences, addresses further specific research needs. Primary research needs have been defined, such as knowledge provision based on feedback information from earlier product generations or approaches to knowledge generation from product usage data for new requirements specifications [13]. Later, operational data usage was addressed associated with the DT to optimize or develop systems. It was noted that implementation and use in practice, however, is still very much in its early stages [14]. In this research project, services are also being worked on to support the data-based optimization of product sizing and the development of products. For such use cases, it has also already been described that AAS, in particular, is a possible technology as a manifestation of DT [15].

We present our process to address challenges like the ones mentioned. By presenting our procedure, we want to share our approach for a systematic derivation of the DT. In our research project, the DT is based on the AAS. Hence it concludes that this is a matter of systematically deriving the data model and composing it from different submodels of the AAS. This is also intended to address the gap between the functional view of a defined use case and a potentially standardized modelling DT to enable that very use case. As mentioned before, meeting requirements for a digital representation is central to our used definition of a digital twin. Therefore, our approach is strongly driven by requirements engineering and by the related aspects to model a use case as holistically as possible [16]. The modeling of the submodels also depends in particular on the type of asset, the lifecycle and the deployment scenario [17]. With regard to the metamodel, it should become clear, among other things, which submodels are necessary, what its properties are or whether the asset is a type or an instance.

3. Modelling of the use cases

The procedure described in the following is based on functional use case descriptions. However, regarding data structure and flow, these use case descriptions still represent black boxes. The goal of the modelling was to move from this black box approach to a detailed view creating a deeper detailed understanding of the services. The Unified Modelling Language (UML) was chosen because it is internationally established as a uniform modelling language for information systems and is, thus, considered an adequate tool for the research group, which consists of experts from different domains [18].

The modelling of the services consists of three parts. First, the system structure is addressed. The system is modelled together with all components relevant for implementing the use case - corresponding to a Tier 3 architecture. The AAS functions here as an access layer that, for the prototypical implementation, references the operating data in a separate database and already contains the catalogue data of the components. In the future, it is also possible to reference data in other data sources or carry additional information. Thus, the structure and relationships of the components are documented and readable. The component diagram already contains all components that are important for the project's various services. Figure 1 shows an illustration of the component diagram in which the Tier 3 architecture for the services was designed.

The respective services were modelled separately. Only the relevant components were transferred in sequence diagrams in the next step. The sequence diagrams reveal the mutual interactions between components and the services' functionalities. This shows which procedures use the AAS. If applicable, it should also be made clear which functions are called up via the AAS or which information is retrieved from it. Figure 2 illustrates a sequence diagram describing the processes of "Innovation-as.-a-Service".

Last, but not least, the services are modelled in class diagrams. They are particularly valuable for modelling the objects relevant for the service together with their attributes and for defining the relationships between the objects. From them, it should be evident which properties must be included in the data model DT. In the use case considered here, the data relevant for the following evaluations is collected in a targeted manner. However, a broader database may also be available, and an explorative approach to a project can be pursued.

