Architecture Options to Orchestrate Digital Twins in an Industrial Metaverse for the Predictive Production with AI Methods

Evaluation of Options and Proposal for a High-Level Roadmap

Bernd Lüdemann-Ravit
Institute for Production and Informatics
University of Applied Sciences,
Kempten, Germany
e-mail: bernd.luedemann-ravit@hs-kempten.de

Frieder Heieck
Institute for Production and Informatics
University of Applied Sciences,
Kempten, Germany
e-mail: frieder.heieck@hs-kempten.de

***Abstract*— The Industrial Metaverse (IM) is an upcoming topic for companies and offers new possibilities to digitalize and optimize their business processes together with AI capabilities. In the production domain, the Industrial Metaverse is a step towards the vision of predicting factory behavior for optimization purposes. A central challenge is a complete factory model necessary as the base to predict its behavior. Therefore, the IM approach is promising to build and contain this model out of available single Digital Twins of factory parts. Consequently, an IT target landscape is required to build an Industrial Metaverse for Digital Twins. This paper evaluates different design pattern options for an industrial IT architecture reference implementation of an IM that companies can use in current IT landscapes. It also proposes a high-level roadmap towards the proposed target IT architecture of an IM.**

Keywords-Industrial Metaverse; Digital Twin; Virtual Commissioning; PLC; Digital Factory.

I. INTRODUCTION

The term Metaverse is widely used at the moment to describe a virtual environment with the ability for immersive collaboration among participants using technologies such as virtual, mixed or augmented reality [1]. In the industrial context, a similar term emerged: the Industrial Metaverse (IM) [2]–[4]. In the context of manufacturing, the common understanding of this term is a virtual representation of a factory (Figure 1), either with a connection to a real factory as a Digital Twin (DT) of the complete factory, or as a model of the planned, not yet realized factory [5].

Figure 1. Example of a photorealistic Digital Twin in NVIDIA Omniverse

Several companies have started projects to gain the benefits of an IM, e.g., project iFactory by the BMW Group in NVIDIA Omniverse [6]. One of the most promising benefits lies in integrating Artificial Intelligence (AI) into IM. The vision is to predict the behavior of production processes inside a virtual representation of the plant so as to prevent unforeseen defects before commencing the real manufacturing process, e.g., path and action planning of Automated Guided Vehicles (AGVs) for intralogistics to avoid jams or deadlocks, or assembly of a rare product variant not previously validated.

After this introduction, this paper focus on use cases for an IM in Section II. Then, the requirements for these use cases are defined in Section III and related state of the art work is presented in Section IV. In Section V the single components of the architecture options are defined, The architecture options itself are presented in Section VI. This paper ends with the roadmap towards the recommended target architecture in Section VII and sums up central results in Section VIII.

II. USE CASES FOR AN INDUSTRIAL METAVERSE

The following section describes use cases that it was not possible to realize in the past without an IM, or only with extraordinary effort. The listed use cases is not exhaustive, but it demonstrates the high potential that an IM offers.

Firstly, one of the most obvious use cases for an IM is to integrate planning data created by different production planning departments such as logistic, factory or process planning to check the validity, maturity and consistency of production planning scenarios along the product lifecycle [7]. Additionally, many external suppliers contribute to the realization of the factory. Therefore, a new milestone with an OEM approval for all suppliers based on the virtual representation of the complete factory, and not only parts of it than today, can increase overall efficiency of the realization process and boost reliability during the ramp-up phase.

Secondly, IM can be used as a training platform for employees to train processes or maintenance aspects before building the real factory.

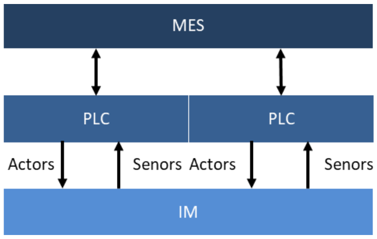


Figure 2: Architecture for a VC for MES

Thirdly, current virtual commissioning (VC) of single programmable logic controllers (PLC) is state of the art for original equipment manufacturers (OEM) in the automotive industry [8]. A VC of several PLCs or superimposed control systems, such as manufacturing execution systems (MES), is a new use case enabled by an IM (Figure 2) to validate control strategies deeply before ramp up.

