Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 119 (2023) 681–686

www.elsevier.com/locate/procedia

33rd CIRP Design Conference

Development of an Information Model for Simulation Data Management in the Digital Twin

Benjamin Roehm [a]*, Reiner Anderl[a], Benjamin Schleich[a]

[a]*Product Life Cycle Management, Technical University of Darmstadt, 64287 Darmstadt, Germany*

* Corresponding author. Tel.: +49-6151-16-21843; fax: +49-6151-16-21843. *E-mail address:* roehm@plcm.tu-darmstadt.de

---

Abstract

This paper presents an information model for the association of Computer-Aided Engineering (CAE) behavior models and sensor-based simulation data under consideration of UML-association classes and artifacts of model-based testing (MBT). This scientific work extends the concept of the Digital Twin, consisting of a Digital Master and a Digital Shadow, by the entity of a Simulation Data Management in the Digital Twin (SDM-DT). The required information model of the SDM-DT represents the association between simulation data and simulation models and builds the foundation for the model-based management of simulation conditions for aggregating simulation load cases for predictive use cases. The information model is implemented for a 3D printing use case.

Keywords: Simulation Data Management in the Digital Twin; SDM-DT; Information Model; Advanced Systems Engineering; ASE

---

1. Introduction

In 2018 the term Digital Twin was classified to the "Peak of Inflated Expectations" phase of the Gartner Hype Cycle [1]. Since then, the term has gained shape and comprehensive concepts have been developed to address the expectations and requirements of the technology.

Expectations derive from promising use cases for both users and suppliers. Users see the possibilities of condition monitoring, system diagnosis, behavior prediction and behavior prescription [2]. Furthermore, suppliers see the possibility of selling their virtual behavior-describing models as part of the physical product. This paper's scope is the design of an information model that connects the two parts of existing virtual behavior-describing models with the real-world sensor architecture of the physical system to deploy those behavior models with real-world data in the product use phase.

In detail, sensor data of the physical system are processed into simulation data and used as input for the simulation of behavior models. However, only simulation data representing a critical system behavior shall be simulated, especially regarding ecological sustainability. Therefore, a conditional linkage between the open parameters of the behavior models and the sensor-based simulation data is required. This paper presents an approach for a conditional linkage based on model-based system engineering and model-based testing methods.

The paper is structured as follows. It starts with the introduction of the topics Digital Twin (DT), Advanced Systems Engineering (ASE) for the development of model-based behavior models and the concept of Model Based Testing (MBT). From this, the research gap is derived and the objective is defined. The paper's core is an information model that links the domain of the behavior-describing system model with sensor-based simulation data.

2. State of the Art

As a background for this work, the topics of Digital Twin (DT), Advanced Systems Engineering and Model-Based Testing are introduced.

10.1016/j.procir.2023.03.120

2.1. Digital Twin (DT)

The original motivation for using a DT is the virtual validation, prediction and optimization of a mechanical system throughout the entire product lifecycle without affecting the actual use phase. According to the primary definition by Grieves, a Digital Twin is a "...virtual image that contains all the information of a physical product and reflects it throughout the entire product lifecycle" [3] and can be traced back to the development efforts of the National Aeronautics and Space Administration (NASA) [4–6]. Starting with a physical spacecraft in space, they used a second spacecraft (the physical test twin) on Earth to simulate scenarios in a safe test environment before issuing mission-specific orders. With the upcoming virtual product development, the physical test twin evolved into a Digital Twin, connected to its Physical Twin through a continuous exchange of data [5].

The definition was refined in 2020 by the German Scientific Society for Product Development (WiGeP), which defines a Digital Twin as a "digital representation of a product instance" that represents "selected characteristics, states, and behaviors of the product instance or system" [7]. In addition, the entities Digital Master and Digital Shadow are defined for the first time. Thus, the paper goes on to state, "a Digital Twin is created from the Digital Master" and "a Digital Shadow […] contain data of a real product instance [...]" [7]. Accordingly, the Digital Master and Digital Shadow represent partial entities of the Digital Twin. This concept was extended in 2021 by the entity of Simulation Data Management in the Digital Twin (SDM-DT), which manages the connections between the behavior models of the Digital Master and the simulation data of the Digital Shadow [8; 9].

