Report from Dagstuhl Seminar 22362

Model-Driven Engineering of Digital Twins

Loek Cleophas[1], Thomas Godfrey[2], Djamel Eddine Khelladi[3], Daniel Lehner[4], Benoit Combemale*[5], Bernhard Rumpe*[6], and Steffen Zschaler*[7]

1 **Eindhoven University of Technology, NL & Stellenbosch University, SA.** l.g.w.a.cleophas@tue.nl
2 **King's College London, GB.** thomas.1.godfrey@kcl.ac.uk
3 **CNRS, IRISA, Univ. Rennes, FR.** djamel-eddine.khelladi@irisa.fr
4 **Johannes Kepler Universität, Linz, AT.** daniel.lehner@jku.at
5 **University & IRISA – Rennes, FR.** benoit.combemale@irisa.fr
6 **RWTH Aachen, DE.** rumpe@se-rwth.de
7 **King's College London, GB.** steffen.zschaler@kcl.ac.uk

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 22362 "Model-Driven Engineering of Digital Twins".

Digital twins are an emerging concept with the potential for revolutionising the way we interact with the physical world. Digital twins can be used for improved analysis and understanding of complex systems as well as for control and transformation of these systems. Digital twins are themselves complex software systems, posing novel software-engineering challenges, which have so far not been sufficiently addressed by the software-engineering research community.

The seminar aimed as a key outcome to contribute to a solid research roadmap for the new Software Engineering subdiscipline of Model-Based Development of Digital Twins. This paper is an intermediate result, which is thought to be further discussed in the research community that has also been built using this seminar.

Seminar September 4–9, 2022 – http://www.dagstuhl.de/22362
2012 ACM Subject Classification Software and its engineering → Software notations and tools
Keywords and phrases Software Engineering, Model-Driven Engineering, Digital Twins, Model Management, Data Management, Models@runtime
Digital Object Identifier 10.4230/DagRep.12.9.20

* Editor / Organizer

Model-Driven Engineering of Digital Twins, *Dagstuhl Reports*, Vol. 12, Issue 9, pp. 20–40
Editors: L. Cleophas, T. Godfrey, D. Khelladi, D. Lehner, B. Combemale, B. Rumpe, and S. Zschaler
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Summary

Loek Cleophas
Thomas Godfrey
Djamel Eddine Khelladi
Daniel Lehner
Benoit Combemale
Bernhard Rumpe
Steffen Zschaler

Digital twins are an emerging concept with the potential for revolutionising the way we interact with the physical world. Early versions of digital twins have already been applied successfully in almost all known areas, including engineering and development areas, but also scientific domains and cultural, social, and economic domains.

Digital twins are leveraging the digitalization of increasingly more application domains and are intensively using various models of different types in a descriptive, predictive, and prescriptive way. Furthermore, the the ever-increasing availability of data, the last improvements of sensor technologies, and the reliable connnectivity enable direct inspection and manipulation of real-world systems, both for physical systems and objects as well as social systems and organisations respectively their processes.

The concept has seen strong interest in industry, where there is a desire to control increasingly complex systems of systems, ensuring they behave as expected and to control their adaptation to the environment or any deviations with the initial plan. Digital twins can be used for improved analysis and understanding of complex systems (in silico experimentation) as well as for control and transformation of these systems. Digital twins are themselves complex software systems, posing novel software-engineering challenges, which have so far not been sufficiently addressed by the software-engineering research community.

There is a need for solid foundations to ensure the development of tools and methods according to well-established principles. We believe that Model-Driven Engineering (MDE), will be a key technology for the successful systematic engineering of Digital Twins. In this Dagstuhl Seminar, the goal was to bring together both practitioners and researchers to

- (i) reflect on the concept of Digital Twins and the software-engineering challenges posed,
- (ii) identify relevant existing MDE approaches and technologies that can help tackle the challenge of systematically engineering digital twins, and
- (iii) define an academia–industry research roadmap for systematic engineering of digital twins based on MDE.

As the intended primary goal of the seminar is to create a community and establish a research roadmap, we have been discussing the following topics:

- Challenges faced in real-world development of Digital Twins.
- Opportunities offered by MDE.
- Active exploration of collaboration opportunities.

The following paper reflects the discussions and some of the outcomes, however, we also identified that the overall topic is not only relevant, but also highly innovative, which is why this paper does only reflect an intermediate status of discussions and results, but the community will vividly go on to solve the challenges identified and addressed in the rest of this paper.

One key outcome of the seminar and its continuing community activities will be to contribute to a solid research roadmap for the new Software Engineering sub-discipline of Model-Based Development of Digital Twins.

Definitions to set the stage

There are two core terms that need appropriate definitions, namely model driven engineering (MDE) and digital twin. While MDE (and with it the terms model and modeling language) are relatively straightforward, there are different variants of definitions for digital twins that served as a starting base.

