Towards Machine Learning-based Digital Twins in Cyber-Physical Systems

Felix Theusch[1,*], Lukas Seemann[2], Achim Guldner[2], Stefan Naumann[2] and Ralph Bergmann[1,3]

[1]German Research Center for Artificial Intelligence (DFKI), Branch Trier University, 54296 Trier, Germany
[2]Institute for Software Systems, Trier University of Applied Sciences, Environmental Campus Birkenfeld, 55761 Birkenfeld, Germany
[3]Artificial Intelligence and Intelligent Information Systems, Trier University, 54296 Trier, Germany

Abstract

The use of Artificial Intelligence, and especially Machine Learning methods, promise to play key roles in the development of Digital Twins due to their outstanding properties in processing large IoT data streams. However, so far, there is a lack of research on the systematisation of Machine Learning-based Digital Twins (MLDTs) as well as on their methodological development and implementation processes in productive environments. The scientific literature describes various applications of MLDTs - even if they are not called this way - and specialised methods and architectures, but a generic reference model is still missing. Therefore, this paper proposes a systematisation of the characteristics of MLDTs and their specific challenges. Furthermore, a first proposal of a process model for the systematic development of MLTDs according to the Machine Learning Operations (MLOps) paradigm is presented as a tentative instance of a future reference model for MLDTs. We incorporate established software development methods as well as insights gained from the examination of several industrial applications in the field of water resource management, one of which we present during the paper. We expect that the process model allows practitioners to consistently develop and maintain MLDTs and researchers to find potentials and research gaps.

Keywords

Digital Twin, Machine Learning, Characterisation, MLOps, Water Resource Management, Process Model

AI4DT&CP@IJCAI 2023: The First Workshop on AI for Digital Twins and Cyber-Physical Applications in conjunction with 32nd International Joint Conference on Artificial Intelligence, 19th - 25th August 2023, Macao, S.A.R
*Corresponding author.
felix.theusch@dfki.de (F. Theusch); l.seemann@umwelt-campus.de (L. Seemann); a.guldner@umwelt-campus.de (A. Guldner); s.naumann@umwelt-campus.de (S. Naumann); bergmann@uni-trier.de (R. Bergmann)
https://www.dfki.de/en/web/about-us/employee/person/feth02 (F. Theusch);
https://green-software-engineering.de (L. Seemann); https://green-software-engineering.de (A. Guldner);
https://green-software-engineering.de (S. Naumann);
https://www.uni-trier.de/en/universitaet/fachbereiche-faecher/fachbereich-iv/faecher/informatikwissenschaften/professuren/wirtschaftsinformatik-2/chair (R. Bergmann)
0009-0002-0100-9579 (F. Theusch); 0009-0007-1667-9053 (L. Seemann); 0000-0002-7532-4523 (A. Guldner); 0009-0000-6542-2229 (S. Naumann); 0000-0002-5515-7158 (R. Bergmann)

1. Introduction

Cyber-Physical Systems (CPSs) and their seamless connection of control devices, physical assets, and IT systems towards an Internet-of-Things (IoT) are current megatrends and promise improvements in efficiency and resilience in almost all domains [1]. Recently, the CPS paradigm has also been rapidly expanded to include approaches to process planning, monitoring, and control [2]. From their first mention at the beginning of the 21st century, Digital Twins have become a cornerstone of CPSs, providing virtual representations of real physical objects, systems, or processes. In doing so, they provide the nexus between the physical and digital world [3]. Based on the data gathered from IoT devices and systems, they can perform complex simulations and data-driven decisions to optimise the control of the physical CPS layer [4]. Meanwhile, Artificial Intelligence (AI) has become a central enabler in the further development of Digital Twins, both in research and in industrial applications [5]. Groshev et al. [5] see AI as the central puzzle piece in the Digital Twin architecture to tackle common challenges within the concept such as the effective usage of real-time data streams, the fulfilment of safety or performance requirements, and optimal network usage.

Grieves [3], who was instrumental in establishing the concept of the Digital Twins, identifies "Intelligent Digital Twins" built on AI technologies as the next evolution of the Digital Twin paradigm [6]. On closer examination, the main focus is often placed on the use of Machine Learning (ML) methods and their significant advantages in processing large IoT data sets using statistical models [7]. Despite all the advantages of using ML in the environment of Digital Twins, it also leads to specific questions and challenges, for example regarding an effective Digital Twin ML model management [8], that have only been addressed in a few scientific publications so far. For instance, there is a lack of characterisation and established methods to support an effective and qualitative implementation of ML-based Digital Twins and their productive operation. Therefore, in this paper, we aim to narrow down and characterise *ML-based Digital Twins* (MLDTs) in CPSs, based on findings in related literature and investigations of productive ML-based Digital Twin applications in the water industry. This contributes to the clarification and definition of the concept of MLDTs and to a deeper understanding of their properties. Based on best practice examples from a case study in the domain of water resource management and established software implementation methods, a first outline of a process model for the development and productive operation of MLDTs is presented.

The outline of this paper is as follows: Sect. 2 describes the role of AI and ML in the context of Digital Twins and provides a definition of MLDTs and their characteristics. The use of this type of Digital Twins is demonstrated in Sect. 3 with a practical example showing the case study of artificial neural networks (ANN) to control a water distribution network. As the main focus of this paper, in Sect. 4, we develop a first approach for a process model for the implementation and operation of MLDTs in the context of CPSs. This process model can be seen as a first instance of a future reference model for MLDTs.