2.2. Advanced Systems Engineering (ASE)

The concept of the Digital Twin relies on the approaches of Advanced Systems Engineering to develop the behavior models that are transferred from the product design phase via the Digital Master into the Digital Twin. The term Advanced Systems Engineering integrates the methods of Systems Engineering (SE) and Advanced Engineering (AE) for the design of Advanced Systems (AS) [10]. Advanced Systems Engineering provides a framework integrating multiple system-oriented approaches of engineering [10]. It thus considers the methods of Systems Engineering (SE) as an "interdisciplinary approach to enable the development of complex systems, which aims to define customer needs and the required functionality as requirements early in the development process and proceed them with the system model in its entirety" [11]. It represents the advancement of document-based product development to model-based system engineering, in which the system model represents the core element within product development, from which the required views can be derived dynamically [12]. Based on the system model, the domain-specific CAE (Computer Aided Engineering) behavior models are derived and elaborated in detail within the specific application tools. Model-Based System Engineering (MBSE) enables the representation of the entire system within a formal information model according to SysML [13].

2.3. Model-Based Testing (MBT)

Model-based testing represents a particular field of ASE. With increasing system complexity and an increasing number of system requirements, it is necessary to check and document the system requirements entirely during system validation and the final product certification. Model-based testing methods support this challenge through automation employing machine-readable test models [14]. Test models describe the systematic process for validating the requirements and thus differ fundamentally from the behavior-describing system models. According to Winter et al., MBT comprises at least one of the following two aspects [14]:

- The modeling of artifacts in the test process, as well as
- the use of models for the automation of test activities.

The first point involves modeling-relevant artifacts, such as requirement diagrams, information models, and behavior diagrams that describe the testing process [13; 15]. The second point is the automated application of the developed test models. The system model, developed according to model-based system engineering, must be extended by the test and environment models. While the system model represents the behavior of the system and the test models define the test procedure, environment models describe the present and/or future boundary influences on the system, e. g. by users, forces and environmental impacts. In the case of model-based system engineering, parts of the system model can be used for the MBT.

2.4. Research Gap and Objective

The research gap relates to the model-based association of system-describing behavior models and sensor-based simulation data within the Digital Twin and the management of the associations within the Simulation Data Management in the Digital Twin (SDM-DT). In distinction to existing SDM tools, the SDM-DT "enables the management of sensor-based simulation data for a particular application in the Digital Twin" [8]. For this purpose, this research considers the FMU/FMI (Functional Mock-up Unit/Interface) standards for describing behavior models [16; 17], the methods of model-based testing [14] and the UML standard for modeling and describing the information model [18]. The objective is an information model for the SDM-DT for a model-based association of behavior models with sensor-based simulation data. The information model is one part of the design of the SDM-DT.

The design of the SDM-DT is subdivided and includes the following steps:

- Requirements definition [19],
- Process definition [20],
- Information Model development and
- Deployment.

This paper extends existing approaches [20] with the artifacts of model-based testing, contributes to the development of the Simulation Data Management System in the Digital Twin and considers the defined requirements and processes from previous scientific publications [19; 20].

3. Concept

The conceptual design of the information model is shown in Fig. 1. The graphic is divided into two areas the real world, where the Physical Twin operates, and the virtual world, where the Digital Twin is located. A bidirectional data exchange connects the two areas.

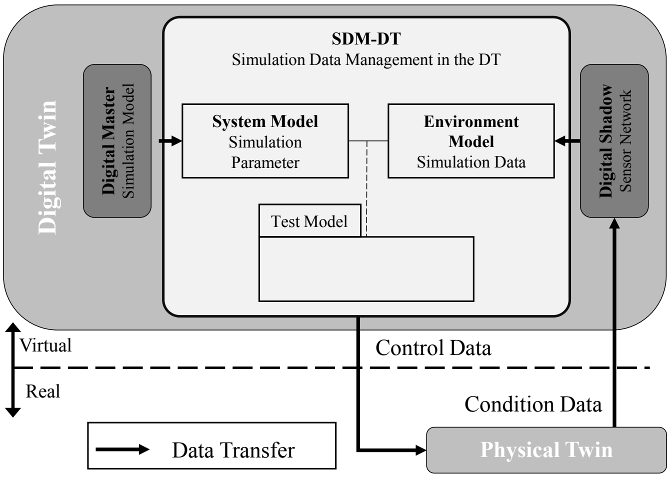

Fig. 1 Concept for the Association of System Model and Environment Model using an Association Class and the Artefacts of Model Based Testing