▶ **Definition 1** (Model Driven Engineering (MDE)). *Model Driven Engineering* is a state-of-art software engineering approach for supporting the increasingly complex construction and maintenance of large-scale systems [4, 3, 2]. In particular, MDE allows domain experts, architects, developers to build languages and their tools that play an important role in all phases of the development process [7].

As digital twins are currently a relatively young and in particular evolving area of research, it is not surprising, that there is a variety of different definitions available. (E.g. [1] identifies more than hundred different definitions).

At the beginning and during the seminar we identified the following definitions to be particularly of interest.

▶ **Definition 2** (Digital Twin (DT)). A *digital twin* (DT) is a comprehensive digital representation of an actual system, service or product (the Physical Twin, PT), synchronized at a specified frequency and fidelity [5]. The digital twin includes the properties, conditions and behavior of the physical entity through models and data, and is continuously updated with real-time system data [6]. The exchange of data between the digital and the physical twins takes place through bidirectional data connections.

An alternative definition can be found at the website of the software engineering institute of RWTH Aachen University[1] together with some additional discussions:

▶ **Definition 3** (Digital Twin (DT)). A *digital twin* of a system consists of

- a set of models of the system and
- a set of digital shadows, both of which are purposefully updated on a regular basis,
- provides a set of services to use both purposefully with respect to the original system, and
- can send
  - information about the environment and
  - control commands to the original system.

▶ **Definition 4** (Digital Twin System (DTS)). Based on these definitions, a *Digital Twin System* can be defined as the combination of an actual system and the DTs of this actual system.

[1] `https://www.se-rwth.de/essay/Digital-Twin-Definition/`

2 Table of Contents

Timeslot	Monday	Tuesday	Rest of the week
09:00-10:00	Opening and Seminar Objectives Participant introductions	Presentations: Futuristic scenarios for DT use 10+5min each	Plenary session
10:00-10:30	Coffee Break		
10:30-12:15	Presentation and discussion: Digital Twin Terminology	Presentations: Futuristic scenarios for DT use 10+5min each	Groups discussion
12:15-13:30	Lunch		
13:30-15:30	Presentations: Existing DTs in different contexts 10+5min each	Discussion: "Requirements elicitation": what do we need to build to create a full DT?	Groups discussion
15:30-16:00	Coffee Break		
16:00-17:30	Discussion: Properties of existing DTs, commonalities, differences, challenges	Discussion: Planning for rest of the week (schedule and breaking down)	Groups discussion
18:00-19:15	Dinner		
19:30-20:30	Discussion: Planning for rest of the week (main objectives)	*Organizer's meeting*	Presentations from participants and debrief
Evenings	Time for demonstrations, discussions, pool, …		

Figure 1 Schedule of the seminar.

3 Schedule of the seminar and presentations

This section presents the schedule of the dagstuhl week and the various presentations we had during the seminar.

3.1 Schedule of the week at Dagstuhl

Figure 1 depicts the schedule of the seminar week. It started by opening the seminar week and then with presentations and discussions about the terminology, context, and properties of digital twins. We then discussed the planning of the rest of the week. In the second day, we had presentations of real world scenarios of digital twins. After that, we discussed about requirements for digital twins and finalized the planning of the rest of the week. Finally, for the rest of the week, we split in three groups to work in parallel, with a plenary session in the morning and the evening to debrief each other about what was done and will be done.

4 Overview of Talks

This section presents the various presentations we had during the seminar. It details the author and the talk's title and abstract.

4.1 Learning Digital Twin Models

Shaukat Ali (Simula Research Laboratory – Oslo, NO)

Given that operational cyber-physical systems (CPS) produce continuous data, a complementary approach to model-based engineering is to learn digital twins models with machine learning techniques and providing functionalities such as predictions and anomaly detection.

This talk will start with presenting an opinion on the next generation of digital twins (Quantum Digital Twins), where some aspects of digital twins will be implemented as quantum software and executed on quantum computers, e.g., for simulating the physical environment that can be realistically simulated with quantum-mechanical principles.

Followed by this opinion, the talk will present some recent works on learning digital twins from historical data and continuous updates of digital twins with continuous data from operational CPS. Various machine learning techniques were applied, such as generative adversarial networks, curriculum learning, and transfer learning to learn digital twins. The digital twins were built for use cases from the transportation domain and water distribution/treatment plants. These digital twins were focused on anomaly detection and waiting time predictions.

4.2 Theory building and sociotechnical digital twin: MDE Requirements

Balbir Barn (Middlesex University – London, GB)

Depending upon the flavour of digital twin (and that is an entirely different problem), a digital twin of a real world artefact is designed for a purpose. Such a purpose can can range from trying to understand an "as is" situation, analysing multiple options in a decision making scenario through to tractable transformation of the real world artefact. Underpinning all these purposes is a form of theory building and assumptions of underlying theories.