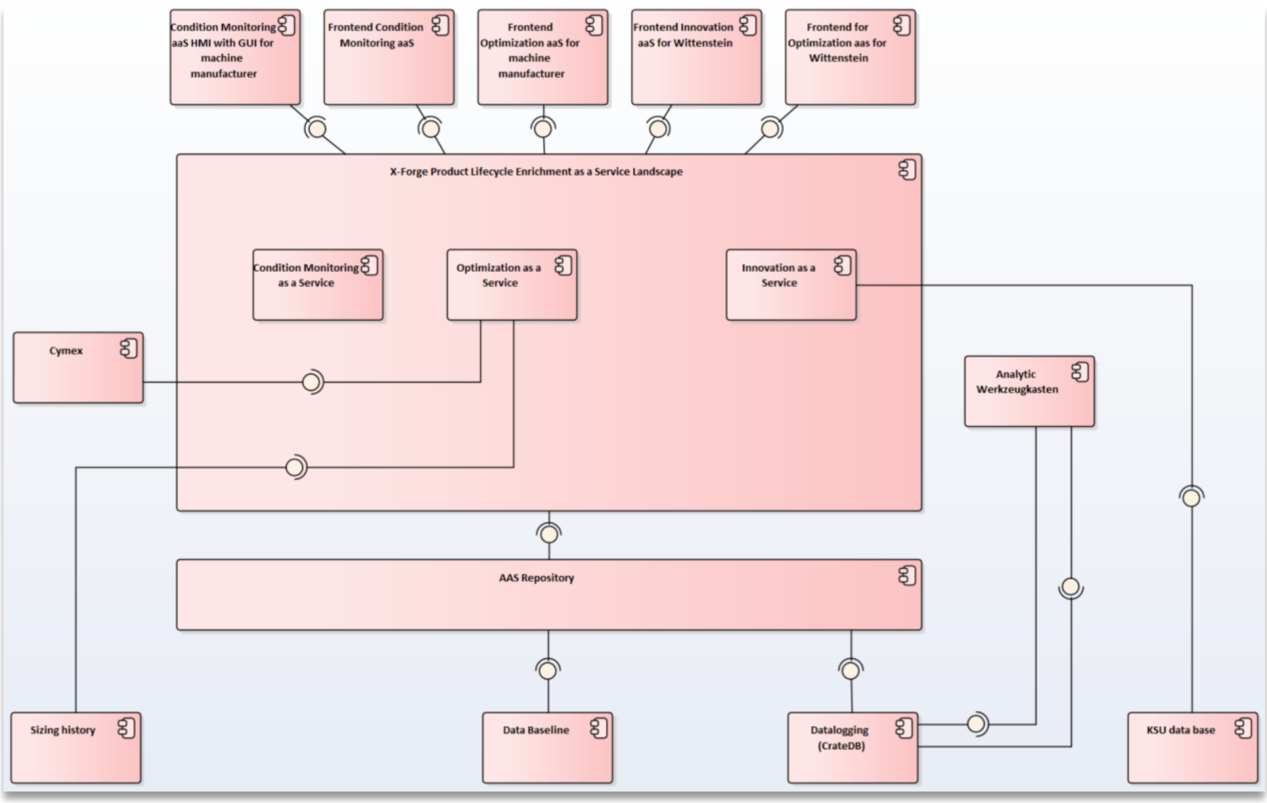


Figure 1: Component diagram of the Tier 3 architecture for the service landscape of the research project

