

A systematic description of twinning levels of measured data and models in digital twins

Svenja Nicole Schulte[✉], Carl-Philipp Grunenwald[✉], Philipp Schulze[✉], Tobias Breiten[✉] and Rainer Stark[✉]

Technische Universität Berlin, Germany

✉ svenja.schulte@tu-berlin.de

ABSTRACT: Digital Twins are digital representations of products or product-service systems comprising a Digital Master, which consists of product description models, and a Digital Shadow, which encompasses data collected throughout the product's life cycle. To create a Digital Twin, the Digital Master and Digital Shadow must be interlinked. The Digital Master, Digital Shadow, and thus their twinning can vary in complexity and analytical capabilities. This paper introduces a systematic description of six twinning levels ranging from simple data exchange based on generic models to more complex forms targeting model parameter and Digital Twin goal optimization. The example of a valve is used for illustration. The presented description aids in understanding the potential of Digital Twins and serves as a guide to select appropriate twinning levels based on specific product requirements and use cases.

KEYWORDS: digital twin, coupling level, complexity, functional modelling, product modelling / models

1. Introduction

Digital Twins (DTs) offer transformative potential by providing valuable data and insights throughout the entire product lifecycle, enhancing transparency and collaboration among stakeholders. This information can, for example, enhance product development and design. However, engineers face significant challenges in their development, implementation, and utilization. One primary issue lies in the lack of structured, systematic methodologies for DT design, stemming from a heterogeneous understanding of DTs and their development both across industry and research. This heterogeneity is further amplified by the wide variety of DT applications, which differ in their functions, capabilities, contexts and, consequently, their complexity. As a result, the models, data, and twinning mechanisms incorporated in DTs vary widely depending on the specific application. Due to this variability, engineers often struggle with determining the appropriate depth of integration between DT components.

To support engineers in designing and developing DTs effectively, a systematic description characterizing DTs based on their twinning complexity is essential. This can serve as a foundation for general, universal DT design methodologies, independent of specific use cases. By categorizing a DT into a distinct level of twinning complexity, engineers can better estimate the required development scope for DT implementation and select appropriate strategies tailored to specific product and DT needs. Without such a categorization, engineers risk either under-engineering or over-engineering their DT implementations, leading to inefficiencies in cost, performance, or scalability.

This paper proposes a systematic description of six twinning levels, ranging from simple data exchange using generic models to more complex forms targeting optimization of models and DT operation. It combines engineering and mathematical descriptions to characterize DTs by their models, data, and their twinning. The mathematical perspective provides an abstract and versatile description independent of application scenarios, while the engineering perspective provides a technical, functional view useful for integration into DT development processes. By bridging both perspectives, this approach enhances not only

theoretical understanding but also practical applicability. It enhances the understanding of DTs by providing a clearer comprehension of DT twinning mechanisms and offering guidance for selecting appropriate twinning levels to meet specific requirements of the product, DT, and use case during DT development.

2. State of the art

The development and deployment of DTs is a multifaceted process that integrates diverse technologies and methodologies. DTs are digital representations of products or product-service systems comprising a Digital Master (DM) (or Digital Prototype (DP) and a Digital Shadow (DS). The DM consists of product description models; the DS encompasses data collected throughout the product's life cycle. The DP relates to simulation models that are used to represent additional aspects of the system as well as its interactions with additional external systems or environments, for example, additional load cases or product and system functions. To create a DT, the DS must be linked to the DM/DP (Stark & Damerau, 2019; Stark et al., 2017; Stark, et al., 2020), enabling dynamic information flow via the twinning engine (Stark, 2022). However, existing research lacks a systematic approach to differentiate between these twinning mechanisms, making it challenging to establish clear development pathways for various DT applications.

The implementation and utilization of DTs have become prominent research areas. There exist abstract methodologies applicable to various use cases. However, such methodologies often lack detailed guidance on how to execute each step, thereby providing limited methodological guidance for practical application. They also frequently lack granularity in other aspects such as the differentiation between models and data, and their connections (Ariansyah, et al., 2020; Psarommatis & May, 2023; Tao, et al., 2019). Furthermore, relevant standards and guidelines such as DIN EN ISO 11354 (International Organization for Standardization, 2011) and VDI/VDE 2206 (Verein Deutscher Ingenieure, 2021) are broad in scope and, despite their potential to ensure consistency and interoperability, do not adequately address the unique challenges of developing and deploying DTs across diverse industrial sectors. In addition, more targeted approaches to DT development exist. For example, Oulefki et al. (2023) present a structured development process that refers to specific tools and models tailored for creating DTs of buildings. Another example of such targeted approaches was developed by Florescu (2024), which describes the development of a DT of a specific experimental Flexible Manufacturing System. While useful for their respective niche, the frameworks' applicability to broader industrial use cases is limited. Frameworks and models that systematize the characteristics of DTs have the potential to effectively bridge the gap between abstract methodologies and use case-specific approaches. A variety of such frameworks and models exist (Kim et al., 2020). Stark et al. (2019) present a model of eight dimensions of DTs which was later extended by Stark and Schulte (2021). This model accounts for critical factors such as update frequency and simulation capability. Furthermore, Stark et al. (2019) identified a set of six critical design elements. Another notable approach to systematize characteristics of DTs is the periodic table on DT capabilities, which includes 61 capabilities that can be allocated to data services, integration, intelligence, UX, management, and trustworthiness (Digital Twin Consortium, n.d.). Furthermore, DT capabilities are often classified by the intelligence of the system, including the levels "describe", "predict", "prescribe", and "actualize" (Walton et al., 2024).