Fourthly, the most promising use cases that an IM offers originate from its integrated AI capabilities to simulate, predict and optimize the production process before operation, e.g. to avoid undesirable incidents or identify optimized process parameters [9].

Last but not least, IM has an enormous potential to promote the use of AI technologies in real production scenarios in the factory by providing synthetic data to train AI algorithms to identify production problems, such as quality issues, using computer vision systems. In order to improve the accuracy and reliability of AI algorithms, the necessary training data can be generated out of the IM, whereas it is either impossible to obtain from reality or only very sparsely available.

III. REQUIREMENTS

This section describes the main requirements for an IM used to predict production behavior.

A. *Integration into Business Processes*

First, the main requirement for successfully implementing IM use cases so as to achieve its benefits is its integration into the company's business processes. Figure 3 shows a proposal. This can mean either restructuring current business processes, e.g., integrating the VC results into an IM, or implementing new business processes, such as predictive production. Ignorance of integrating new technologies into business processes is a widespread pitfall [10].

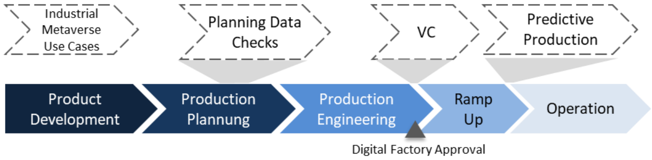


Figure 3: Use Case Integration into Business Processes throughout the Production Lifecyle

B. *Integrating IM into Current IT Landscapes*

The second essential requirement for successfully implementing an IM is its integration into the company's current IT landscape. Due to costs, complexity and time-constraints, a greenfield approach, that is the planning and realization of a complete new factory, is seldom practical. In the current IT landscape holding data for a DT of the factory, the following kinds of authoring systems can contribute to the virtual representation of the complete factory:

- Suites for product data management (PDM) and product lifecycle management (PLM) [11]
- Individual planning systems [12]
- Systems for virtual commissioning (VC) [8]

These IT systems need to provide some kind of interface to transfer data for the virtual representation of the factory into the IM. A new additional process besides the established business processes to build up this representation is generally not an economical solution.

IV. RELATED WORK

DTs for VC to pre-program PLCs before the physical ramp-up of the controlled equipment have already been established for years in the automotive industry [8][13], especially in the body shop. In other areas, there is still a lot of work to be done towards extensively using DT for VC. DTs exist normally for single cells or stations, but there is currently no known VC system for an entire factory, as this would require VC of an MES. These single DTs are normally archived after the physical ramp-up, e.g., on a file system. If the automation equipment needs extending or reorganizing for new products or product variants, the DTs are reused to plan for these changes. In all other cases, the DTs are generally not touched after their physical counterparts have been built. The creation of DTs for a VC is performed using standard systems [8]. Current approaches add automated testing to VC systems [14].

Additionally, there are architecture recommendations for cyber-physical systems (CPS) to support a DT connected to its physical counterpart [15]. None of these are capable of realizing an IM where several people can collaborate on one virtual representation.

There are several architecture proposals for an IM, e.g., those cited in sources [16][17]. One of the most popular systems used in the industry is NVIDIA Omniverse (Figure 4) [18].

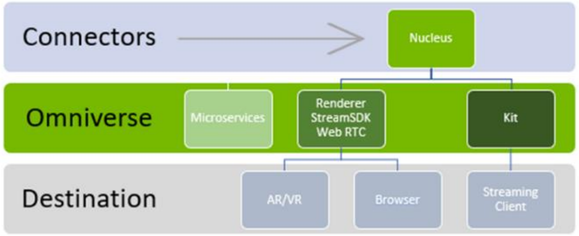
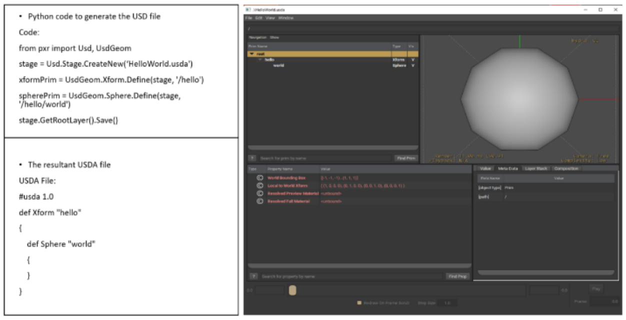