2. ML-Based Digital Twins and their challenges

As already stated in the introduction, technologies from the field of ML are increasingly applied in the context of Digital Twins. Therefore, we first define the term ML-based Digital Twin, based on an extensive literature review and our experiences from practical Digital Twin applications. From this, we derive the most important components of an MLDT and identify its specific challenges.

2.1. Related Work

There are many different methods for the realisation of Digital Twins, e. g., Geographical Information System (GIS), Building Information Modelling (BIM), or Computer-Aided Design (CAD) models [9] or the use of data-related technologies such as OPC UA[1] or AutomationML[2] in manufacturing [10] and various frameworks for DTs have been presented (cf. [11, 12]). According to Tao et al. [13], the modelling approaches of a digital twin can be divided into four dimensions: A *geometric* digital twin model is used to describe the geometric properties, whereas a *physical* model represents the physical properties, such as fluid dynamics of the real entity. Digital twins based on a *behavioral* model represent the dynamic responses of the physical entity to internal and external mechanisms and uses similar tools to physical modelling, while the *rule* model reflects the real world by incorporating historical data to extract tacit knowledge. The latter include, in particular, digital twins, which are based on machine learning methods and will be defined in more detail in the remainder of this paper.

Min et al. [14] propose a framework for ML-based digital production control optimisation in the petrochemical industry and demonstrate their solution with a case study. Furthermore, they propose different chronological steps to develop an MLDT based on a mathematical simulation model. According to their concept, the ML-based "Digital Twin Model" is created through model training and validation based on prepared, historical training data. Ritto and Rochinha [15] investigate the integration of physics-based models with ML to construct a Digital Twin to identify structural damage to wind turbines in real time. By transforming the physical models into an ML model, it is possible to benefit from its performance in processing a large amount of IoT data.

The increasing importance of ML for Digital Twins leads to a need for practice-oriented process models that control the development process of Digital Twins based on ML techniques. With CRISP-DM and MLOps there are established procedure models from the Data Mining and ML perspective. So far, only a few works address concrete procedures for Digital Twins that focus especially on the ML aspect [16]. To the best of the authors' knowledge, there are no publications to date on a deeper systematisation of Digital Twins based on ML models in the direction of a reference model (e. g., according to [17]), which guides focused research and application development of MLDTs.

[1] https://opcfoundation.org/about/opc-technologies/opc-ua/ [2023-07-03]
[2] https://www.automationml.org/ [2023-07-03]

2.2. Definition and Characteristics of ML-Based Digital Twins

In the context of this paper, an *ML-based Digital Twin* is defined as a special type of Digital Twin, where ML models form the central basis for the twin's ability to model and simulate the physical world. These models are adapted to the specific requirements of the Digital Twin by training with large amounts of data and can also recognise previously unknown patterns and react to unknown incoming data in real time.

In comparison to other types of Digital Twins, MLDTs defined in this paper have some special characteristics which are summarised in the following.

Task specialisation: On the one hand, Digital Twins based on ML methods can be applied in all kinds of CPSs, regardless of the domain (manufacturing, smart grids, etc.) or the task (intelligent control, condition monitoring, etc.). On the other hand, an individual MLDT is trained to perform a very specific task. This means, for example, that a neural network, optimised for industrial quality analysis [14], cannot be used simultaneously in the energy efficiency improvement of buildings [16].

Physics-based model integration: Physics-based models in the context of Digital Twins are computer-based models that mimic the physical properties and behaviours of real objects or systems, for example, representing energetic or thermal and other physical properties in mathematical form. MLDTs are able to benefit from the strengths of these models (interpretability, generic applicability, etc.) on the one hand and from the performance of data-driven models on the other hand through the targeted integration of physical models, e. g., to supplement or generate training data [15].

Data-driven model building: Every Digital Twin needs models to represent the behaviour of its real counterpart and to be able to make predictions or perform simulations based on (IoT) data [10]. While, for example, BIM, GIS, or CAD-based Digital Twins usually derive their information from (3D) models of buildings, infrastructures, or other physical objects, MLDTs are, at their core, based on ML models applied to real-time data or other sources [14], e. g., from IoT-sensors in the field. This enables MLDTs to recognise previously unknown patterns or correlations and to make higher-quality decisions based on real-world data.

Data complexity and processing: MLDTs are specialised in processing large amounts of data from various sources in near real-time and can process them, for example, in cloud environments [2] or on the edge [18], depending on the individual use case requirements for performance and confidentiality. Furthermore, by integrating additional (domain) knowledge in the different phases of the ML process, the robustness of the MLDT can be increased. Thus, prior knowledge can not only support the selection of a suitable model or the interpretation of model predictions, but also help in the data preparation phase to clean the data, fill in missing values, or remove outliers [19].

Adaptability and learning: Digital Twins operate in dynamic and permanently evolving environments and must therefore be able to adapt changes as efficiently as possible. These changes can be both sudden or gradual, as well as unconscious or planned (e. g., the planned change of technical components versus their gradual wear and tear). The (real-time) processing and analysis of large amounts of data, gives the MLDT the ability to adapt to changes in their environment and provides a major advantage over other forms of Digital Twins [14].

Federation and transfer of learning outcomes: Digital Twins often process sensitive data

concerning intellectual property or personal data. By applying federated ML techniques in the context of Digital Twins, distributed (cross-company) data sets can be used efficiently to increase model performance and scale, while preserving data privacy. In addition, MLDTs can use transfer learning to access pre-trained models and apply them to similar problems [20].