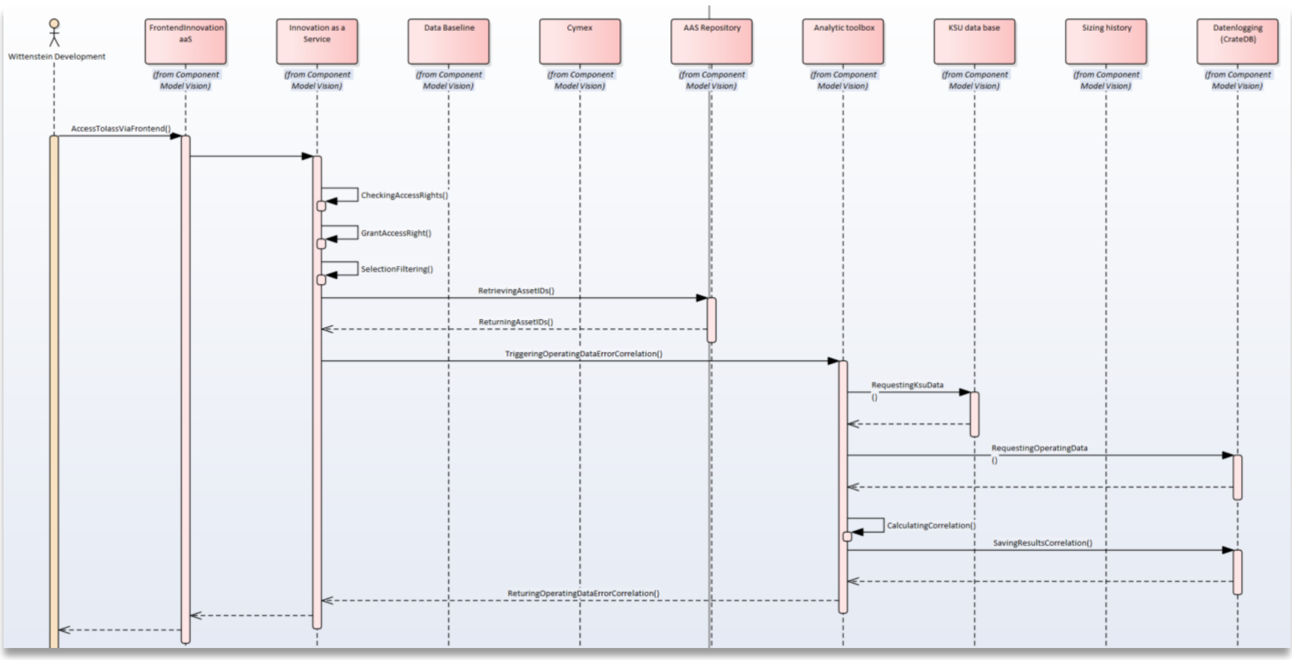


Figure 2: Excerpt from the sequence diagram for "Innovation as a Service"

In this case, a detailed consideration should clarify whether all properties must be included in the data model. Figure 3 hows excerpts of the class diagram and indicates the properties that will be processed as part of "Innovation-as-a-Service".