The Digital Twin comprises the entities of the Digital Master, the Digital Shadow and the SDM-DT. Within the SDM-DT, the association of the system models derived from the Digital Master, and the simulation data taken from the Digital Shadow, takes place. When associating the two domains, the association itself is designed by an association class according to UML 2.0. Association classes describe the relationship between two classes and define the properties of a class to the relationship [15; 21]. Aligned to the UML notation, the association class is connected to the association by a dashed line (cf. Fig. 2). The association class is part of the test model package and is further described in section 3.3.

The overall information model of the SDM-DT is shown in Fig. 3. It comprises three packages **System Model**, **Test Model** and **Environment Model**.

3.1. System Model

The behavior-describing system model is extracted from the Digital Master within the package *System Model*. It includes the class *Simulation Model* for which a *Load Case* model exists. The load cases, aggregated over the runtime, are managed within the *Simulation Library*. Each load case comprises a *Simulation Objective* and an arbitrary number of *Open Model Parameters* to be determined via the input simulation data from the environment model. In addition, each simulation model has a *Simulation Time* and *Simulation Limits*. The simulation time describes the time needed to run a simulation. The class *Simulation Limit* represents the range to apply the CAE behavior model. For example, structural mechanical models for the description of elastic strain can only be used within a specific range of values. If the input values exceed this range, other models, e.g. for plastic deformation, must be applied. The class *System Boundary* describes this value range and is available for each *Open Model Parameter*. A *Parameter Default* value is available for each open model parameter in case of missing input values.

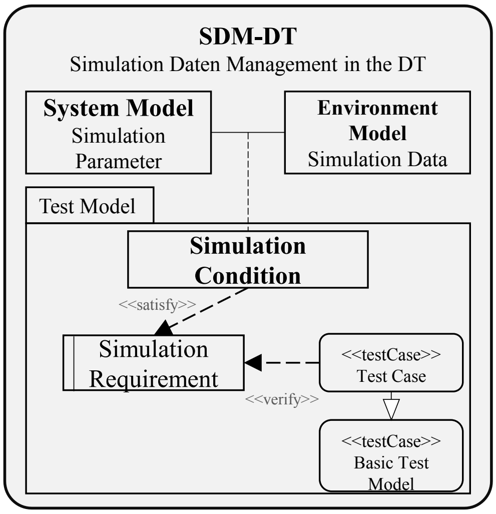

Fig. 2 Concept Information Model of the SDM-DT

3.2. Environmental Model

The Environment Model represents the sensor-based simulation data of the sensor network of the Physical Twin. The focus lies on the simulation data, which are used as input values for the open model parameters of the CAE behavior models. The class *Simulation Data* represents the sensor measurement data processed into simulation data. Sensor measurement data also includes measurement errors and measurement uncertainties. Simulation data are already cleared of these errors and can be used as simulation input. In addition to considering isolated simulation data, simulation data is also generated based on *Trend Functions*. Here, the incoming sensor measurement data is put to data analysis using AI methods to predict further data progression. On the other side, *Process Data* is represented within the package. It includes the *Process Run Time* and *Process Progress*, from which the remaining process time can be derived and compared to the simulation time. Simulations that exceed the remaining process time do not provide any benefit information for the current usage cycle and, therefore, only have a secondary benefit. The class *Simulation Data* is associated with the class *Open Model Parameter* of the System Model. The properties of this association are described within the test model in the following section.