A theory in its most sparse understanding is a statement of what causes what, and why, and under what circumstances. A theory can be a contingent statement or a proven statement. We use theories all the time. Peter Naur referred to programming as form of theory construction. Decision makers in organisations use theory every day. They make decisions on some basis of cause and effect, often without being specific about their reasoning. Naturally, theories are empirical or can remain conceptual explanations.

The most widely accepted notion of a Digital Twin (DT) is one where there is a cyber-physical component. DTs of civil structures such as bridges are one such example. Such DTs rely on well understood physics based laws which provide empirically tested theories. New generations of DT belong to a socio-technical context where there is heady mix of human action, systems integration and emergent behaviour that cannot readily be assumed to follow a particular predicted route. Such Socio-technical DTs when they are constructed need input from social science, psychology and other inter-disciplinary fields. Theories for justifying concepts, relationships and rules of such a STDT are therefore obtained from these non-IT fields. How do we embed and use these theories?

The challenge for model driven engineering of STDT is therefore further attenuated and more engineering support is needed. For example, we need a modelling language that is sufficiently rich to capture working descriptions of key social science theories. We will need libraries of these theories together with working examples. We will need theory integration environments and accompanying methods that support theory building. We want to be able to run a model and view the execution of the theory and its explanation.

The benefits of being able to demonstrably prove that an established theory underpins a component of a DT specification model provides external validity through a reference bench mark of a well understood theory that has been critiqued at scale and over time. Model validity concerns are therefore mitigated. Essentially this is a research challenge and I am not yet aware of significant progress in this area with respect to Digital Twins.

4.3 Digital twins Automotive

Ion Barosan(TU Eindhoven, NL)

Advancements in the automotive industry is being highly dependent on Software technology as the industry is now changing from manual driving to different levels of autonomous driving. In order to develop autonomous driving vehicles, digital twins are now used to achieve the desired features and functionality. The approach to use digital twins saves huge resources as it accelerates product development and perform virtual simulation forecasts before production. Another use case is using the twin in tandem with the physical environment providing functionalities of monitoring, fault detection to name a few. The Automotive Technology research lab at Eindhoven University of Technology, Netherlands is performing research and development of such systems. As a part of this lab, this project focuses on design and implementation of a digital twin software framework for autonomous articulated vehicles within a distribution center. The goal is to develop such a system, that replicates 'the real world moving of autonomous trucks in a distribution center' within a Digital Twin. Additionally, the virtual system should be able to gather data from various sensors in order to enhance the development of the physical truck. This data can then be used by users to further study or visualize. In order to develop this complex system, a software system representing the digital twin called the TruckLab-DTF has been developed. Using the TruckLab-DTF product development teams working on achieving autonomy in distribution centers can apply insights from the virtual twin and physical system directly to their development. Additionally, requirements can be verified in the digital twin early in the design phase, saving time and money. Thus, the use of digital twin with this domain is multi and this project is an initial step towards the realization of much larger "system of systems." In extension, designed software can be also used as an educational technology tool for vehicle dynamics and control-based courses.

4.4 Digital twins = models@run.time + Simulations

Nelly Bencomo (Durham University, GB)

Digital twins (DT) have emerged as a promising paradigm for run-time modelling and performability prediction of cyber-physical systems (CPS) that can be applied in multiple domains. Different definitions and industrial applications of DT have materialised, going from purely visual three-dimensional models to predictive tools, many of them focusing on data-driven evaluations. We want to focus on a conceptual framework based on autonomic systems to host DT run-time models based on a structured and systematic approach. A model at run-time can be defined as an abstract representation of a system, including its structure and behaviour, which exist alongside the running system. Run-time models provide support for decision-making and reasoning based on design-time knowledge. However, they can also offer themselves as a run-time abstraction based on information that may emerge at run-time and was not foreseen before execution. New techniques based on machine learning (ML) and Bayesian inference offer great potential to support the update of run-time models during execution. Run-time models can be updated using these new techniques to provide better-informed decision-making based on evidence collected at run-time. The syncing of the real and the digital twin: Models@runt.time and the MAPE-K loop can provide the structured basis of the software architecture presented and how the required casual connection of run-time models is realised to sync the real and the digital twins. This process keeps the running system inside an envelope of good behaviour.

4.5 Digital Twin for DevOps Process Improvement

Francis Bordeleau (ETS – Montreal, CA)

DevOps emerged in the last decade as the prominent approach to increase productivity and system quality in the software industry. It advocates for automation and monitoring at all stages of software development and operations, and aims for shorter development cycles, increased frequency of deployment, and more reliable releases. Its adoption by industry leaders (e.g. Amazon, Facebook, Google, and Netflix) has resulted in spectacular progress. However, evolving/improving the software process remains a main challenge and many companies are struggling with the implementation and evolution of software processes. The lack of a systematic approach makes continuous improvement an ad hoc journey in which decisions are based on intuition rather than facts. To enable the systematic improvement of DevOps processes, we propose a digital twin approach that addresses two main challenges: 1) the continuous monitoring and measurement of DevOps process according to specific objectives to detect issues so that improvement actions can be taken; and 2) the evaluation of various modification alternatives to reach a specified DevOps process improvement objective; this allows to take decisions on a scientific basis rather than in an ad hoc manner.