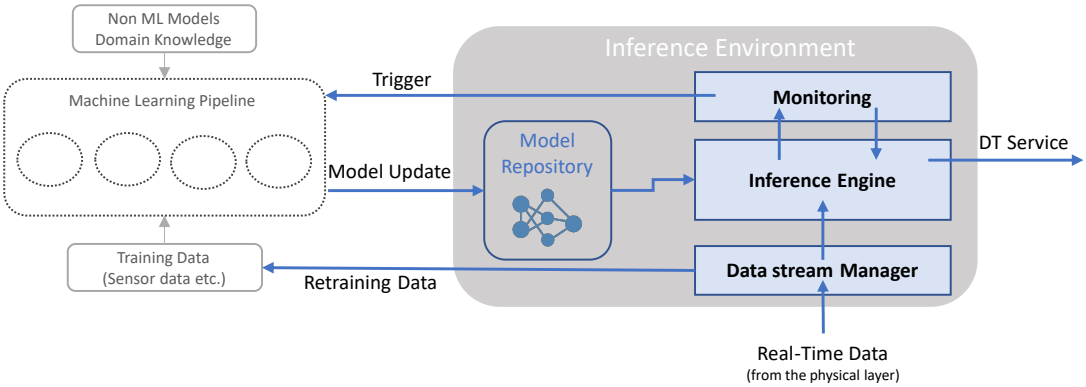

Figure 1: Structure of an MLDT with related ML pipeline

Fig. 1 shows the rough structure of an MLDT with the associated ML pipeline. The exact structure of an individual MLDT depends on various criteria, such as whether it is operated locally or in the cloud, or whether a federated learning approach is used. The ML pipeline generates an ML model, for example a trained neural network, and loads it into a model repository on the *inference environment*, which applies the trained model to new, incoming (IoT) data from the physical layer. The model training infrastructure and the MLDT inference infrastructure are often separated from each other so that they can each be optimised for their respective tasks [14]. As described previously, the training phase of the MLDT often integrates both historical (IoT) sensor data and information from other Digital Twin domain models, e. g., using feature engineering steps or generating additional training data.

Within the inference environment, the trained model is managed and applied by the inference engine to real-time data from the physical layer. To do so, the data stream manager prepares the incoming data for the inference engine, for example by cleaning or normalising it. This enables the MLDT to offer its service and to perform control, simulation or monitoring tasks. To assure prediction performance of the MLDT measured by specific ML performance indicators, (automated) triggers can initiate a retraining of the models if threshold values are exceeded or not reached, for example in case of quality issues.

2.3. Challenges in the implementation of ML-based Digital Twins

The development of an MLDT in a CPS is a highly complex, interdisciplinary process that requires both a profound understanding of the subject domain and its underlying technical processes, as well as in-depth knowledge of data analytics and ML, software development, and project management.

The performance of the Digital Twin application depends significantly on the choice of a well-fitting and robust ML model. On the one hand, a sufficient amount of high-quality data is essential for the training of ML models. This requires, in particular, the preparation of the data, including the cleaning, transformation, and integration of different data sets from different sources. This step often also requires the conversion of specific knowledge models, for example based on CAD, BIM, or GIS information, into a format adequate for the learning process, for example through synthetic data generation in domain systems [15]. On the other hand, Digital Twins operate in dynamic and permanently evolving environments and therefore must be able to adapt to changes as efficiently as possible. In all cases, the ML pipeline must be created robust enough to adapt (autonomously) to changing environmental conditions or to allow the retraining of their models in a structured way during live operation [14]. The overall challenge is to develop a robust and powerful MLDT that can evolve and learn throughout its entire lifetime (cf. continual lifelong learning) [21]. This results in ML-specific challenges, such as the avoidance of catastrophic forgetting tendencies in neural networks or the respective ML model by developing adequate strategies [21].

Another challenge, in regard to especially high compute- and data-intensive components of MLDTs (like the training processes, model federation, etc.) is their resource- and energy efficiency and thus, their impact on the environment. While it is true that, as stated in the introduction, the efficiency in the underlying domain-specific processes can be improved through the optimisation with MLDTs (Green by IT) [22, 23], it is important that the systems themselves are built in a way that they adhere to sustainability specifications and do not become resource drivers [24, 25]. This is even more significant when we consider that the IoT systems that form the basis of the MLDTs in the physical world are usually lightweight and distributed in the field [26]. Thus, it is important to ensure that the hardware resources are available and that the energy supply is sufficient (batteries, solar cells, energy harvesting, etc.).

3. Case Study: ML-Based Digital Twins in Water Resource Management

AI and the DT paradigm also find application in the Water Resource Management (WRM) domain. As a result of a thorough literature search, the authors were able to identify four categories in WRM into which previous publications can be classified. While being prevalent in WRM, these categories certainly are not WRM-specific and can also be applied to other domains.

For one, MLDTs in WRM are used in the context of *Forecasting* to predict the behaviour of the water cycle through the timeline. Typically needed forecasts in WRM include the required extraction from water sources [27] or water demand patterns [28]. Another category is *Monitoring & Maintenance*, considering that various papers show that the seamless operation of water utilities can be supported by MLDTs (e. g. [29, 30]). The next predominant category is *Optimisation & Controlling*, as for example, energy efficiency through optimal pump and valve operation is also a significant topic in WRM [29] and ML algorithms are a perfect fit for these optimisation problems. Lastly, Digital Twins are used for *Decision Support*, e. g., for infrastructure planning [29] or employee training [31]. In most use cases, the Digital Twins are a holistic representation of the water utility, so usually they can fit into more than one category

as they serve multiple purposes.