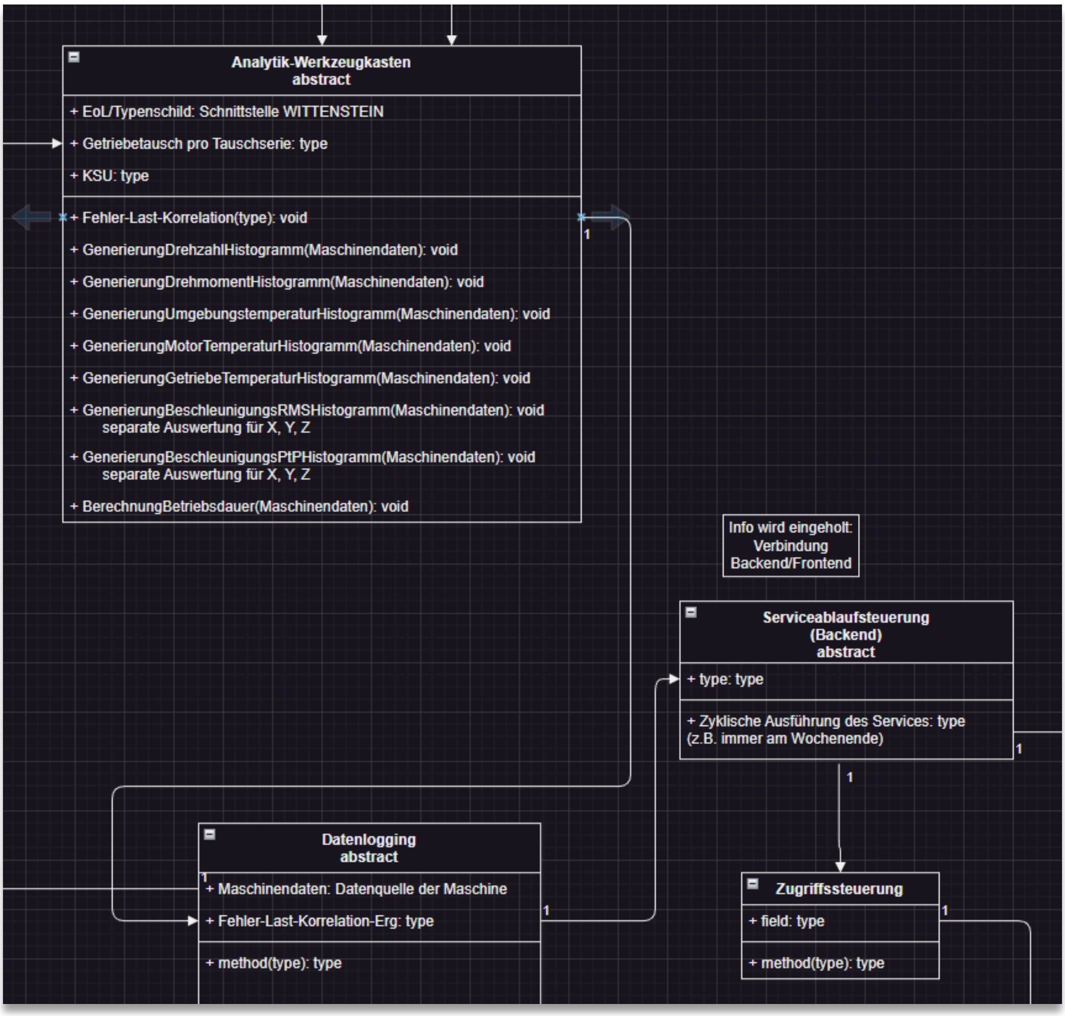


Figure 3: Excerpt from the class diagram for "Innovation as a Service"

4. Structuring the data model of the digital twin

The use cases are defined in several stages and modelled from different perspectives, creating a uniform understanding within the team of the services to be implemented in the research project. The rough idea of the services as defined at the beginning of the research project served as input for this. This is shown in steps one and two of Figure 5. The next step is the concrete derivation of the submodels of the AAS. Important aspects of the AAS are the standardization activities that define submodels. Their reuse and uniform structure simplify the digital twin's derivation and foster interoperability in the industrial context. The Industrial Digital Twin Association (IDTA) publishes these submodels. It is assessed whether there are completed submodels that have already gone through the standardization process and are suitable for application in the specific use case. In the same way, it is assessed whether there are non-standardized proprietary submodels that may already be used in the company's context. In our case, it is about identifying specific drive systems. As a result of these steps, the necessity of the standardized submodel "digital nameplate" was determined. Based on our described data and knowledge of the associated data types, we know the need for a submodel based on the submodel for time series data that is in the process of standardization. According to Figure 5, step 3 visualizes this procedure.

To successively enrich the AAS with further submodels relevant to the use case, steps four and five are devoted to the created models of the services (see section 3). On the one hand, already specified submodels can be adapted and enriched with additional properties. On the other hand, it is also intended to develop and use proprietary submodels that only meet specific requirements. The adapted or proprietary submodels benefit from the standardized interfaces of the AAS and thus ensure interoperability between AAS and other systems. Figure 4 shows an example of how the derivation of the submodels, including properties, can occur based on the previous modeling. To list exemplary structures adopted: assess which components need an