4.6 Model Driven Adaptable Digital Twins

Tony Clark (Aston University – Birmingham, UK)

A series of architecture models for digital twins of increasing sophistication is presented, leading to a digital twin that adapts in order to control a real-world asset. A technology is in development that supports the construction of adaptable digital twins. An overview of a simple use of the technology is presented.

4.7 Model-Driven Engineering for Enterprise Digital Twins: Opportunities and Challenges

Benoit Combemale (IRISA, University Rennes, FR)

Enterprises are rapidly evolving as complex socio-technical systems that require continuous adaptation regarding a dynamic and uncertain environment. Various stakeholders are involved and deliver continuous knowledge through heterogeneous digital models (systems, processes, organizations, etc.) manipulated with various scientific and engineering environments. This modeling continuum open new opportunities for adaptable and efficient enterprises, but also raise new modeling challenges. In this talk, I explore the use of model-driven engineering to develop and operate enterprise digital twins. The talk covers the current state of the art, and provide concrete implementations for smart trade-off analysis and decision making. Finally, I discuss open challenges and draw a roadmap for the community.

4.8 Conceptual Modelling for Risk Modelling

Georg Grossmann (University of South Australia – Mawson Lakes, AU)

The modelling of risks and predicting the impact of risks is crucial in physical asset management. Digital Twins can support the risk modelling and prediction by providing a holistic view of the assets but there is no existing standard on how to model risks. The challenges of modeling risks comprise the support of different abstraction levels, integration of different views of risks and supporting the evolution of risk models over time through Digital Twins. We provide an overview of those challenges with examples from energy industry.

4.9 Digital Twins for Cyber-Physical Systems

Gabor Karsai (Vanderbilt University – Nashville, US)

The talk will give a very brief summary of existing Digital Twin efforts in the US, and then introduces some ideas about the specific challenges digital twins have to solve in the domain of Cyber-Physical Systems. A specific approach for addressing the model integration problem is discussed, as well as how DT-s can assist in autonomous system operations. Finally, five fundamental challenges are posed.

4.10 The MB.OS Approach: How can Complexity be Managed in a Software-Defined World?

Oliver Kopp (Mercedes-Benz AG – Stuttgart, DE)

Product complexity is ubiquitous – and it is constantly increasing. Instead of working against it, we should accept it as challenge and make complexity manageable.

Why are software methods becoming increasingly important in this context? Because that is exactly where they come in: Complexity becomes controllable by software methods. For the automotive business, this means that we must use systems engineering methods to manage product variance holistically, based on a central data model having an end-to-end scope over lifetime. To do this, we follow a data-centric approach that maps every aspect: Both in the problem space (requirements, features, regulatory specifications) including the product configurations and in the solution space (types, options, functions, components) including the corresponding type configurations.

In this talk an overview of our concept “Typebased Product Line Engineering” is presented: How can we move from a reactive, quantity-based product documentation approach to a proactive product documentation approach based on individual and concrete configurations? The role of a Digital Twin is outlined – especially in the context of a vehicle that can always be upgraded with new features via over-the-air updates throughout its lifetime in customer hands.

Acknowledgement: The talk was originally presented by Christian Seiler, Mercedes-Benz AG at the Digital Product Forum 2022. This work is partially based on the research project SofDCar (19S21002), which is funded by the German Federal Ministry for Economic Affairs and Climate Action.

4.11 OASIS TOSCA: A meta and meta-meta model for modeling and managing of structured applications

Oliver Kopp (Mercedes-Benz AG – Stuttgart, DE)

Deployment denotes the installation and the running of necessary components of an application. There are mulitple tools available to do so: Terraform nad Ansible form two examples. They all have different meta-models to capture the desired application topology (also called "infrastrucutre-as-code"). OASIS TOSCA ("Topology and Orchestration Specification for Cloud Applications") is a standard including both a meta-model and a meta-meta model to model the application topology in a standardized and vendor-neutral way. This talk outlines aspects of TOSCA and presents Eclipse Winery as one tool to model applicaton topologies using TOSCA.

Acknowledgement: This work is partially based on the research project SofDCar (19S21002), which is funded by the German Federal Ministry for Economic Affairs and Climate Action.