3.3. Test Model

The association class *Simulation Conditions* includes properties of the relationship that cannot be directly assigned to either the class *Open Model Parameters* or the class *Simulation Parameters*. It consists of the simulation conditions and is associated with the artifacts of model-based testing. The artifacts are *Simulation Requirements* and *Test Cases*. The simulation requirements are formally specified requirements that must be satisfied by the association class (*Simulation Condition*) before initiating the system model with the sensor-based simulation data.

For example, a simulation requirement may specify the valid range of values of the input simulation data. If simulation data exceeds this limit, the requirements are unmet and a connection will not be established. The process for validating the simulation requirements is described with a specific test model derived from the generic *Basic Test Model*.

The *Simulation Conditions* are associated with the *Simulation Requirements* via a *satisfy*-link. Whether they meet the specified simulation requirements are continuously tested. The test procedure is described within the test models, associated with the simulation requirements via a *verify*-link. Finally, simulation worthiness can be defined via the class *Simulation Worthiness*. It depends on a methodical classification of the simulation-worthy load cases, particularly with multiple simulation models.

With the designed models, an information model for the SDM-DT is now available, which integrates the behavior models' information and the sensor network and puts them into relation via the package test model. The development of the information model is now followed by validation using an application scenario.

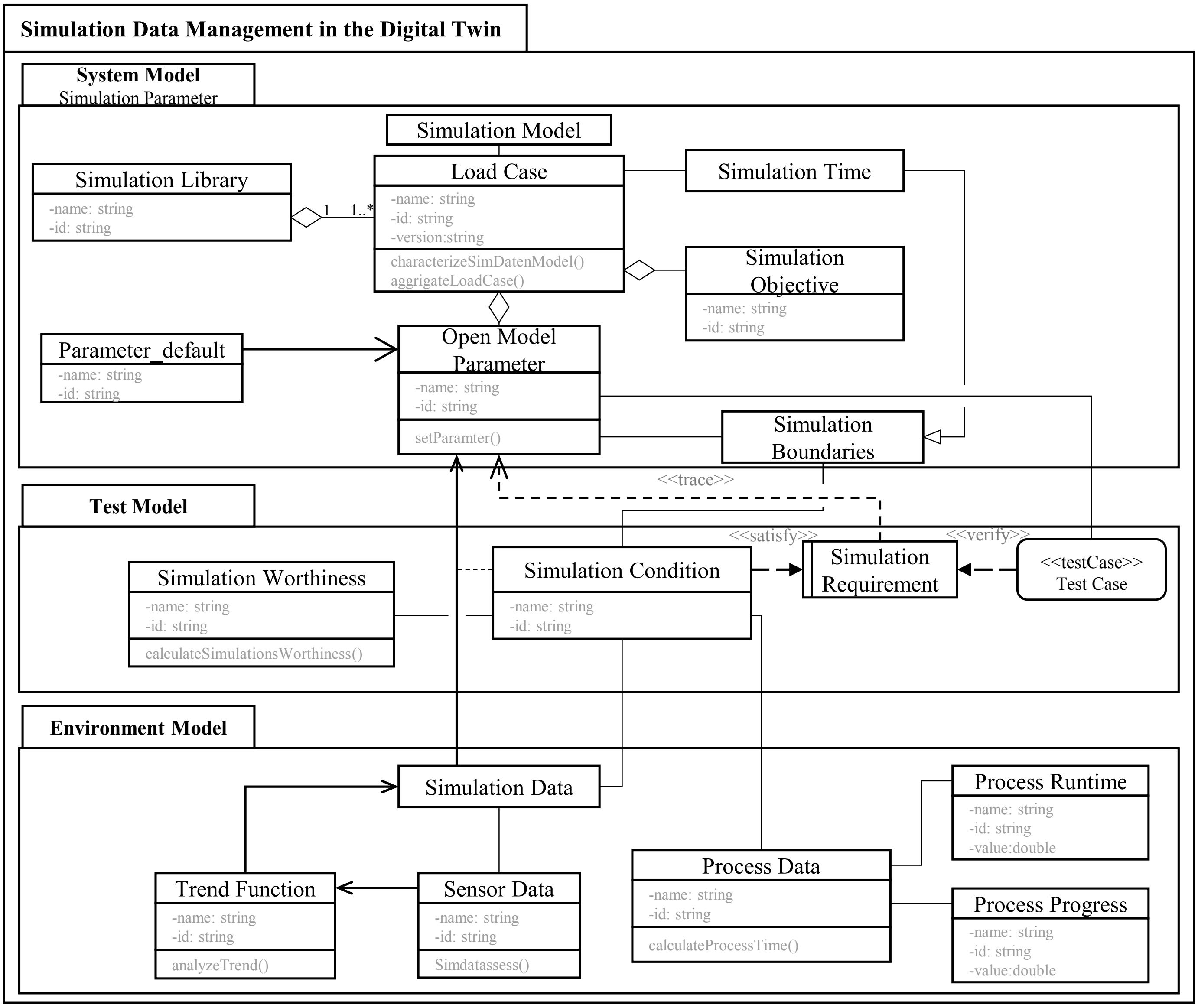

Fig. 3: Information Model SDM-DT

4. Application Scenario

The validation of the information model for the SDM-DT is based on a 3D printing application scenario. The software *Cameo Systems Modeler 2021x* was used to implement the information model and the designed artifacts.