Figure 4: Architecture NVIDIA Omniverse

NVIDIA Omniverse (OV), which was released in 2019, is a collaboration platform for 3D production pipelines that enables multiple partners to work independently on different parts of the scene using a wide range of tools and syncing the changes instantly across all the tools. In order to facilitate this, the assets and scenes are represented using Pixar's Universal Scene Description (USD), as shown in Figure 5 [20], and accessed via OV's database and collaboration engine Omniverse Nucleus. Omniverse also supports material descriptions written in the Material Definition Language (MDL). Along with the developer framework, Omniverse Kit comes pre-packaged with a world-class rendering software and support for physics-based simulation.

The ability to connect existing authoring systems to the Nucleus through an open application programming interface (API) enables easy integration of existing IT systems into the IM. There are already several connectors on the market, e.g., for the VC system Visual Components or a proof-of-concept connector for the VC system CIROS. Through the API connector, a DT from a VC system can be streamed directly to the OV nucleus. The responsibility of the master DT can remain with the VC system. The standardized API connector makes it is possible for DTs created with different systems by multiple vendors to directly interact in a single virtual representation of the factory inside OV, so that there is no danger of a vendor lock between connected DTs.

Additionally, OV offers a large variety of AI algorithms that can be executed efficiently with high performance on NVIDIA hardware. These are ideal prerequisites for implementing AI methods for predictive production.



- Python code to generate the USD file

```
Code:
from pxr import Usd, UsdGeom
stage = Usd.Stage.CreateNew('HelloWorld.usda')
xformPrim = UsdGeom.Xform.Define(stage, '/hello')
spherePrim = UsdGeom.Sphere.Define(stage,
'/hello/world')
stage.GetRootLayer().Save()
```

- The resultant USDA file

```
USDA File:
#usda 1.0
def Xform "hello"
{
   def Sphere "world"
   {
   }
}
```

Figure 5: Python snippet to generated USD files and its rendering

Therefore, we focus our architecture options in the next sections on OV to integrate existing DTs into a single virtual representation of a complete factory.

V. ARCHITECTURE COMPONENTS

The following sections describe different architecture options for integrating interacting DTs into an Industrial Metaverse. Each architecture option orchestrates the same architecture components, as follows:

A. *Programmable Logic Controller (PLC)*

The PLC component interacts with actuators and sensors through a digital or analog input/output interface (I/O). The control logic that controls the sequence of the PLC outputs that are set according to the PLC inputs is implemented inside the PLC in a generally standardized PLC language [19].

B. *Manufacturing Execution System (MES)*

The MES is a control logic that superimposes current PLCs. The essential responsibility of an MES is to represent all orders, products, materials and other resources currently in use inside a factory. The MES provides Key Performance Indicators (KPI) such as Overall Equipment Effectiveness (OEE) to control the behavior of the factory. Modern MESs integrate functionality, such as the control logic for PLCs. The communication between an MES and the shop floor is also performed via digital/analog input/output.

C. *Digital Twin (DT) for Virtual* Commissioning (VC)

The information for single DTs of production equipment, such as a cell or a station, is stored in standard systems for VC [8]. These systems generally contain a geometric 3D model of the equipment. Additionally, the DT contains kinematic axis modeling to define the mechanical behavior of the linear or rotational axis of the equipment. Some systems support even the physics of rigid bodies, such as gravitation or friction. Besides this mechanical model, the objects have an I/O behavior model that communicates with the controller and other sensors/actuators via I/O. An example of a behavior model is a conveyor that starts with a digital input and informs of its current operation status through a digital output.