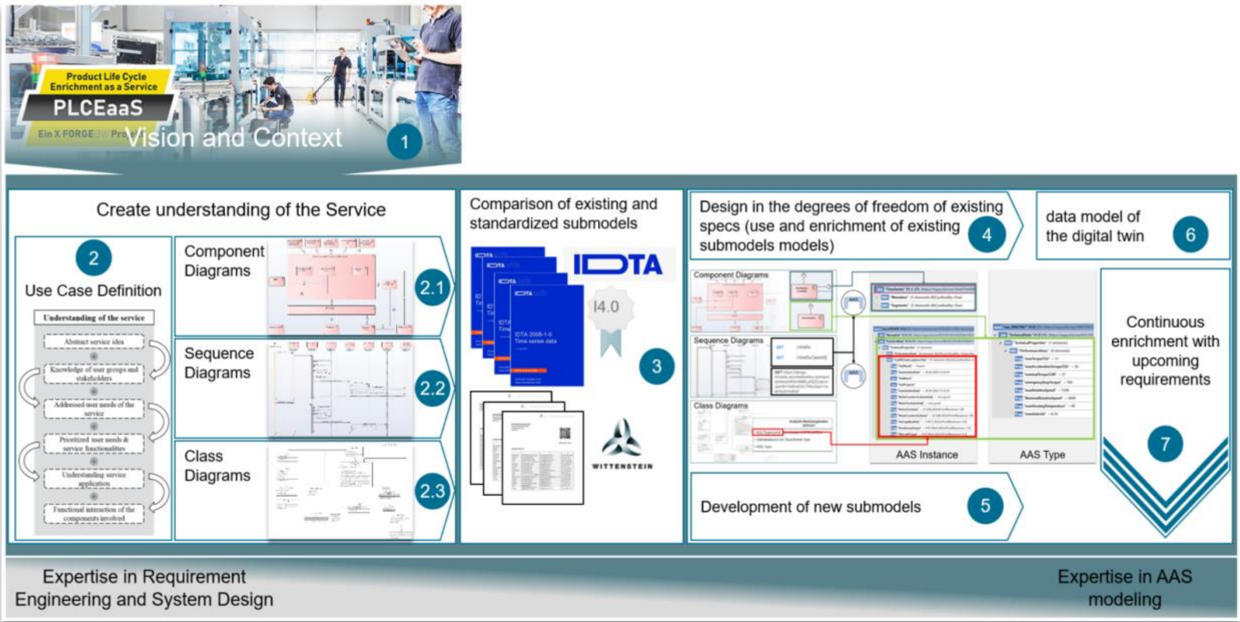


Figure 5: Process for creating a use case specific DT based on the AAS

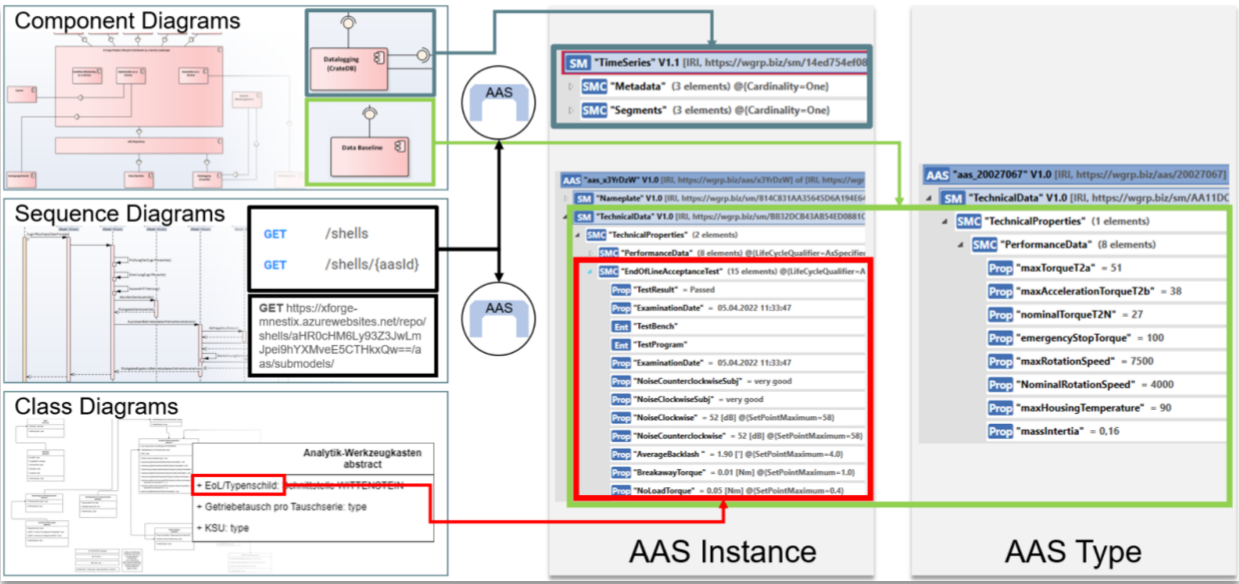


Figure 4: Derivation of specific data structure and properties from previous modelling

AAS based on the component diagram. It is not necessary to assign an AAS to each component. Rather the communication structure must be considered to see which entities communicate with each other and exchange data. For this user-specific case, the current strategy and state of the art focuses primarily on equipping the hardware (gearbox) with the AAS.