For the implementation, a simulation model and sensor data from the product use phase were represented via the developed information model and linked to a set of defined simulation requirements. A time-discrete view of the implemented information model is shown in Fig. 4.

The simulation model is a multi-body simulation of the 3D printer, with the open model parameter *Platform Speed* and the model objective *Distance*. The use case considers the problem of printed products with a much smaller base surface than the component height detaching from the printing bed at high printing speeds, resulting in a loss of the printed product. The simulation is used during in the printing process to determine whether the printed product can withstand the prevailing speeds at full component height. The *Parameter Threshold* value for the printing bed speed is 55 mm/s. This means the printing process has been simulated and validated for a printing bed speed of up to 55 mm/s. Beyond this, there is no simulative validation. Two simulation requirements (ID 1.1.2 and 1.1.3) are given for the model parameter *Platform Speed*:

- For a simulation, the *Simulation Data* must be higher than the *Parameter Threshold* and
- The *Simulation Data* must be smaller than the upper *Parameter Simulation Threshold*.

The input simulation data is continuously assessed according to the simulation requirements in a model-based approach. The view of the information model in Fig. 4 represents simulation data with a value of 58 mm/s. The simulation data is linked to the platform speed parameter, and the link has the simulation condition designed as an association class. The association class is linked to the simulation requirements via the *satisfy* connection and those are linked to a test case via a *verify*-link. The simulation condition association class contains two operations to calculate the difference between the simulation data and the parameter threshold and the difference between the simulation data and the parameter simulation threshold (*subtract( )*). The results are represented as the class's attributes (result1 and result2).