The case study in this paper is about the MLDT for a drinking water network in southwestern Germany and can be assigned to the *Forecasting* and *Optimisation & Controlling* categories.

ANN-control of Drinking Water Distribution Systems

The "Stadtwerke Trier" (SWT), a municipal utility, operates a drinking water network which has a capacity of 10 to 11 million cubic meters of water, serving a population of approximately 110,000 residents in the city, located in southwestern Germany. The city's drinking water network is supplied by two water utilities, one of which obtains its water from a reservoir at a higher altitude and generates about 1 million kWh of energy annually via two turbines (2 x 250 kW) integrated in the water inlet. In addition, four rooftop and one ground-mounted system provide a cumulative photovoltaic (PV) capacity of approximately 500 kWp. Compared to the electricity generation, the energy consumption of the grid is also considerable and ranges from 1.6 to 1.7 million kWh per year, especially due to several water pumps that are used for transferring the water to different storage reservoirs and grid zones within the city due to the topographical location.

In 2017, SWT started an automation project together with the industry supplier Xylem[3], whose objective was to create a Digital Twin for the online simulation and energy-efficient optimisation of drinking water distribution based on ANNs. The overarching target parameters of the Digital Twin are the provision of drinking water in the required quantity and water pressure for the end users, while at the same time minimising energy consumption. Similar to the division of the water distribution system (WDS) into separate water network zones, local optimisation takes place in the Digital Twin at the level of these WDS zones, along with global optimisation at the level of the overall network. The structure of the MLDT used in this case study can be seen in Fig. 2.

To enable the local optimisation, two types of zone-specific models are required. On the one hand, water demand prognosis requires *forecasting models* that have been trained on the basis of historical consumption and weather data and represent the water demand of a WDS zone. On the other hand, there is also the need for *infrastructure models* that replicate the hydraulic and energetic behaviour of the physical components. Both energy consumers (pumps, valves, etc.) and energy producers (turbines) are modelled. Such simulations are already available in the form of deterministic models which are provided by a domain-specialised software for the simulation, calculation and analysis of utility networks[4]. Although simulations of individual WDS zones with these deterministic models on their own is possible, they are very time and resource consuming and therefore not suitable in live operation. The modelling of the physical WDS properties by ANNs ensures significant performance gains in the simulation of optimised operation modes in live operation. Nevertheless, the data from simulation runs of the deterministic infrastructure model combined with expert knowledge from the domain are used as a valuable training input for the ANN. With these two types of trained models for

[3]The ANN-based control of water distribution networks and water treatment plants is offered by Xylem under the brand name BLU-X: https://www.xylem.com/de-de/products--services/digital-solutions/blu-x-treatment-plant-optimization/ [2023-07-03]

[4]SWT uses STANET for WDS modelling and calculation: https://www.stafu.de/en/home.html [2023-06-30]

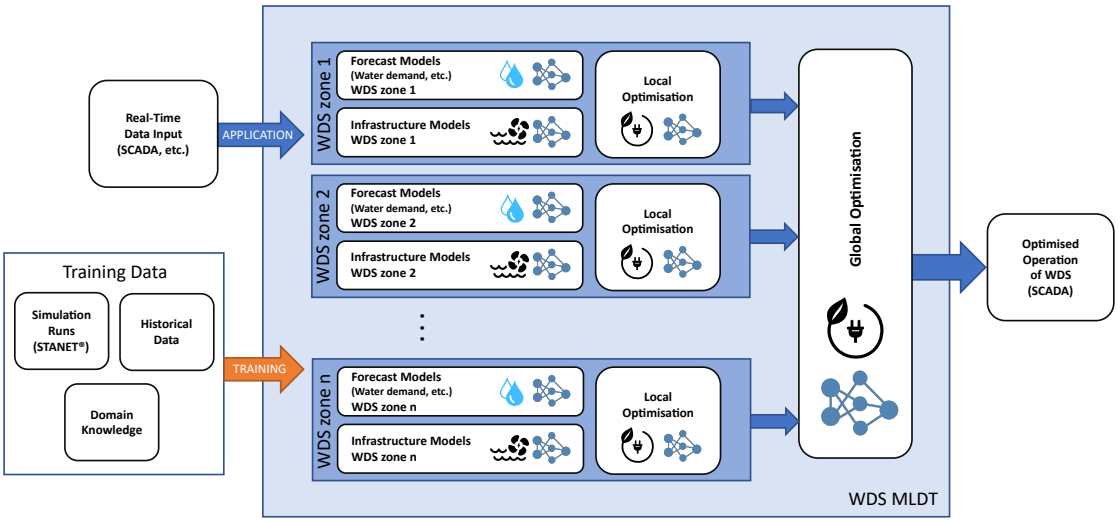

Figure 2: Optimised operation of the SWT WDS through an MLDT

each WDS zone, an optimal control of the physical components is found for energy-optimised operation in terms of power generation and consumption.

Based on the incoming real-time data, provided by the SCADA system or additional data sources and considering different interactions between individual WDS zones, a global overall optimisation of the WDS is carried out according to the principle of modular ANNs. As a result, the optimised operational suggestions can be carried out by the WDS SCADA system to control the physical layer (e. g. pumps, valves, etc.). By abstracting the WDS-based on the MLDT presented here, the self-consumption of green energy can be increased from 60 to 90 % through optimised operation[5].