The operational data recorded and stored in a CrateDB are all in the form of time series data, so the need for the time series data submodel can be derived from their nature. The AAS represents an abstract construct that enables interoperability and a description with metadata, independent of technologies already in use and ideal technologies for the use case. In this case, CrateDB was chosen for reasons of performance, the capabilities of the project partner, and integration into their existing data platform. The service Data Baseline provides the customer catalogue and test bench data. On the one hand, specific properties for the AAS type can be derived from this. On the other hand, the properties for the respective AAS instance are also enriched with instance-specific measurement data.

The sequences of the sequence diagrams determine which data should be retrieved through the AAS or which functions should be called. It is necessary to consider this in advance so that the exact data required is loaded with individual commands without redundant data being included or data missing. For example, the get command is shown, which refers once to multiple AAS and once to an asset with a specific ID. A single flow in the diagram should usually be considered a single query providing data necessary for the interaction.

The respective properties and parameters can be taken from the class diagram. The example given shows how the exact data points relevant for the end-of-line test of a drive system and, thus, also for the data baseline service can be read from the respective class properties in the UML class diagram. The example shows the service, which processes data not only from the gearbox but also from the engine. The listing of the parameters can, therefore, not be transferred without further thinking to a model, so the integration of the parameters must be product-specific. Information and properties may well be provided several times by different submodels in the AAS to meet the different requirements.

If an asset has been provided with an AAS and its submodels are defined, this does not mean that the data model is unchangeable. Instead, new requirements imposed on an asset, for example, as a result of ongoing servitization or changing user needs, should be included. Individual submodels can still be added, removed, or adapted. Different expert knowledge is required for the respective tasks. While expertise from requirements engineering and system design is needed, particularly in the beginning, the later phases require more expertise to model the AAS. In the beginning, it is especially relevant to capture the stakeholders' needs to define necessary capabilities or systems properties and to transfer these needs, capabilities, or properties into a documented representation [16]. The latter phases, on the other hand, require much more knowledge about data, interfaces, and interaction with other systems.

5. Critical reflection

Although the interdisciplinary research team accepted our approach very well and the approach consolidated the understanding of all participants, it should be further questioned to what extent this elaborate procedure is justified. There is no comparison to an approach with prior modelling and then testing against the requirements of a use case. With the presented approach and the effort involved, the detailed conception should not hinder the implementation. Rather, the aim is to parallelize activities.

There may be several applicable modelling languages for transferring the use cases and, thus, the requirements into a documented representation. We focused on widely known and widespread tools. This selection does not imply that only these types of modelling are useful or that one should limit oneself to them. It is important that all views of the use case are covered, documented, and still practicably adaptable. It should also be noted that the current state of the use case specifications may be adjusted as the project progresses.

The submodels' basic structure and hierarchical order have not yet been worked out. Here, the content has been derived, but it makes sense and serves a standardized approach if the structure also follows a logic independent of the different domains in the production context. This general structure provides orientation and can then be filled with content using the approach shown.

Sequence diagrams can provide insight into different roles in the distribution of access rights. Currently, the security aspect is not explicitly considered, and all users have access to all data. For an industrial application, this aspect needs to be addressed.

6. Outlook

In the project, the prototypical implementation of the services is planned. For this purpose, relevant operating data must be collected on a test bed, and fault scenarios are to be simulated. The actual implementation will show to what extent the DT data model meets the requirements and provides all the data necessary for the services. Based on this, assessing whether the analytical methods chosen achieve the desired results will be necessary.

The success of AAS will undoubtedly depend on the extent to which it is adopted in industrial practice, the role of shared data spaces, and the need for operability of a wide range of assets. An idea of how AAS can be derived can help in the acceptance of the concept. This requires further elaboration to be more generally applicable and to provide detailed recommendations for action. In the context of the progressing research, it should be an object of consideration to what extent general statements can be made for deriving the data model from the models. The goal is a comprehensive chain of methods that accompanies the development of a digital twin based on the management shell, starting from a functional understanding of the service. For this purpose, steps 4 to 7 of the presented procedure, in particular, should be generalized for more services and enriched with more details at the same time. This should also include checking if any aspects are missing or any blind spots that are fundamental for the implementation.