Additionally, approaches to mathematically describe DTs exist. A notable example is presented by Kapteyn et al. (2021) with subsequent publications building on this approach (e.g., Kapteyn & Willcox, 2022). The authors introduce an abstraction of DTs, and their physical assets based on key quantities, utilizing probabilistic graphical models to enable scalability and updating. However, limitations of the presented abstraction include ensuring the generalizability of the model across diverse applications.

Despite these advancements, the field remains fragmented with limited consensus on methodologies that address the diverse requirements of DTs across industries. This lack of a standardized approach hinders consistency and scalability across different DT implementations, leading to challenges in their efficient development and deployment. This highlights the need for more comprehensive frameworks that integrate theoretical insights with practical considerations, facilitating scalable and efficient DT development and deployment while optimizing design trade-offs between benefits and costs for specific use cases. Therefore, we propose a systematic description of twinning levels between key DT components to support effective DT development and deployment. This framework considers the varying intelligence and capabilities of DTs, from basic data visualization to advanced systems capable of optimization. By clearly differentiating twinning levels, it offers a practical guide for tailoring DT implementations to specific application needs and requirements. This approach is particularly relevant in

the early stages of the V-model (Verein Deutscher Ingenieure, 2021), where requirements and system architecture are planned.

3. Systematic description of twinning levels

The components of a DT - namely the DM, the DS, and their twinning - can vary in complexity and analytical capabilities. To address the requirements of a specific use case with the DT design, it is essential to identify the appropriate level of interaction between the DS and DM and thus the complexity of the twinning during DT development. “Twinning” refers to the degree and nature of the interaction between the DM/DP and DS. We propose the differentiation of six twinning levels between the DM/DP and the DS, supporting a structured and scalable approach to DT design and implementation. Depending on the level, a differentiation between a generic or instance specific DM/DP is made. Different DSs are defined depending on whether the data is unprocessed, processed, or generated synthetically. This paper focuses on individual DTs; the concept of interconnected or aggregated DTs (e.g., for product classes), however, is not addressed in the current scope. For the description of these twinning levels, the following assumptions are made:

1. Each component of the DM and DP can be represented by a function.
2. Each function is characterized by variables, parameters, and constants.
3. The functions of the DP do not necessarily coincide, intersect, or relate to the functions, properties, and characteristics contained in the DM. The DP typically depends on less or specifically derived parameters as some of the functions, properties, and characteristics have been substituted by knowledge extracted from the generic load cases associated to a DM. Some functions, properties, and characteristics need specialization or precision due to the simulation model associated to the DP.
4. Both the DM and the DS can exist in either an uninstantiated or instantiated form.

The twinning levels can be grouped into (i) monitoring of behavior (level 0 and 1), (ii) planned adaptation of behavior (level 2 and 3), and (iii) independent optimization of behavior (level 4 and 5) in order to take into account, the increasing complexity of the connection between the DM/DP and DS. In the following, each twinning level is described from both an engineering and a mathematical perspective. Additionally, a practical example of a DT of a valve is used to further illustrate the individual levels. Each of the following sections describes a level based on its key elements (DM/DP, DS, and twinning).

3.1. Monitoring of behavior

The first two twinning levels characterize DTs that are used to detect, record, or monitor specific product or system behavior. All models are assumed to be static and are not modified over time. Also, they are generic and not specific to an instance. Figure 3 provides an overview of twinning levels 0 and 1.

Level 0 - detached

The detached level can be used to record, analyze, and visualize raw product data without advanced processing or interaction with a DM.

Digital Shadow: At level 0, a DS exists that comprises measured data from the real object, which may be derived from either a single specific object or multiple instances. No data processing or analytics are performed on this data, i.e., the DS remains unprocessed (preDS). Mathematically, the DS is described by a collection of n -tuples of the form $(q_{1,i}, \dots, q_{n,i})$ for $i = 1, \dots, d$.

Digital Master and Digital Prototype: A generic DM (genDM) exists consisting of models pertaining to the product or system. Additionally, a related generic DP (genDP) can exist. The models are initialized from the model repository, which includes all product description models. Mathematically, a genDM/genDP can be considered as a subset of functions $f_i: \mathcal{Z}_i \rightarrow \mathcal{Y}_i$ from the model repository. They vary from general descriptions by a partial differential equation over computer aided design (CAD), e.g., a function mapping a set of three-dimensional coordinates to a 3D model for visual illustration, and finite element (FE) models to simplified ordinary or algebraic equations as well as data-driven models generated by artificial neural networks/machine learning. These models include parameters and constants that are unknown/unspecified. From an abstract point of view, each model i can be seen as a function $f_i: \mathcal{Z}_i \rightarrow \mathcal{Y}_i$, which maps an *argument* from \mathcal{Z} to a *value* from \mathcal{Y}_i . Here, a generic argument $\mathcal{Z}_i \ni z$ may itself consist of variables x_{j_1} , constants c_{j_2} , or (model/operation) parameters $p_{j_3}, p_{M,j_4}, p_{O,j_5}$, i.e., $z = (x_1, \dots, x_m, c_1, \dots,$

$c_{m_2}, p_1, \dots, p_{m_3}, p_{M,1}, \dots, p_{M,m_4}, p_{O,1}, \dots, p_{O,m_5}$). The output values can be categorized into explicit measurable and implicit computable values. Figure 1 shows a visual representation of the genDM or genDP comprising functions, parameters, and outputs.