Since the MLDT is the virtual representation of a highly dynamic, physical network, it has to be considered as a constantly evolving system. For example, an even deeper integration of renewable energies into the described system is planned in the future by considering forecasts of PV power generation. In order to be able to integrate new models like these into the existing system and, if necessary, also update the existing models, it is helpful to have clearly defined processes in which system adaptations and MLDT applications can run in parallel.

4. Process Model for the Development of ML-Based Digital Twins

The development of Digital Twins, and in particular of MLDT applications, is a cross-organisational, time- and knowledge-intensive process that requires a variety of different skills and collaboration between domain experts (often supported by external technical planning

[5]https://www.swt.de/p/CO2_freies_Trinkwasser_f%25C3%25BCr_Trier-5-7330.html [2023-06-30]

offices), automation engineers, component manufacturers, data scientists, and ML engineers. In the following sections, we present a first approach for a six-phase process model that structures the different work steps of MLDT development. For this purpose, we first describe the methodology of the process model definition on the basis of established data science and software development procedures as well as best practice examples.

4.1. Methodology

ML applications are typically complex and require careful planning, development, and implementation to ensure they can be used safely and effectively in a production environment [32]. In practice, ML projects often fail because insights from the data exploration phases are not effectively applied in productive ML models, which can lead to inaccurate predictions, higher costs, and risks. To structure the procedure of effective development of reliable operational ML applications, the Machine Learning Operations (MLOps) approach has emerged in recent years. MLOps is a cross-functional, collaborative, and iterative paradigm that adopts established DevOps practices from software development, such as the Continuous Integration (CI) and Continuous Delivery (CD) principles, and combines them with data engineering and ML methods [33].

The development of an overarching process model is strongly oriented on selected scientific publications on MLOps on the one hand and on expert interviews and studies of ML application projects in water resource management on the other. With regard to a framework MLOps structure, the procedure model described here is loosely oriented on the MLOps architecture according to [33]. As a deeper look at the practice also shows that structured data-mining and software development methods have already been adopted in industrial water management applications, we base the detailed MLOps workflow definition for the data-mining and engineering steps on CRISP-DM [34], the de-facto standard workflow for industrial data science projects.

4.2. Six-Phase process model

Fig. 3 shows a six-phase process model for the development of MLDTs, developed according to the methodology described above. Each phase is divided into a different number of tasks which are ordered chronologically (indicated by the numbering in the round brackets).

Phase 1: Digital Twin Alignment and ML Problem Definition

Usually, data science or ML projects start with a phase in which the problem is specified and a preliminary project plan is set up. In CRISP-DM, this initial step is very closely linked to the phase of data understanding, since the problem definition is based on hypotheses about possible data patterns [34]. Based on this, the first steps of the proposed process model include the (1) definition of the business problem to be addressed by the Digital Twin and the definition of an overall project plan that regulates the allocation of tasks between domain experts, data scientists, ML engineers, or software developers. Subsequently, the raw data required for a rough (2) exploratory data investigation are compiled and subjected to an initial (3) data quality check. Based on these initial findings and the defined business goals associated with Digital

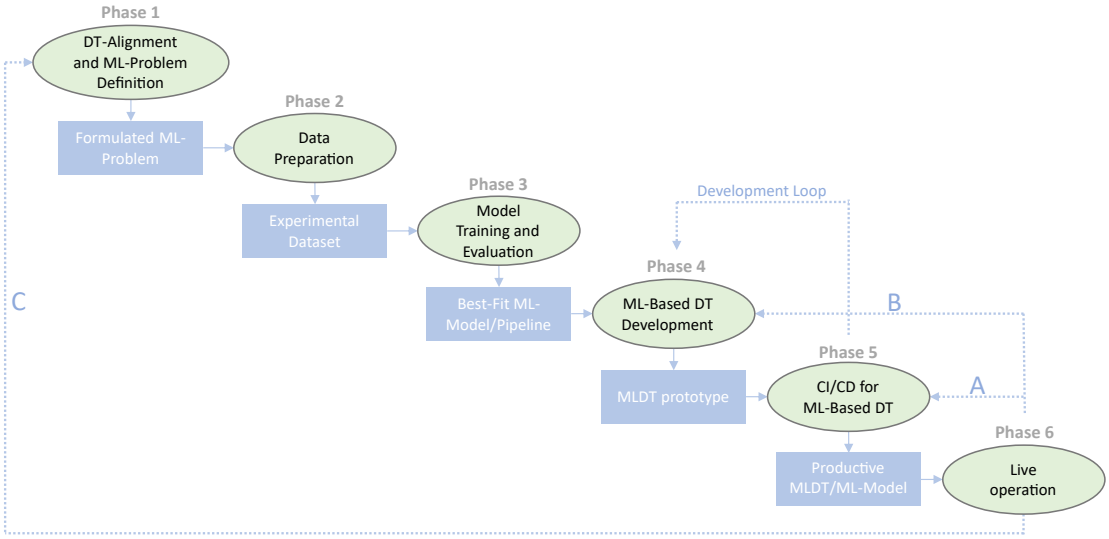

Figure 3: Six-phase process model for the development of MLDTs

Twin development, the (4) ML problem (regression, classification, etc.) to be solved is defined at the end of the first process model phase.