4.12 JabRef as a Literature Management Tool and a Software Engineering Training Tool

Oliver Kopp (Mercedes-Benz AG – Stuttgart, DE)

JabRef is a literature management software based on BibTeX written in Java. The project management is mainly done using GitHub's features. The talk will fist outlined JabRef's functionalities and then dives into open source software development. We showed how the JabRef team supports newcomers to find new issues and to craft a code contribution – and how this helps to use JabRef as teaching object for training Software Engineering on intermediate level.

4.13 Towards Dynamic Self-adaptive DT Architectures

Daniel Lehner (Johannes Kepler Universität Linz, AT)

With the emergence of Digital Twins, more and more architectures are developed to use these Digital Twins in particular context to serve a specific purpose. However, the components of such architectures are usually targeted to its particular context and purpose, with limited adaptability. This makes it hard to change an architecture once the purpose of the underlying system changes, or reuse existing components in developing new architectures. To solve this challenge, we propose a feature model of Digital Twins that enables to put together new software architectures that leverage these Digital Twins. The component-based realization of the individual features on the software and modeling language level should enable an efficient plug and use of Digital Twin architectures in the future.

4.14 Engineering/Working with Digital Twins

Bernhard Mitschang (Universität Stuttgart, DE)

From a data-centric point of view digital twins are just data structures/models that keep relevant information of real-world artifacts (machines, products, environments etc.). Thus "working with digital twins (cf.my title)" simply means adapting/changing/extending/merging/intersecting/snipping/... these structures/models. My main questions raised – and perhaps even answered during this seminar – are:

- Is this really so simple?
- What does it mean to merge two digital twin models and what is the impact and consequences thereof?
- Don't we need to consider (domain) semantics?

4.15 On the Conceptual Modeling of Behavior: Dynamic Reclassification of Entities

Alfonso Pierantonio (University of L'Aquila, IT)

The concept of classification as realized in most traditional object-oriented computer languages has certain limitations that may inhibit its application to modeling more complex phenomena. This is likely to prove problematic as modern software becomes increasingly more integrated with the highly dynamic physical world. In this paper, we first provide a detailed description of these limitations, followed by an outline of a novel approach to classification designed to overcome them. The proposed approach replaces the static multiple-inheritance hierarchy approach found in many object-oriented languages with multiple dynamic class hierarchies each based on different classification criteria. Furthermore, to better deal with ambiguous classification schemes, it supports potentially overlapping class membership within any given scheme. Also included is a brief overview of how this approach could be realized in the design of advanced computer languages.

4.16 Simulation to digital twin

Fiona A. C. Polack (University of Hull, GB)

We have already explored engineering a simulation so that it can be trusted (in the non-strict sense), and so that the basis for the trust can be interrogated and challenged: the principled simulation approach has been used in e.g. immune system simulation and robotics. In 2000s we proposed MDE as one way to automate development so that effort could focus on abstraction, design and interpretation of results, plus use of the simulation for its purpose. Digital Twin development might perhaps learn from engineering principled simulation, particularly in areas such as purpose, results interpretation/use, fitness/validation, management of trustworthiness. Again MDE could allow "people" to focus on these non-automatable but critical areas.

4.17 Digital Twins – Challenges from a Modelling Perspective

Matthias Riebisch (Universität Hamburg, DE)

This talk describes challenges for the development of the digital twin from a modelling perspective.

4.18 Federation of Digital Twins to conform with Manufacturing Systems

Matthias Riebisch (Universität Hamburg, DE)

Manufacturing systems are structured as multi-layered systems, with different goals and constraints for the layers. Digital Twins must reflect this by being split into parts what we call Federated Digital Twin. For its parts, and for the bridges between them, appropriate decisions on coverage, structure, technology, etc. are required.

4.19 Digital Ecosystems and Digital Twins

Markus Stumptner (University of South Australia – Mawson Lakes, AU)

Digital Twins need to accommodate the interactions of the System under Study (SuS) to be able to collect data, stimuli and outputs. This requires interaction with the (software) ecosystem surrounding the SuS, determined by relevant functions, and defining a landscape of models and their interactions.

4.20 The importance of being Uncertain

Antonio Vallecillo (University of Málaga, ES)

A fundamental characteristic of software models is their ability to represent the relevant characteristics of the system under study, at the appropriate level of abstraction. We now live in the age of cyber-physical systems, smart applications and the Internet of things, which require some forms of interaction with the physical world. Uncertainty is an inherent property of any system that operates in a real environment or that interacts with physical elements or with humans but, unfortunately, the explicit representation, management and analysis of uncertainty has not received much attention by the software modeling community. In this talk we analyze the impact of measurement uncertainty on the behavior of the system using examples, and describe the traditional ways of dealing with it, namely using fixed and confidence (adaptive) intervals. We then discuss the need to consider all attributes as random variables and the importance of properly comparing them.