Twinning: There is no twinning between a DS and a DM/DP at this level. Therefore, no DT exists.

Example: In the example of a valve, the preDS is used for a dashboard displaying timeline data of valve openings and closings. The data consists of data series in the form of tuples $(t_1, v_1), \dots, (t_d, v_d)$ where $t_i \in R_{>0}$ represents time points and $v_i \in \{0,1\}$ describes the valve state (open vs closed).

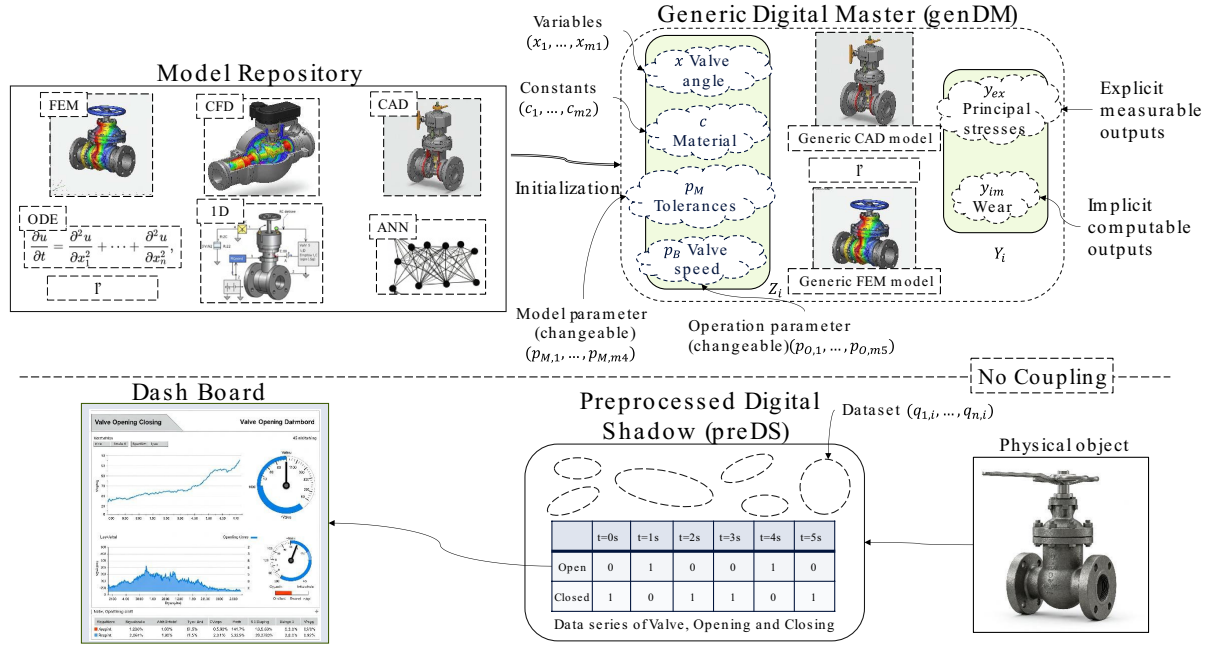


Figure 1. Schema depicting a model repository comprising different models and a preprocessed Digital Shadow in level 0

Level 1 - generic

At the generic level, a DT is created which enables basic simulation, monitoring, and data analysis, providing generalized insights about product behavior, without requiring instance-specific data.

Digital Shadow: At level 1, a preDS exists, same as at level 0. However, at this level, a postprocessed DS (postDS) is also created through processing steps - such as data cleaning, sorting, and clustering - which can represent data from either a single object or multiple objects. Tuples $(q_{1,i}, \dots, q_{n,i})$ collected at level 0 might be sorted by regrouping the data into tuples of smaller size, e.g., $(q_{1,i}, q_{3,i}, \dots, q_{n,i})$ and $(q_{2,i}, q_{4,i}, \dots, q_{n-1,i})$. Alternatively, tuples can be rearranged into vectors $(q_{1,1} \dots q_{1,d})^T, \dots, (q_{n,1} \dots q_{n,d})^T$ or matrices. A first data analysis can be performed, for example by statistical reduction and compression techniques such as singular value decompositions or principal component analysis.

Digital Master and Digital Prototype: The DM and DP at this level are the same as at level 0.

Twinning: Twinning at this level is created between the postDS and genDM or between postDS and genDP, thereby creating a DT. No instance-related twinning is done at this level, meaning not every variable, parameter, or constant in the genDM/genDP has a corresponding equivalent in the postDS. The processed tuples of the postDS $(q_{1,i}, \dots, q_{n,i})$ can be linked to the genDM/genDP by individual grouping of each $q_{k,i}$ into either the set of arguments \mathcal{Z}_ℓ or into the set of values \mathcal{Y}_ℓ .

Example: In the example of a valve, the DT at this level includes the opening and closing timeline linked to a generic FEM model of the valve. A function $f: \mathcal{Z} \rightarrow \mathcal{Y}$ can be defined as a mapping from discrete time instances (e.g., the points available in the timeline) to the visualization of principal stresses. In other words, the input space is $\mathcal{Z} = \{t_1, \dots, t_d\}$ and the output space \mathcal{Y} is a set of 3D pictures of the valve (e.g., open and closed). Alternatively, a continuous input space can be used, e.g., $\mathcal{Z} = R_{>0}$, allowing to evaluate the mapping between two data points t_i, t_{i+1} . This would, however, require some form of interpolation of the values given by $f(t_i), f(t_{i+1})$ which also includes a valve that is in the process of opening or closing, respectively. A schema depicting the twinning is illustrated in Figure 2.