Phase 2: Data Preparation

IoT data is usually not flawless and its quality can vary greatly. Therefore, before training the ML models, it is necessary to (5) remove faulty data or (e. g. synthetically) fill in missing data and perform a final data quality check. The (6) feature transformation and engineering step involves the preparation and processing of input features, including conversion of features into a processable format and creation of new features or modification of existing features to improve the performance of the model. Parallel to the feature engineering task, the (7) integration of additional data takes place, for example also from external sources, which are necessary for the execution of the Digital Twin service. This can include, for example, weather data for Digital Twins in water treatment or energy market data for planning energy-optimised production processes [31]. The outcome of the "Data Preparation" phase is a cleaned and integrated data set that is aligned to the next phase for the training and evaluation of an ML model tailored to the Digital Twin Service.

Phase 3: Model Training and Evaluation

During the third phase, the most suitable ML methodology with regard to the problem defined in phase 1 is to be evaluated, and its learning result subsequently stored as a model. At the beginning, an (8) exploratory data analysis (EDA) takes place, in which the surveyed data is analysed with regard to the statistical correlations of their features. The choice of the EDA environment, for example Matlab [35], R, or Python [36], depends on the project requirements,

the skills and knowledge of the data analysts and the specific domain. Since the previous step is very closely related to (9) model training (different ML approaches require differently prepared data), these activities can be carried out in parallel. ML models are the result of the learning process and depend on the method and data used to train them. In the domain of WRM, these are for example ANNs for process control tasks (see case study in sect. 3) or support vector machines for predicting the water demand [37]. Different model parametrisations allow (10) model validation based on selected performance metrics, for example the Mean Squared Error (MSE) or the Mean Absolute Error (MAE) for the assessment of regression models for the prediction of water demand or the expected wastewater quantity based on weather forecasts. At the end of the Model Training and Evaluation phase, the (11) most promising ML model is evaluated against the Digital Twin requirements defined in phase 1 and a decision is made whether to initiate the development of a productive MLDT environment (phase 4). Phase 3 identifies the best-fit ML model through experimental training, but training on productive data is completed in later steps.

Phase 4: ML-Based DT Development

The fourth phase covers all activities to build an infrastructure on which the MLDT can be evaluated and ultimately operated with productive data. Normally, the infrastructure for CI/CD is divided into at least two environments - with one being the testing or pre-production stage and the other one being the productive environment. The aim of the development in this phase is both an ML pipeline that regularly learns updated ML models initiated by various rules or triggers and the inference environment that applies these models to real-time data (see Fig. 1). Phase 4 starts with the (12) specification and setup of the system infrastructure, where, for example, basic decisions are made about the structure of the server environments (cloud or on-premise, server configuration etc.) and all necessary Application Programming Interfaces (APIs) are defined. To automate the CI and CD of updated ML models and software components of the MLDT, a corresponding (13) CI/CD pipeline must initially be established. Afterwards, step (14) involves the development of the previously defined interfaces and software components as well as the development of the ML pipeline and the inference server. Every newly developed component or functionality triggers the CI/CD pipeline in the pre-production stage, which is further described in phase 5 as the development loop. To monitor the performance of the MLDT in later live operation, various (15) triggers are set up at the end of the implementation phase that initiate either an adjustment of the system environment, the ML pipeline, or the retraining of the ML model (phase 6).

Phase 5: CI/CD for ML-Based Digital Twin

In phase 4, a prototype of the MLDT was provided, but it is still not in live operation and does not yet control the real asset. Before the Digital Twin becomes productive, its functionalities have to be integrated, tested, and deployed on the final infrastructure. This is partly based on the MLOps-approach proposed by Google[6] for CI/CD in ML. During the execution of the

[6] https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning [2023-06-30]

(16) CI pipeline, the software fragments are continuously checked for errors in order to detect and rectify problems at an early stage. Subsequently, in the (17) CD step, the MLDT system, including the ML pipeline or the modified system components, are deployed on the server. In the (18) continuous training step, the ML model is trained, based on the developed and previously deployed ML pipeline. After this step, the trained ML model is stored in the repository of the Inference Environment during the following (19) Model CD.

As mentioned in the previous phase, the proposed CI/CD pipeline (including steps 16, 17, 18, and 19) runs in a development loop as newly developed features are tested, integrated, and deployed in the pre-production environment. When all tests are passed and all MLDT requirements from phase 1 are met, in step (20) the CI/CD pipeline is executed on the productive environment. This final productive CI/CD run concludes the development, as the ML model is now trained on productive data and the MLDT is ready for operation on the productive environment. This whole phase is not only carried out during the initial development of the MLDT, but also during partial adjustments of the system, e. g., adjustments in the ML pipeline or the integration of new data.

Phase 6: Live operation

After the CI/CD phase and the successful go-live, the performance of the Digital Twin is (21) monitored during operation. Thus, adequate re-training methods must be applied in this step so that the MLDT learns continuously and does not lose relevant knowledge at the same time (see section 2.3). The triggers implemented in phase 4 (step 15) can initiate different workarounds: A *type A* trigger (annotations in Fig. 3) provides the impulse that the ML model needs to be retrained (step 18), for example in the case of decreasing prediction quality, recognisable by the deterioration of various performance metrics. A *type B* trigger indicates a different workflow and means that either changes to the software environment (e. g. update, adaptation of API) or to the ML pipeline (integration of new features, hyperparameter modification) must be done. Digital Twins are not static structures, but are often further developed with regard to their business case after their initial implementation. Depending on the intensity of the intervention, it may be necessary to specify these adjustments again starting with phase 1 (*type C* trigger). This could be the extension of the Digital Twin, where entirely new functionalities are to be integrated, e. g., the integration of PV power forecasts into an existing ANN control of the water distribution network.