Currently, the AAS provides a static API built on top of its data structures. A query or search interface is currently not planned and subject to future development. The mentioned queries of submodels are thus dependent on the current API specifications. This may result in further input for the standardization activities regarding the operationalization of the AAS. Thus, we currently have a reactive form of AAS with which software can interact. A proactive form of AAS, which require peer-to-peer interactions between AAS of I4.0 components and allow the components to communicate with an I4.0 language in order to realize plug&play scenarios, is not present here [19].

The scalability issue from DT [10,20] is considered because, on the basis of the AAS, we tried to store and provide data of individual assets as well as aggregated data that is fed from several assets. This consideration will also be relevant for individual services from the projects - for example, for Optimization-as-a-Service, the data of a specific product is relevant, and for Innovation-as-a-Service, the aggregated data of several products.

Acknowledgements

X-Forge is funded by State Ministry of Baden-Wuerttemberg for Economic Affairs, Labour and Tourism and implemented by VDI / VDE. The partners involved in the project are Wittenstein SE, ASCon Systems Holding GmbH, TruPhysics GmbH and Fraunhofer IPA.

References

[1] Bauernhansl, T., 2020. Die Entwicklungsstufen der Digitalen Transformation, in: Hompel, M. ten, Vogel-Heuser, B., Bauernhansl, T. (Eds.), Handbuch Industrie 4.0. Springer, Berlin, Heidelberg, pp. 1–10.

[2] Tao, F., Zhang, H., Liu, A., Nee, A.Y.C., 2019. Digital Twin in Industry: State-of-the-Art. IEEE Trans. Ind. Inf. 15 (4), 2405–2415.

[3] Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., Sui, F., 2018. Digital twin-driven product design, manufacturing and service with big data. Int J Adv Manuf Technol 94 (9-12), 3563–3576.

[4] Grieves, M., Vickers, J., 2017. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, in: Kahlen, F.-J., Flumerfelt, S., Alves, A. (Eds.), Transdisciplinary Perspectives on Complex Systems. Springer International Publishing, Cham, pp. 85–113.

[5] Grieves, M., 2015. Digital Twin: Manufacturing Excellence through Virtual Factory Replication.

[6] Bader, S., Barnstedt, E., Bedenbender, H., et al., 2022. Details of the Asset Administration Shell: Part 1 - The exchange of information between partners in the value chain of Industrie 4.0. Specification. Accessed 4 November 2022.

[7] Abdel-Aty, T.A., Negri, E., Galparoli, S., 2022. Asset Administration Shell in Manufacturing: Applications and Relationship with Digital Twin. IFAC-PapersOnLine 55 (10), 2533–2538.

[8] Industrial Digital Twin Association, 2022. How to create a submodel template specification. Industrial Digital Twin Association. https://industrialdigitaltwin.org/content-hub/downloads. Accessed 7 February 2023.

[9] Liu, M., Fang, S., Dong, H., Xu, C., 2021. Review of digital twin about concepts, technologies, and industrial applications. Journal of Manufacturing Systems 58, 346–361.

[10] Bertoni, M., 2022. Boosting Value Co-creation in Design Through the Product-Service System Digital Twin: Questions to Be Answered, in: Camarinha-Matos, L.M., Ortiz, A., Boucher, X., Osório, A.L. (Eds.), Collaborative Networks in Digitalization and Society 5.0, vol. 662. Springer International Publishing, Cham, pp. 592–602.

[11] Xia, Y., Jazdi, N., Weyrich, M., 2022. Automated generation of Asset Administration Shell: a transfer learning approach with neural language model and semantic fingerprints, in: 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA). 2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), Stuttgart, Germany. 06.09.2022 - 09.09.2022. IEEE, pp. 1–4.

[12] Machchhar, R.J., Toller, C.N.K., Bertoni, A., Bertoni, M., 2022. Data-driven value creation in Smart Product-Service System design: State-of-the-art and research directions. Computers in Industry 137, 103606.