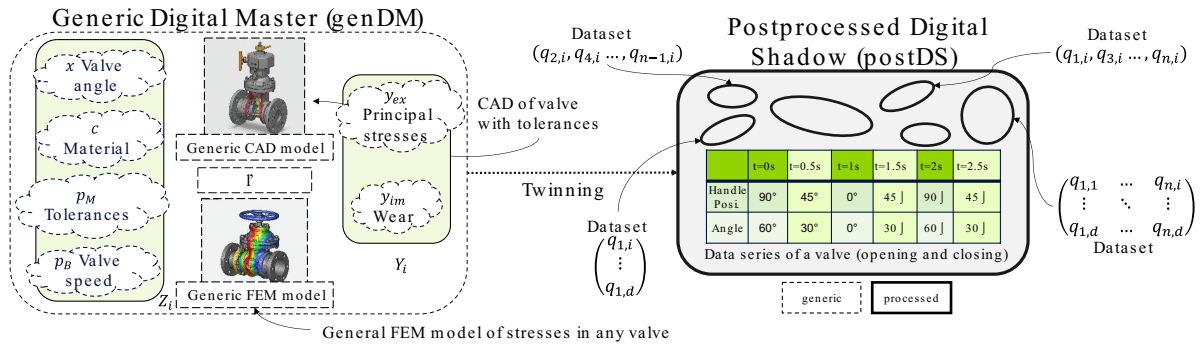


Figure 2. Schema representing a generic digital master/prototype and a postprocessed digital shadow as introduced in level 1

Figure 3 visualizes the arrangement of the DT elements at twinning levels 0 and 1.

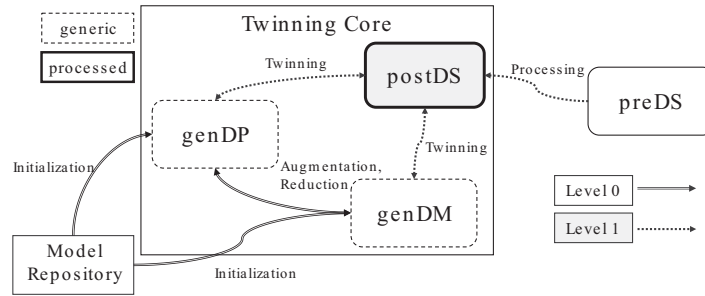


Figure 3. Overview of twinning levels 0 and 1 describing digital twins used for behavior monitoring

3.2. Planned adaptation of behavior

The twinning levels 2 and 3 aim to describe the twinning structure needed to plan and adapt the behavior of the product or system by introducing an instantiation mechanism and synthetic data generation.

Level 2 - instance

At the instance level, the DT is enhanced with instance-specific data and models, allowing for a more precise representation of the actual product or system, including its unique conditions, loads, and state.

Digital Shadow: The DS at level 2 is identical to level 1 but now incorporates instance-specific information in the preDS and postDS. Instance-specific constants and parameters (e.g., geometries, material constants) extend the existing data tuples contained in the postDS.

Digital Master and Digital Prototype: At this level, the DM is instantiated (instDM) by aligning the parameters and constants of the generic model with the specific product instance; thereby, the instDM consists of a reduced argument set of the DM. An instantiated DP (instDP) can exist. For example, tolerances included in the model are replaced with as-built measurements. Mathematically, the instantiation of the DM/DP replaces remaining generic constants c_j and unknown parameters by instance-specific values. As a result, the set of arguments \mathcal{Z}_i of the individual functions \mathcal{Z}_i is reduced to a set of arguments $\tilde{\mathcal{Z}}_i$ comprised of elements of the form $\tilde{\mathcal{Z}} = (x_1, \dots, x_{m_1}, p_{M,1}, \dots, p_{M,m_2}, p_{O,1}, \dots, p_{O,m_3})$. A visualization of the differences between a genDM and an instDM or a genDP and an instDP is provided in Figure 4.

Twinning: At this level, the postDS and instDM or the postDS and instDP are connected, meaning that each variable, parameter, or constant in the instDM/instDP has an equivalent counterpart in the postDS. The instance-specific models are executed in reaction to the processed data of the postDS, which provides a continuous set of variables for the analysis. This twinning can operate in real-time, at set time intervals, or on an event-based basis, as needed. The extension of the data tuples $(q_{1,i}, \dots, q_{n,i})$ from levels 0 and 1 now yields enlarged data tuples $(\bar{q}_{1,i}, \dots, \bar{q}_{n,i})$ which are consistent with elements $(y_{1,i}, \dots, y_{n,i})$ from the argument set \mathcal{Y}_i . The instance-specific functions $f_i: \tilde{\mathcal{Z}}_i \rightarrow \mathcal{Y}_i$ allow for simulation in reaction to the data collected in the postDS.

Example: In the example of a valve, the opening and closing timeline are linked to an instantiated CAD and FEM model of the valve as well as to a specific 1D model of the flow inside the valve which are instantiated to match the exact valve, representing the aging component. In this context, the functions $f_i: \tilde{\mathcal{Z}}_i \rightarrow \mathcal{Y}_i$ can be understood as mappings of the real geometry and materials to the CFD model or the valve openings and closings over time to its outflow (via the solution of the FEM, control, or CFD model).