4.3. Discussion

In the context of the increasing importance of Digital Twins based on ML methods, the process model described above is a first comprehensive attempt to take the specifics of MLDT into account. This involves coordinating the preliminary steps of business goal definition and ML problem formulation as well as the learning of suitable ML models and the development of a suitable inferencing environment and its operation. By integrating established data mining, ML, and software development paradigms with best practices from practical Digital Twin implementation projects, the process model provides a structural framework for interdisciplinary collaboration in MLDT projects. It fosters structured cross-organisational collaboration among

different experts with different skills. At the same time, the strict DevOps focus ensures fast and secure development and deployment of the necessary software components, with particular emphasis on regular updates of the fundamental ML models. This meets the high demands of Digital Twins regarding their adaptability in changing environments.

5. Conclusion and Future Work

As evaluated in this paper, Digital Twins are enabled to process large amounts of data in real time through the use of ML and can thus perform intelligent control and optimisation tasks. This paper therefore proposes an initial characterisation of MLDTs and specifies the associated challenges. To address these challenges, a six-phase process model for the development, deployment, and operation of MLDTs was proposed that considers the aspects of Digital Twin problem definition and evaluation of a suitable ML model as well as its productive implementation within a CPS. The adoption of CI/CD practices ensures integrated monitoring of model performance as well as (semi-) automated model retraining and updating.

Despite the widespread use of ML techniques in the context of Digital Twins, however, there is a lack of comprehensive definitions and differentiation from other Digital Twin modelling approaches. To further encourage research in Machine Learning-based Digital Twins, we propose the development of a general reference model by combining deductive and inductive elements. This should include a comparison of similar reference models from the field of CPSs and findings from an extensive literature study as well as further investigations in industrial MLDT applications, for example in the field of manufacturing or water management.

Acknowledgments

We would like to thank the Ministry for Climate Protection, Environment, Energy and Mobility of Rhineland-Palatinate (Ministerium für Klimaschutz, Umwelt, Energie und Mobilität Rheinland-Pfalz) for the financial support and assistance of the research project "Digital twin in water resource management" in the context of which this publication was realised. We would also like to thank Mr. Nicolas Wiedemeyer from Stadtwerke Trier and Mr. Michael Natschke from Xylem Water Solution GmbH for contributing their expertise to the case study. This work was also funded in part by the German Federal Ministry for Economic Affairs and Climate Action project "Energy-efficient analysis and control processes in the dynamic edge cloud continuum for industrial manufacturing" (EASY) under Grant 01MD22002D.

References

[1] E. A. Lee, Cyber physical systems: Design challenges, in: 2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC), IEEE, 2008, pp. 363–369.

[2] B. Bordel, R. Alcarria, D. S. de Rivera, T. Robles, Process execution in cyber-physical systems using cloud and cyber-physical internet services, The Journal of Supercomputing 74 (2018) 4127–4169.

[3] M. Grieves, J. Vickers, Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems, Transdisciplinary perspectives on complex systems: New findings and approaches (2017) 85–113.
[4] W. Kritzinger, M. Karner, G. Traar, J. Henjes, W. Sihn, Digital twin in manufacturing: A categorical literature review and classification, Ifac-PapersOnline 51 (2018) 1016–1022.
[5] M. Groshev, C. Guimarães, J. Martín-Pérez, A. de la Oliva, Toward intelligent cyber-physical systems: Digital twin meets artificial intelligence, IEEE Communications Magazine 59 (2021) 14–20.
[6] M. Grieves, Intelligent digital twins and the development and management of complex systems, 2022. URL: https://digitaltwin1.org/articles/2-8.
[7] K. Alexopoulos, N. Nikolakis, G. Chryssolouris, Digital twin-driven supervised machine learning for the development of artificial intelligence applications in manufacturing, International Journal of Computer Integrated Manufacturing 33 (2020) 429–439.
[8] S. Schelter, F. Biessmann, T. Januschowski, D. Salinas, S. Seufert, G. Szarvas, On challenges in machine learning model management, IEEE Data Engineering Bulletin (2015).
[9] C. C. Menassa, From bim to digital twins: A systematic review of the evolution of intelligent building representations in the aec-fm industry, Journal of Information Technology in Construction (ITcon) 26 (2021) 58–83.
[10] M. Liu, S. Fang, H. Dong, C. Xu, Review of digital twin about concepts, technologies, and industrial applications, Journal of Manufacturing Systems 58 (2021) 346–361.
[11] Y. Zheng, S. Yang, H. Cheng, An application framework of digital twin and its case study, Journal of Ambient Intelligence and Humanized Computing 10 (2019) 1141–1153.
[12] F. Tao, H. Zhang, A. Liu, A. Y. Nee, Digital twin in industry: State-of-the-art, IEEE Transactions on industrial informatics 15 (2018) 2405–2415.
[13] F. Tao, B. Xiao, Q. Qi, J. Cheng, P. Ji, Digital twin modeling, Journal of Manufacturing Systems 64 (2022) 372–389.
[14] Q. Min, Y. Lu, Z. Liu, C. Su, B. Wang, Machine learning based digital twin framework for production optimization in petrochemical industry, International Journal of Information Management 49 (2019) 502–519.
[15] T. Ritto, F. Rochinha, Digital twin, physics-based model, and machine learning applied to damage detection in structures, Mechanical Systems and Signal Processing 155 (2021) 107614.
[16] T. Y. Fujii, V. T. Hayashi, R. Arakaki, W. V. Ruggiero, R. Bulla Jr, F. H. Hayashi, K. A. Khalil, A digital twin architecture model applied with mlops techniques to improve short-term energy consumption prediction, Machines 10 (2021) 23.
[17] C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, R. Metz, B. A. Hamilton, Reference model for service oriented architecture 1.0, OASIS standard 12 (2006) 1–31.
[18] Y. Lu, X. Huang, K. Zhang, S. Maharjan, Y. Zhang, Communication-efficient federated learning for digital twin edge networks in industrial iot, IEEE Transactions on Industrial Informatics 17 (2020) 5709–5718.
[19] P. Klein, N. Weingarz, R. Bergmann, Enhancing siamese neural networks through expert knowledge for predictive maintenance, in: IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning: Second International Workshop, IoT Streams 2020, and First International Workshop, ITEM 2020, Co-located