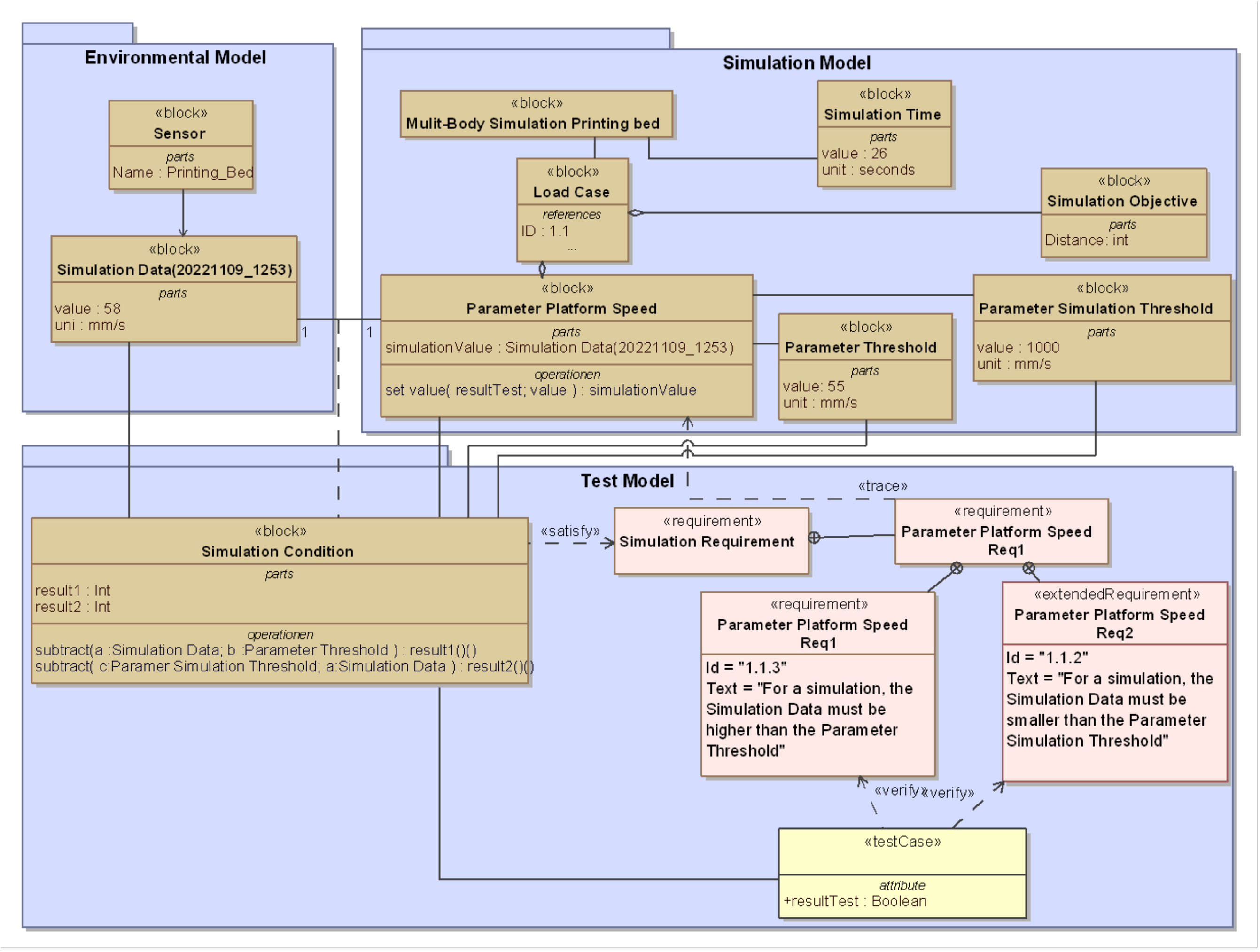

Fig. 4: Implementation of the SDM-DT Information Model with Cameo Systems Modeler

The designed data model provides the foundation for model-based verification of *Simulation Requirements* and shows the model-based tested simulation requirements.

At present, both simulation requirements are satisfied (pass), the simulation data is higher than the simulation threshold and is within the simulation limit. If the simulation requirements are positive, the *testCase* passes the boolean *true*. This boolean is processed in the class *Parameter Platform Speed*, and the parameter value is set equal to the simulation data. Table 1 presents an exemplary view of the current requirements.

Table 1 Requirement Test Result 20221107_1253

	Simulation Requirement	
Model Parameter	Req1.1.2	Req1.1.3
Paltform Speed	pass	pass

The present use case shows the functionality of the developed information model for the model-based association of behavior models with sensor-based simulation data using association classes and artifacts of MBT. The strength of the chosen approach can be expected when scaling the Digital Twin to a large number of simulation models and an increasing number of open model parameters, simulation conditions and simulation requirements.

5. Conclusion and outlook

This paper shows the capacity of the model-based linkage of simulation models and simulation data within the SDM-DT for enabling the use cases of the Digital Twin and the interoperability between the real and virtual worlds. It represents a model-based, central management of simulation requirements. The approach distinguishes from decentralized decision logic, which is part of each behavioral model. Considering association classes, simulation requirements and test models, the simulation need can be validated automatically and presented to the users. The UML element, association class, combines the model-based testing's artifacts for model-based linking. The future work is the design of generic test cases for the automated instantiation of test scenarios. Additionally, the approach must be implemented and validated for use cases with multiple behavior models, simulation parameters and requirements.