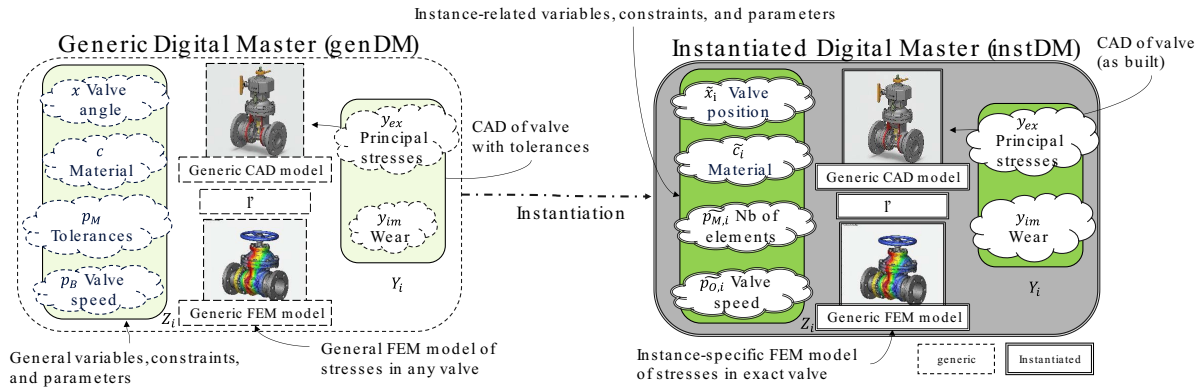


Figure 4. Schema representing the differences between a generic digital master/prototype and an instantiated digital master/prototype as introduced in level 2

Level 3 - extended

At the extended level, the DT incorporates synthetic data generation. This data provides deeper insights into the system's interactions within its context and enables the simulation of scenarios and prediction of behaviors under varying conditions, thereby enhancing proactive decision-making and risk analysis.

Digital Shadow: In addition to the preDS and postDS from level 2, synthetic data sets are generated based on virtual scenarios depicted by instDPs. Simulations based on the instance-specific functions of the instDP $f_i: \tilde{Z}_i \rightarrow \mathcal{Y}_i$ can be used to generate additional data tuples $(y_{1,i}, \dots, y_{n,i})$ for possible enrichment of the available tuples $(\bar{q}_{1,l}, \dots, \bar{q}_{n,l})$. The sum of all synthetic data sets is referred to as the synthetic DS (synDS). Additionally, comparisons between $(y_{1,i}, \dots, y_{n,i})$ and $(\bar{q}_{1,l}, \dots, \bar{q}_{n,l})$ can be drawn in order to assess the prediction and simulation quality of both the instDP and instDM but also the measurement quality of the data in the postDS.

Digital Master and Digital Prototype: The DM and DP at this level remain the same as in level 2.

Twining: As in level 2, the postDS and instDM/instDP are interlinked, meaning each variable, parameter, or constant in the instDM/instDP has an equivalent in the postDS.

Example: A synthetic scenario describing possible pressure peaks in the valve system is generated based on variations of the CFD model from the instDP. The synthetic scenario is compared and evaluated based on postDS data, allowing for analysis of projected versus actual pressure values. Mathematically, such a synthetic scenario is generated by varying the variables x_{j_i} from the argument set of the instance-specific functions $f_i: \tilde{Z}_i \rightarrow \mathcal{Y}_i$ and evaluating the mappings f_i . The resulting values $(y_{1,i}, \dots, y_{n,i})$ can be compared with $(\bar{q}_{1,l}, \dots, \bar{q}_{n,l})$ by means of individual metrics that allow to measure the “closeness” of these data sets,

e.g., by computing
$$\left\| \begin{pmatrix} \bar{q}_{1,1} & \cdots & \bar{q}_{n,1} \\ \vdots & \ddots & \vdots \\ \bar{q}_{1,d} & \cdots & \bar{q}_{n,d} \end{pmatrix} - \begin{pmatrix} y_{1,1} & \cdots & y_{n,1} \\ \vdots & \ddots & \vdots \\ y_{1,d} & \cdots & y_{n,d} \end{pmatrix} \right\|$$
 for an appropriate norm $\|\cdot\|$.

Figure 5 shows the generation of a synthetic scenario summarized as the synDS based on the variation of a CFD model as well as control behavior model in the instDP.

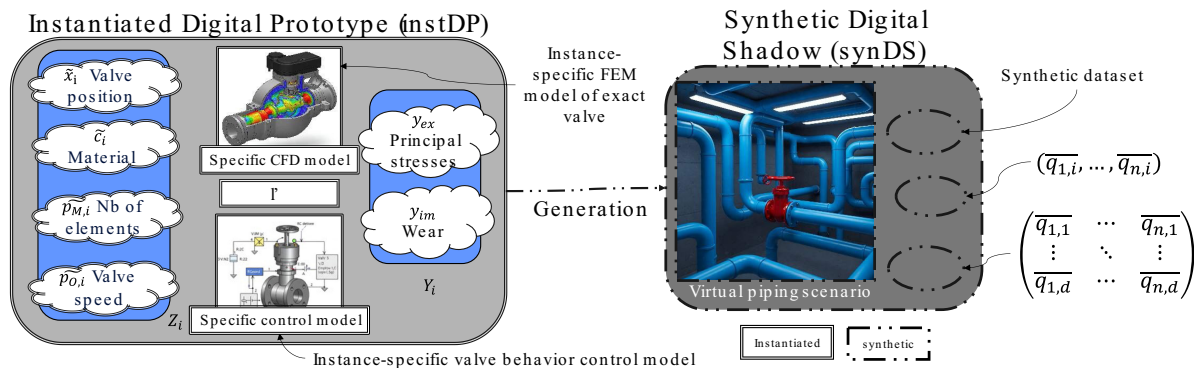


Figure 5. Schema representing the synthetic digital shadow and its generation based on a variation of an instantiated digital prototype

A visual representation summarizing the interdependencies of the newly introduced DT elements in the form of instDM, instDP, and synDS at twinning levels 3 and 4 is provided in Figure 6.