4.21 Digital Z (model/shadow/twin/passport/ ...) "twinning" for and by Systems Engineering

Hans Vangheluwe (University of Antwerp, BE)

Throughout their life-cycle (design, production, assembly, operation and optimization, maintenance, re-purposing, disposal), smart, adaptive systems will include ecosystems of Digital Zs (models/shadows/twins) to achieve a plethora of goals such as condition monitoring, fault diagnosis, predictive maintenance, optimization, etc. . This builds on many existing techniques, architectures and standards from real-time simulation, co-simulation, systems and control theory, IoT, knowledge management, machine learning, experiment/validity frames, surrogate modelling, etc. To satisfy system goals, a federated knowledge repository (graph) (a "Modelverse") containing both "linguistic" and "ontological" information, is used as a starting point for inferencing. This leads to the (product line: goals to realizations) construction of new Digital Z "experiments" which, when deployed, in turn, yield new data/knowledge which is merged (during or at the end of an experiment) with the information already present in the Modelverse. As multiple concurrent inferencing and experiment processes may exist, the Modelverse acts as a blackboard. Conceptually, one Digital Z "experiment" is created per Property of Interest (to be monitored, satisfied or optimized).

Multi-Paradigm Modelling (MPM) principles are used throughout: model explicitly, most appropriate formalism(s), abstraction(s) for architectures and views, requirements and designs, with workflows modelled explicitly too.

4.22 Reflections about Digital Twins

Andreas Wortmann (Universität Stuttgart, DE)

Digital twins are starting to appear everywhere. In this talk, I reflect upon definitions, purposes, and tools related to engineering and operating digital twins based on the largest cross-domain mapping study on the topic to date.

4.23 Digital Twins for Learning Healthcare Systems

Steffen Zschaler (King's College London, GB)

I report on our experience building agent-based simulations and, further, digital twins for learning healthcare systems, the role that MDE has to play in this, and the challenges for future work.

4.24 Are Digital Twins going to rule the world?

Mark van den Brand (TU Eindhoven, NL)

Digital Twins play an important role in engineering complex (high-tech) systems. They allow for real-time analysis of engineered systems in order to detect anomalies, to predict maintenance, and to optimise behaviour. The use of model-driven techniques accelerates the development of digital twins. The focus of the Software Engineering and Technology group from the TU/e focuses on the efficient development of Digital Twins, by means of model reuse and advance orchestration of the reused models.

The use of models used in the engineering process leads to the question whether the digital twin can replace the supervisory control of the engineered system. What are the requirements to facilitate this:

- Supervisory control in the cloud by means of the engineered system
- Fast communication channels between physical entity and the engineered system (5G and beyond)
- Sensors, actuators and edge computation on the engineered system.

5 Problem Space Summary

5.1 Contexts

In this group, the participants focused on identifying the different contexts in which a Digital Twin may be produced, the goals and purposes of those Digital Twins, and their qualities and properties. We motivated our definitions based on the range of talks given during the seminar, as outlined in section 4. The talks included a variety of contexts for digital twins, including a variety of assets for which a digital twin is constructed, a variety of environments in which the digital twin is embedded, and a variety of purposes for which the digital twin was constructed. Below, we provide a short summary of the key discussion points within the group, and our findings:

From the range of talks demonstrated in section 4, we have seen that there are many different assets for which a digital twin can be developed. We came to define the 'context' of a digital twin as the asset type, as well as the asset's environment. Properties of the environment may include whether there is human intervention in the environment, whether the system is cyber-controlled or not, and whether the asset was engineered or is natural.