[13] Abramovici, M., Gebus, P., Savarino, P., 2018. Engineering smarter Produkte und Services: Plattform Industrie 4.0 STUDIE. acatech – Deutsche Akademie der Technikwissenschaften. https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/hm-2018-fb-smart.pdf?__blob=publicationFile&v=3. Accessed 20 February 2023.

[14] Gausemeier, J., Meyer, M., Steglich, S., Eva-Maria, G., Rasor, R., 2022. Strategie Advanced Systems Engineering: Leitinitiative zur Zukunft des Engineeringund Innovationsstandorts Deutschland. acatech – Deutsche Akademie der Technikwissenschaften. https://www.acatech.de/publikation/die-advanced-systems-engineering-strategie/. Accessed 20 February 2023.

[15] Himmelstoss, H., Bildstein, A., Schel, D., 2021. Solution Approach For Challenges In Product Development In The Context of Emerging Cyber-Physical Systems. Hannover : publish-Ing.

[16] Pohl, K., Rupp, C., 2021. Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level, 5., überarbeitete und aktualisierte Auflage ed. dpunkt.verlag, Heidelberg, 241 pp.

[17] Bedenbender, H., Bock, J., Boss, B., Diedrich, C., Garrels, K., Gatterburg, A., Heidrich, K., Hillermeier, O., Rauscher, B., Sauer, M., Schmidt, J., Werner, T., Zimmermann, P., 2020. Verwaltungsschale in der Praxis: Wie definiere ich Teilmodelle, beispielhafte Teilmodelle und Interaktion zwischen Verwaltungsschalen (Version 1.0). https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/2020-verwaltungsschale-in-der-praxis.html. Accessed 8 September 2023.

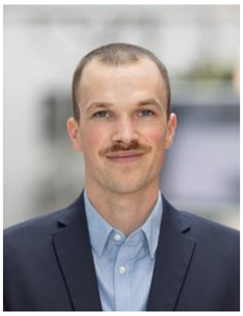
[18] Tremp, H., 2022. Agile objektorientierte Anforderungsanalyse. Springer Fachmedien Wiesbaden, Wiesbaden.

[19] Miny, T., Stephan, G., Usländer, T., Vialkowitsch, J., 2021. Functional View of the Asset Administration Shell in an Industrie 4.0 System Environment: Discussion Paper. https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/Functional-View.html. Accessed 7 September 2023.

[20] Bertoni, M., Bertoni, A., 2022. Designing solutions with the product-service systems digital twin: What is now and what is next? Computers in Industry 138, 103629.

Biography

Henry Himmelstoß (*1991) is a research associate and project leader at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart, Germany. Since 2019 he has been working in the department Competence Centre for Digital Tools in Manufacturing.



Roland Hall is a research associate at the Fraunhofer Institute for Manufacturing Engineering and Automation in Stuttgart, Germany. As part of the research group "Digital Industrial Engineering", his research topics focus on the interaction of Lean Production and Industrie 4.0 and the conception of digital tools for production.

Bernd Vojanec, expert in cooperating I4.0 systems at WITTENSTEIN SE, places particular emphasis on cross-manufacturer networking and the interoperability of I4.0 solutions, which is not limited to smart products and data-driven services but is especially relevant for business processes and the digital product twin. Furthermore, Bernd Vojanec is active in working groups of the ZVEI, VDMA and IDTA.

Paul Thieme (*1970) has been researcher and project leader at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA since 2005. Paul Thieme received his PhD in 2013 in the field of process performance measurement in administration, teaches at various universities in Stuttgart, and gives guest lectures at various European universities. Since 2017 he has been working in the department Competence Centre for Digital Tools in Manufacturing.

Prof. Dr.-Ing. Thomas Bauernhansl has been head of the Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart and the Institute for Industrial Manufacturing and Factory Management IFF at the University of Stuttgart since September 2011. Bauernhansl studied mechanical engineering at the RWTH Aachen and received his doctorate in 2002. He then worked for Freudenberg, most recently as Director of the Technology Centre at Freudenberg Sealing Technologies.

His scientific focus areas include personalized production, the biological transformation and Industrie 4.0.