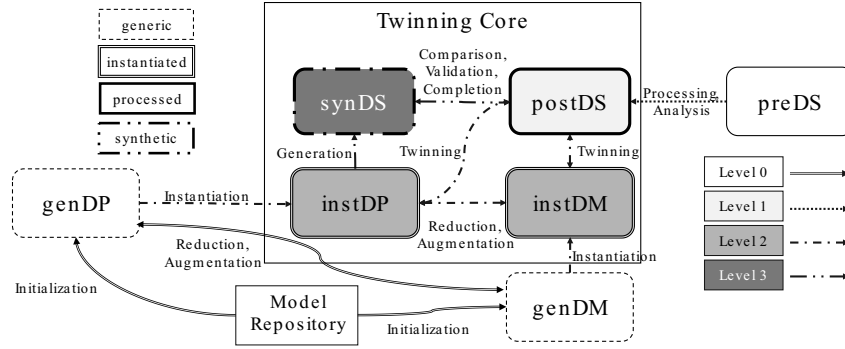


Figure 6. Overview of twinning levels 2 and 3 describing digital twins used for planned behavior adaptation

3.3. Optimization of behavior

The twinning levels 4 and 5 include the active optimization of models and operation of the DT to enable increasingly automatic, even autonomous, properties.

Level 4 - model-optimized

At the model-optimized level, the DT evolves by optimizing model parameters, continuously enhancing simulation accuracy and currency, and predictive capabilities of the DT based on dynamic data from the real-world. The update frequency adjusts to system degradation or changes in operational conditions.

Digital Shadow: The DS at this level remains the same as in level 3.

Digital Master and Digital Prototype: At level 4, the instDM/instDP is optimized by adapting model parameters. The optimization problem requires a suitable cost functional $\mathcal{J}_{\text{misfit}}$, which generally depends on the set of model parameters $p_{M,j}$ and comprises individual simulation versus measurement data misfit terms which should be minimized. A typical optimization problem could be of the form:

$$\min_{p_{M,j} \in \mathcal{P}_a} \mathcal{J}_{\text{misfit}} := \sum_{i=1}^d \|(\bar{q}_{1,i} \cdots \bar{q}_{n,i}) - (y_{1,i} \cdots y_{n,i})\|^2 \text{ s.t. } f_i(x_{i,k}, p_{M,j}, p_{O,i,k}) = (y_{1,i}, \dots, y_{n,i}) \quad (1)$$

where \mathcal{P}_{ad} denotes a set of *admissible* parameters, considering that not all parameter values may be realizable in practice (e.g. due to physical constraints such as maximal geometries, pressure bounds, energy limitations). The goal of this minimization is to find model parameters which reduce deviations between the instDM/instDP on the one side and the postDS on the other side.

Twinning: At this level, the postDS and instDM or the postDS and instDP are interlinked, same as in level 3; however, this twinning is continuously refined by the aforementioned model parameter optimization. In particular, the model parameters $p_{M,j}$ are now implicitly linked to the postDS via the optimization problem.

Example: Over time, model parameters (such as the number of CFD elements, material properties, and friction coefficients) are continuously adapted to better align with the real valve characteristics recorded in the preDS. This ensures that the models accurately reflect the valve's current performance and wear state, supporting predictive maintenance. A neuronal network used to predict the future wear and tear of the valve could be retrained using DS data to better match the real valve.

Level 5 - operation-optimized

At the operation-optimized level, the DT is optimized to reflect the system's changing behavior (as in level 4), as well as proactively optimized to enhance system operations. The aim is to maximize overall DT objectives, such as minimizing costs, maximizing profit, and extending longevity through continuous operational adjustments. The optimized operation parameters correspond directly or indirectly to sets of actuator, control, or state values that influence system behavior.

Digital Shadow: The DS at this level remains the same as in level 3 and level 4.

Digital Master and Digital Prototype: For the (changeable) operation parameters $p_{O,j}$ of the instDM/instDP, a second (or multiple further) cost functional \mathcal{J}_{DTgoal} is defined. Typically, \mathcal{J}_{DTgoal} comprises of different, generally concurrent, sub cost functionals which themselves model costs, profit, or product longevity. For example, one may consider a minimization of a functional $\mathcal{J}_{DTgoal} = \mathcal{J}_{costs} - \mathcal{J}_{profit} - \mathcal{J}_{longevity}$. Since the optimization of the operation parameters $p_{O,j}$ will generally result in a change of the model functions contained in the instDM/instDP, it is crucial that this will not cause undesirable diminishment of their prediction and simulation capabilities. Hence, these operation parameter optimizations should typically either go hand in hand with a subsequent model parameter adaptation/audit or directly be conducted via a suitable bilevel optimization problem of the abstract form

$$\min_{p_{O,j_1}} \mathcal{J}_{DTgoal}(p_{O,j_1}, p_{M,j_2}) \text{ s.t. } p_{M,j_2} \in \arg \min_{r_{j_2}} \mathcal{J}_{misfit}(p_{O,j_1}, r_{j_2}) \quad (2)$$

where the first minimization ensures optimization of the overall DT objects, while the second minimization guarantees the DT to still accurately reflect the underlying measurements and processes.