with ECML/PKDD 2020, Ghent, Belgium, September 14-18, 2020, Revised Selected Papers 2, Springer, 2020, pp. 77–92.
[20] B. Maschler, D. Braun, N. Jazdi, M. Weyrich, Transfer learning as an enabler of the intelligent digital twin, Procedia CIRP 100 (2021) 127–132.
[21] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, S. Wermter, Continual lifelong learning with neural networks: A review, Neural networks 113 (2019) 54–71.
[22] R. Vinuesa, H. Azizpour, I. Leite, M. Balaam, V. Dignum, S. Domisch, A. Felländer, S. D. Langhans, M. Tegmark, F. F. Nerini, The role of artificial intelligence in achieving the sustainable development goals, Nature Communications 11 (2020).
[23] D. Rolnick, P. L. Donti, L. H. Kaack, K. Kochanski, A. Lacoste, K. Sankaran, A. S. Ross, N. Milojevic-Dupont, N. Jaques, A. Waldman-Brown, A. S. Luccioni, T. Maharaj, E. D. Sherwin, S. K. Mukkavilli, K. P. Kording, C. P. Gomes, A. Y. Ng, D. Hassabis, J. C. Platt, F. Creutzig, J. Chayes, Y. Bengio, Tackling climate change with machine learning, ACM Comput. Surv. 55 (2022).
[24] S. Naumann, M. Dick, E. Kern, T. Johann, The greensoft model: A reference model for green and sustainable software and its engineering, Sustainable Computing: Informatics and Systems 1 (2011) 294–304.
[25] R. Schwartz, J. Dodge, N. A. Smith, O. Etzioni, Green AI, Communications of the ACM 63 (2020) 54–63.
[26] A. Guldner, J. Murach, Measuring and assessing the resource and energy efficiency of artificial intelligence of things devices and algorithms, in: V. Wohlgemuth, S. Naumann, G. Behrens, H.-K. Arndt, M. Höb (Eds.), Advances and New Trends in Environmental Informatics, Springer International Publishing, Cham, 2022, pp. 185–199.
[27] C. Sun, V. Puig, G. Cembrano, Real-time control of urban water cycle under cyber-physical systems framework, Water 12 (2020) 406.
[28] A. Niknam, H. K. Zare, H. Hosseininasab, A. Mostafaeipour, M. Herrera, A critical review of short-term water demand forecasting tools—what method should i use?, Sustainability 14 (2022) 5412.
[29] P. Conejos Fuertes, F. Martínez Alzamora, M. Hervás Carot, J. Alonso Campos, Building and exploiting a digital twin for the management of drinking water distribution networks, Urban Water Journal 17 (2020) 704–713.
[30] H. M. Ramos, M. C. Morani, A. Carravetta, O. Fecarrotta, K. Adeyeye, P. A. López-Jiménez, M. Pérez-Sánchez, New challenges towards smart systems' efficiency by digital twin in water distribution networks, Water 14 (2022) 1304.
[31] E. Torfs, N. Nicolaï, S. Daneshgar, J. B. Copp, H. Haimi, D. Ikumi, B. Johnson, B. B. Plosz, S. Snowling, L. R. Townley, et al., The transition of wrrf models to digital twin applications, Water Science and Technology 85 (2022) 2840–2853.
[32] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, D. Dennison, Hidden technical debt in machine learning systems, Advances in neural information processing systems 28 (2015).
[33] D. Kreuzberger, N. Kühl, S. Hirschl, Machine learning operations (mlops): Overview, definition, and architecture, arXiv preprint arXiv:2205.02302 (2022).
[34] R. Wirth, J. Hipp, Crisp-dm: Towards a standard process model for data mining, in: Proceedings of the 4th international conference on the practical applications of knowledge

discovery and data mining, volume 1, Manchester, 2000, pp. 29–39.
[35] W. L. Martinez, A. R. Martinez, J. Solka, Exploratory data analysis with MATLAB, Crc Press, 2017.
[36] K. Sahoo, A. K. Samal, J. Pramanik, S. K. Pani, Exploratory data analysis using python, International Journal of Innovative Technology and Exploring Engineering (IJITEE) 8 (2019) 4727–4735.
[37] I. S. Msiza, F. V. Nelwamondo, T. Marwala, Artificial neural networks and support vector machines for water demand time series forecasting, in: 2007 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2007, pp. 638–643.