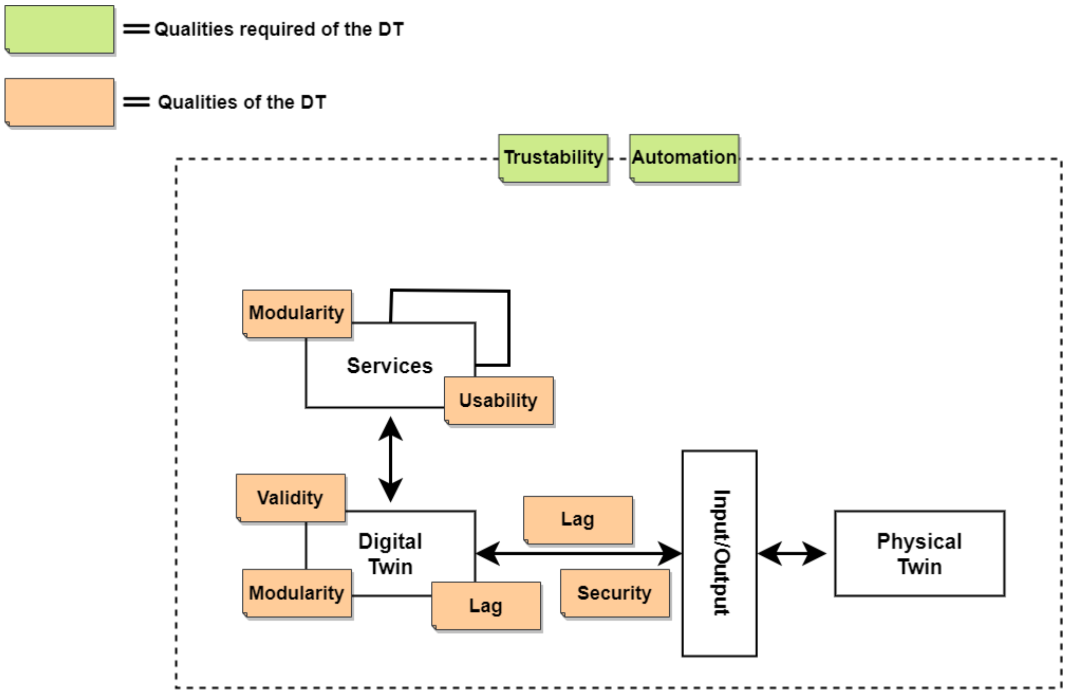

Figure 2 Digital Twin schema annotated with qualities.

5.2 Qualities of Digital Twins

In the problem space group, we then discussed the "qualities" of digital twins. I.e. the desirable properties of the system towards some measure or mitigation. We discussed the qualities of both the digital twin as an artifact, as well as the qualities that the digital twin must achieve for the user. Qualities apply to all engineered aspects including I/O and data flows between the physical and digital twins as well as the digital twin itself. We do not, however, comment on the qualities of the 'physical twin' as this is beyond the scope of the concerns of a digital twin developer.

During our discussions, we reached the opinion that qualities in the digital twin context are closely linked to qualities in the systems/software engineering context more generally. We, therefore, did not endeavor to produce a comprehensive list of qualities but instead aimed to identify some of the qualities of relevance to (different) digital twins.

In figure 2 we annotate a digital twin architecture diagram with the most relevant quality properties. These include:

- Validity: A measure to show that the digital twin is a "good enough" digital representation of the physical entity/twin
- Fidelity: How much reliance can be placed on the outputs (data, control signals, observations, etc.) of the digital twin.
- Trustworthiness: Closely linked to properties of fidelity and traceability. The degree to which stakeholders trust the validity of the digital twin as a whole.
- Modularity: The capacity to allow for the substitution of components such as services to fit the needs of the user in a given context
- Usability: The human-computer interaction properties of the system, and the standard of user experience the system provide
- Synchronisation: The properties of 'lag' between the digital and physical twin

- Security: Including considerations on risk analysis, data protection, and data biases

The goal of this group's discussions was to identify the fundamental scope and properties of digital twins. These properties were then used to motivate the requirements for the design space group as discussed in Section 6.

6 Design Space

The discussions in this group were targeting the design space aspect of using MDE techniques for Engineering Digital Twins. As a first discussion point, a conceptual model of digital twins was developed in order to gain a common understanding and terminology of the individual aspects and components of digital twins among the participants. The aim of this conceptual model was that it can be instantiated in various ways for specific domains, including cases where some of the components are missing.

A list of instantiations has been created for use cases provided by the participants in the domains on Anomaly Detection, Predictive Maintenance, Optimization, Diagnostics, and Policy Planning. The conceptual model was validated in these particular use cases, and in addition, for each use case, a design pattern and a list of open challenges was derived. In a final round of discussions, the results from the individual use cases were consolidated to come up with (i) a method for developing digital twins based on the individual design patterns, and (ii) a list of open challenges for designing digital twins using MDE techniques. The found challenges include (i) supporting experts in validating a DT system, (ii) capturing assumptions and the scope of a DT effectively, (iii) specification languages for DT systems, and (iv) design technologies for modeling privacy and security aspects of DTs.

7 Solution Space

The discussions in this group were targeting the necessary components needed for the MDE of Digital Twin Systems (DTSes). A model-based approach requires models that (1) may be used in the digital twin of the physical twin and (2) form the basis for generation and engineering tasks.

The distinction between descriptive, predictive, and prescriptive roles of models was discussed, as well as the fact that models are heterogeneous and composition covering various integration aspects is needed. Metamodels form the basis for model transformations from the various model kinds to other models or artefacts. MDE provides support for composition, consistency, an overall management of models. Overall, MDE provides a number of solutions for the challenges to be addressed in development of digital twins, but it is also evident that general MDE techniques and methods need to be adapted for the specific challenges of digital twin development.

Regarding connectivity, the group discussed how communication protocol choices depend on needs i.t.o. coupling, fidelity, service level, and quality of service; both for communication inside the DT as well as to its environment.

The importance of data storage, and of abstracting from data for purposes depending on the services the data is required for, were addressed. Suitable data pipelines, and integration with external services such as maintenance and operations management are important aspects regarding data and its management. Of course, models themselves need to be stored and processable so can be considered a form of data as well.