Twinning: At this level, the twinning between the postDS and instDM/instDP is tailored through both model parameter optimization and operation parameter optimization. Figure 7 shows a visual representation.

Example: The model parameters, such as numbers of elements, are continuously adapted to match the real valve characteristics. In addition, the behavior of the valve is continuously updated during operation to maximize the remaining lifespan.

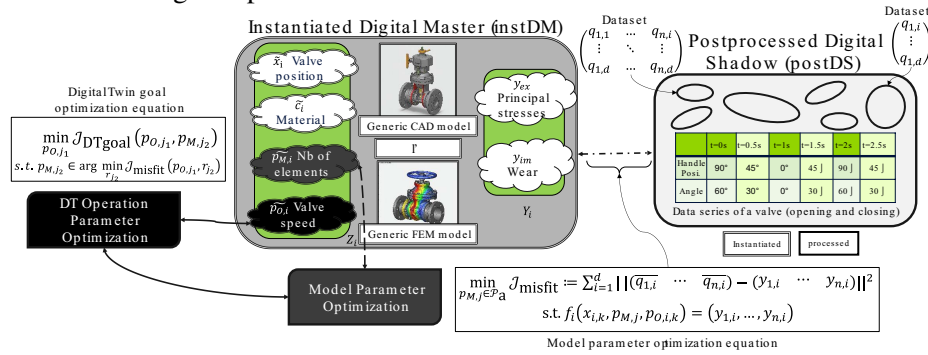


Figure 7. Schema depicting the relationship between digital twin goal optimization and model parameter optimization as introduced in level 5

Figure 8 summarizes all DT elements and their arrangement into the proposed six twinning levels.

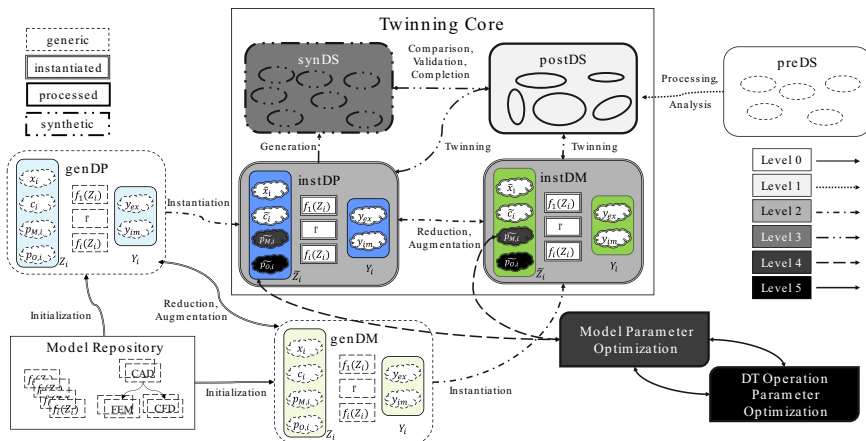


Figure 8. Summary of all Digital Twin elements of the six twinning levels

4. Discussion and Outlook

The presented systematic description of twinning levels helps to address the challenges outlined in the state of the art, including the lack of detailed methodological guidance and the limited applicability of existing development processes to diverse contexts. The description provides guidance for the understanding and development of DTs at varying levels of complexity and integration, thereby making it easier to tailor DT implementations to specific needs. Additionally, the described twinning levels provide a scalable and adaptable framework that accommodates the unique requirements of different industries, ensuring that the model is both comprehensive and versatile.

The proposed characterization of twinning levels is based on the assumption that any and all models can be represented as a function mapping specific parameter sets to output values. While this enables the consideration of both deterministic and probabilistic functions, not all models might fit this criterion. Especially engineering models that are often defined for a specific use case during the development process of products or systems might not be suited to be reused, repurposed, or even executed in the context of a DT. Metamodels putting into relations different, possibly unquantifiable or inexecutable aspects of a product or system (such as requirements or technical drawings) might not fit the proposed framework but could still generate great insights in the context of DTs at levels 0 and 1, aiming to generate insights about unknown correlations. Estimating the usefulness, aptitude, and compatibility of models and data remains particularly challenging and is highly dependent on the specific application. Another limitation of the presented systematic description is that mechanisms for aggregating multiple DTs into cohesive systems are not yet included. Aggregating the data in the form of multiple DSs generated by multiple different instances of one identical product or system could lead to deeper insights but would require additional aggregation, analysis, and twinning mechanisms. In addition, data sharing across different DTs of different systems might also require further investigation. In complex environments (such as manufacturing plants) multiple DTs must interact and synchronize. While the current framework provides a solid foundation for individual DTs, extending it to environments with interconnected DTs could unlock new opportunities. Future research should address data harmonization, state synchronization, and interdependency management to enable seamless integration of multiple DTs. Therefore, applying the systematic description to DT networks is a promising direction. Interconnected DTs introduce challenges like ensuring interoperability, managing communication latency, and coordinating decisions across distributed nodes. Addressing these will enhance the scalability and applicability of DTs to large-scale ecosystems like smart cities or industrial networks. By addressing these areas, the framework can help to support complex real-world applications, advancing its theoretical foundation and practical utility across diverse industries.