Acknowledgements

This research and development project CyberTech is funded by the German Federal Ministry of Education and Research (BMBF) within the "Innovations for Tomorrow's Production, Services, and Work" Program (02J19B012) and implemented by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the content of this publication.

References

[1] Sony Shetty. Gartner 2018 Hype Cycle for IT in GCC. Available from: https://www.gartner.com/en/newsroom/press-releases/2018-12-13-gartner-2018-hype-cycle-for-it-in-gcc-identifies-six-technologies-that-will-reach-mainstream-adoption-in-five-to-10-years. 2018. Accessed on 11 November 2018.

[2] Schuh G, Walendzik P, Luckert M, Birkmeier M, Weber A, Blum M. Keine Industrie 4.0 ohne den Digitalen Schatten. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb; 2016; ;11:745–748.

[3] Grieves M, Vickers J. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In: Kahlen F-J, Flumerfelt S, Alves A, editors. Transdisciplinary Perspectives on Complex Systems. Cham: Springer International Publishing; 2017. p. 85–113.

[4] Grieves M. Virtually perfect -Driving innovative and lean products through product lifecycle management. Cocoa Beach Florida: Space Coast Press; 2011.

[5] Grieves M. Digital Twin White Paper: Manufacturing Excellence through Virtual Factory Replication; 2014.

[6] Ohnemus T. The Digital Twin – A Critical Enabler of Industry 4.0. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb; 2020; 115;special:23–25.

[7] Stark R, Anderl R, Thoben K-D, Wartzack S. WiGeP-Positionspapier: „Digitaler Zwilling“. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb; 2020; 115;special:47–50.

[8] Roehm B, Anderl R. Concept of a System Architecture for a Simulation Data Management In the Digital Twin. ASME 2021 30th Conference on Information Storage and Processing Systems: American Society of Mechanical Engineers; 2021.

[9] Kutscher V, Olbort J, Roehm B, Plesker C, Anderl R. Web-Based Digital Twin. Journal of Innovation - Innovations in Digital Twins; 2021; ;16:1–18.

[10] Dumitrescu R, Albers A, Riedel O, Stark R, Gausemeier J. Engineering in Deutschland – Status quo in Wirtschaft und Wissenschaft, Ein Beitrag zum Advanced Systems Engineering. Paderborn; 2021.

[11] Eigner M. 3. Engineering 4.0—Foundations of the Digitalization of Engineering. In: Eigner M, editor. System Lifecycle Management. Wiesbaden: Springer Fachmedien Wiesbaden; 2021. p. 27–100.

[12] Zimmermann TC, Masuhr C, Stark R. MBSE-Entwicklungsfähigkeit für Digitale Zwillinge. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb; 2020; 115;special:51–54.

[13] Alt O. Modellbasierte Systementwicklung mit SysML. München: Carl Hanser Fachbuchverlag; 2012.

[14] Winter M, Roßner T, Brandes C, Goetz H. Basiswissen modellbasierter Test. 2 ed. Heidelberg: dpunkt.verlag; 2016.

[15] Rupp C, Queins S. UML2 glasklar. 4 ed. München: Hanser; 2012.

[16] Pohlmann U, Wagner R. Einsatz des FMI/FMU-Standards zur frühzeitigen Simulation von Software- und Hardwaremodellen komplexer mechatronischer Systeme. In: Maurer M, Schulze S-O, editors. Tag des Systems Engineering. München: Carl Hanser Verlag GmbH & Co. KG; 2013. p. 229–238.

[17] Modelica Association. Functional Mock-up Interface Specification.

[18] OMG Standards Development Organization. Unified Modeling Language; 2017.

[19] Roehm B, Emich B, Anderl R. Approach of simulation data management for the application of the digital simulation twin. Procedia CIRP Design; 2021; 100:421–426.

[20] Roehm B, Anderl R. Simulation Data Management in the Digital Twin (SDM-DT) – Evolution of Simulation Data Management along the Product Life Cycle. Procedia CIRP; 2022; 105:847–850.

[21] Kecher C. UML 2. 4 ed. Bonn: Galileo Press; 2014.