Finally, concerning services, we distinguished between internal and external services, with the latter for example covering data visualization and interaction with the digital twin. Important from the services point of view as well are data exchange, the availability of suitable execution and simulation engines, and data analytics. For service generation and operation, (meta)models also play an important role.

The group identified a list of challenges around the solution space for MDE of digital twins, concerning (1) integration and data exchange between different components; (2) consistency of models, data, and metamodels under evolution scenarios; (3) modeling environment interoperability (possibly including wrapping); (4) runtime support for time series data; (5) further standardization to allow interoperability and reuse of the artefacts involved in MDE based digital twins; (6) variability in DTs when considering product lines; (7) wrapping of tools patterns for the description and MDE of DTs; and (8) flexible definition of new services.

8 Conclusion

In this paper we have documented a fruitful week of discussions and community building in the the Dagstuhl Seminar **22362 – Model-Driven Engineering of Digital Twins**. This paper threat describes preliminary results of these discussions. We strongly expect that the discussions we started will continue in fruitful collaborations, potentially EU or national research programs addressing the question how to define digital twins in a more efficient and essential way and finally also lead to further events, e.g. such as a scientific conference about model-based engineering of digital twins.

As a short summary: we have identified that MDE does provide a number of solutions for the challenges to be addressed in development of digital twins, but it is also evident, that a specific adaptation of the general MDE techniques and methods is needed to address the specific challenges of digital twin development. And because the domain of digital twins is relevant, it is absolutely worthwhile addressing these challenges with MDE techniques.

Finally, we really want to thank all the local assistance, the organizational assistance and the strategic organizational people for making Dagstuhl this wonderful place to meet and discuss and put innovations forward. Thanks alot, we really appreciate it, especially inn these pandemic times.

References

1 Manuela Dalibor, Nico Jansen, Bernhard Rumpe, David Schmalzing, Louis Wachtmeister, Manuel Wimmer, and Andreas Wortmann. A cross-domain systematic mapping study on software engineering for Digital Twins. *Journal of Systems and Software*, November 2022.

2 John Hutchinson, Mark Rouncefield, and Jon Whittle. Model-driven engineering practices in industry. In *Proceedings of the 33rd International Conference on Software Engineering*, pages 633–642. ACM, 2011.

3 John Hutchinson, Jon Whittle, Mark Rouncefield, and Steinar Kristoffersen. Empirical assessment of mde in industry. In *Proceedings of the 33rd International Conference on Software Engineering*, pages 471–480. ACM, 2011.

4 Stuart Kent. Model driven engineering. In Michael Butler, Luigia Petre, and Kaisa Sere, editors, *Integrated Formal Methods*, pages 286–298, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

5 MDT. Digital twin consortium, "glossary of digital twins,". `https://www.digitaltwinconsortium.org/glossary/index.html`, 2021.

6 Paula Munoz, Javier Troya, and Antonio Vallecillo. Using uml and ocl models to realize high-level digital twins. In *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pages 212–220. IEEE, 2021.

7 Juha-Pekka Tolvanen and Steven Kelly. Metaedit+: defining and using integrated domain-specific modeling languages. In *The 24th ACM SIGPLAN conference companion on OOPSLA*, pages 819–820, 2009.

Participants

- Shaukat Ali
Simula Research Laboratory – Oslo, NO
- Balbir Barn
Middlesex University – London, GB
- Ion Barosan
TU Eindhoven, NL
- Nelly Bencomo
Durham University, GB
- Francis Bordeleau
ETS – Montreal, CA
- Tony Clark
Aston University – Birmingham, GB
- Loek Cleophas
TU Eindhoven, NL
- Benoit Combemale
University & IRISA – Rennes, FR
- Thomas Godfrey
King's College London, GB
- Georg Grossmann
University of South Australia – Mawson Lakes, AU
- Gabor Karsai
Vanderbilt University, US
- Djamel Khelladi
CNRS – IRISA – Rennes, FR
- Oliver Kopp
Mercedes-Benz AG – Stuttgart, DE
- Daniel Lehner
Johannes Kepler Universität Linz, AT
- Bernhard Mitschang
Universität Stuttgart, DE
- Paula Muñoz Ariza
University of Mĺaga, ES
- Alfonso Pierantonio
University of L'Aquila, IT
- Fiona A. C. Polack
University of Hull, GB
- Matthias Riebisch
Universität Hamburg, DE
- Bernhard Rumpe
RWTH Aachen, DE
- Holger Schlingloff
HU Berlin, DE
- Markus Stumptner
University of South Australia – Mawson Lakes, AU
- Antonio Vallecillo
University of Málaga, ES
- Mark van den Brand
TU Eindhoven, NL
- Hans Vangheluwe
University of Antwerp, BE
- Andreas Wortmann
Universität Stuttgart, DE
- Steffen Zschaler
King's College London, GB