5. Summary

This paper presents a systematic description of six twinning levels. The differentiation of twinning levels based on the complexity of the connected DM/DP and DS and thus their linkage enhances the understanding of DTs. Therefore, this approach provides guidance for DT development at varying levels of complexity and integration of the DM/DP and DS and thereby effectively bridges the gap between abstract methodologies and practical application. It provides a structured way to approach DT design, ensuring engineers can make informed decisions at each step of development. However, the underlying assumptions introduce certain limitations. Future research should focus on refining aggregation mechanisms for multiple DTs and extending the framework to DT networks. These advancements will further enhance the systematic description's utility in complex and dynamic application domains.

References

- Ariansyah, D., del Amo, I. F., Erkoyuncu, J. A., Agha, M., Bulka, D., De La Puente, J., . . . Penver, S. (2020). Digital Twin Development: A Step by Step Guideline. *SSRN Electronic Journal*. doi:<https://doi.org/10.2139/ssrn.3717726>
- Digital Twin Consortium. (n.d.). *Digital Twin Capabilities Periodic Table*. Retrieved from Digital Twin Consortium: <https://www.digitaltwinconsortium.org/initiatives/capabilities-periodic-table/>
- Florescu, A. (2024). Digital Twin for Flexible Manufacturing Systems and Optimization Through Simulation: A Case Study. *Machines*, 12(785). doi:<https://doi.org/10.3390/machines12110785>

- International Organization for Standardization. (2011). Advanced automation technologies and their applications — Requirements for establishing manufacturing enterprise process interoperability (ISO 11354-1:2011). ISO. Retrieved from <https://www.iso.org/standard/50417.html>
- Kapteyn, M. G., & Willcox, K. E. (2022). Design of Digital Twin Sensing Strategies Via Predictive Modeling and Interpretable Machine Learning. *Journal of Mechanical Design*, 144(9). doi:<https://doi.org/10.1115/1.4054907>
- Kapteyn, M. G., Pretorius, J. V., & Willcox, K. E. (2021). A probabilistic graphical model foundation for enabling predictive digital twins at scale. *Nature Computational Science*, 1, 337–347. doi:<https://doi.org/10.1038/s43588-021-00069-0>
- Kim, Y.-W., Yoo, S., Lee, H., Kim, S. H., & Han, S. (2020). *Characterization of Digital Twin*. Retrieved from <https://www.researchgate.net/publication/353930234>
- Oulefki, A., Amira, A., Kurugollu, F., & Alshoweky, M. (2023). Twinning Buildings: A Methodological Framework for Design and Implementation using Home Assistant Technology. *Proc. of the 4th International Conference on Electrical, Communication and Computer Engineering (ICECCE)*. Dubai: IEEE. doi:<https://doi.org/10.1109/ICECCE61019.2023.10442609>
- Psarommatis, F., & May, G. (2023). A literature review and design methodology for digital twins in the era of zero defect manufacturing. *International Journal of Production Research*, 61(16), 5723–5743. doi:<https://doi.org/10.1080/00207543.2022.2101960>
- Stark, R. (2022). *Virtual Product Creation in Industry*. Berlin, Heidelberg: Springer. doi:<https://doi.org/10.1007/978-3-662-64301-3>
- Stark, R., & Damerau, T. (2019). Digital Twin. In S. Chatti, & T. Tolio (Eds.), *CIRP Encyclopedia of Production Engineering*. Springer. doi:https://doi.org/10.1007/978-3-642-35950-7_16870-1
- Stark, R., & Schulte, S. N. (2021). *Determination of Digital Twin Maturity Levels Within Value Creation Networks*. NAFEMS world congress. Retrieved from https://www.nafems.org/publications/resource_center/nwc21-283/
- Stark, R., Freseemann, C., & Lindow, K. (2019). Development and operation of Digital Twins for technical systems. *CIRP Annals - Manufacturing Technology*, 68, pp. 129–132. doi:<https://doi.org/10.1016/j.cirp.2019.04.024>
- Stark, R., Kind, S., & Neumeyer, S. (2017). Innovations in digital modelling for next generation manufacturing system design. *CIRP Annals - Manufacturing Technology*, 66, pp. 169–172. doi:<https://doi.org/10.1016/j.cirp.2017.04.045>
- Stark, R., Thoben, K.-D., Gerhard, D., Hick, H., Kirchner, E., Anderl, R., . . . Göckel, N. (2020). WiGeP-Positionspapier - Digitaler Zwilling. *ZWF Zeitschrift fuer Wirtschaftlichen Fabrikbetrieb*, 115, 47–50. doi:<https://doi.org/10.3139/104.112311>
- Tao, F., Sui, F., Liu, A., Qi, Q., Zhang, M., Song, B., . . . Nee, A. Y. (2019). Digital twin-driven product design framework. *International Journal of Production Research*, 57(12), 3935–3953. doi:<https://doi.org/10.1080/00207543.2018.1443229>
- Verein Deutscher Ingenieure. (2021). Entwicklung mechatronischer und cyber-physischer Systeme (VDI/VDE 2206). (VDI, Ed.) Retrieved from <https://www.vdi.de/richtlinien/programme-zu-vdi-richtlinien/vdi-2206>
- Walton, R. B., Ciarallo, F. W., & Champagne, L. E. (2024). A unified digital twin approach incorporating virtual, physical, and prescriptive analytical components to support adaptive real-time decision-making. *Computers & Industrial Engineering*, 193, 1–14. doi:<https://doi.org/10.1016/j.cie.2024.110241>