D. *Industrial Metaverse (IM)*

The IM is a platform holding the virtual 3D-representation of the factory. Additionally, it offers optional services such as AI, photorealistic rendering, or support for animation and simulation [18].

E. *Orchestration Layer (OL)*

The orchestration layer is responsible for synchronizing multiple single DTs, e.g., the overall sequence of processes in the factory. For example, the processes of DT B run after completion of the processes executed by DT A. Another example is synchronizing the flow of material between the single DTs, so that the virtual factory in the IM can manufacture a complete spectrum of digital products.

F. *Automation Equipment (AE)*

This architecture component represents the shop floor equipment that is controlled by a controller, either a PLC or an MES.

VI. ARCHITECTURE OPTIONS

This section defines the design criteria and the architecture options of an IM.

A. *Design Criteria*

The presented architecture options for an IM with the goal of predicting production behavior differ in the following design criteria:

1) Location Control Logic: The control logic of the automation equipment is defined either entirely in the PLC, in a hybrid between PLC and MES, or only in MES.

2) Location DT: The services, e.g., the mechanical or I/O behavior model of the DTs, are located either in the original standard VC system or in the IM itself. VC systems are complex, and the current market leader started development ten to fifteen years ago. Therefore, a complete refactoring of the IT landscape for an integration of DTs into IM takes too long. It makes more sense to integrate available VC systems into to the architecture option.

3) Location Orchestration Layer: The OL is located either inside the IM or as an external component outside the IM. The main part of the OL is a messaging system to synchronize the multiple DTs. For instance, Kafka [20] , MQTT [21] or OPC UA Pub Sub [22] could be possible implementations. In addition, a protocol is also required inside the OL to define the semantics of the individual messages and the corresponding behavior of the interacting DTs.

These design criteria present the following six IM architecture options:

B. IM Architecture Option A

Regarding IM architecture option A., the DTs are located in the VC systems (Figure 7).

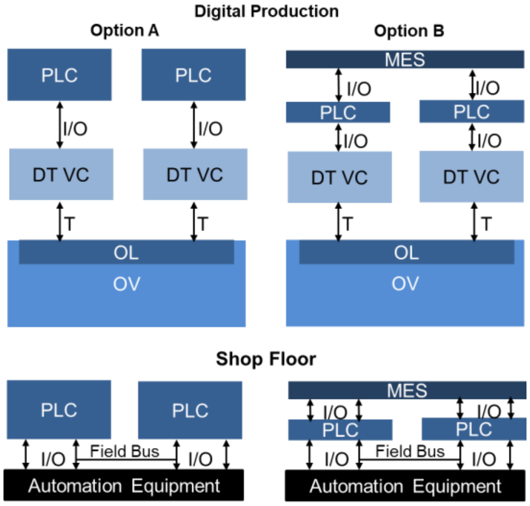


Figure 6: Architecture Options A and B

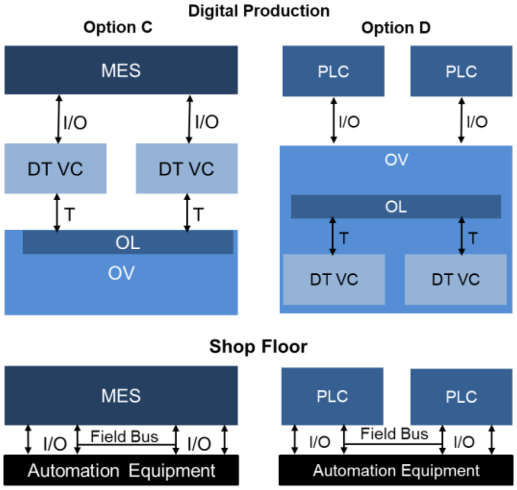


Figure 7: Architecture Options C and D

The Omniverse offers an open API, with which the static geometry of the DT is initially loaded from the VC system to OV through USD. The control logic is located in the PLC, as it is normally implemented on the shop floor. The PLC sends digital and analog output signals to the automation equipment inside the DT via an I/O interface between the PLC and the VC system. It receives digital and analog input signals from the DT about the conditions of the DT objects, such as, e.g., sensors. The VC system calculates the dynamic movements of the DT objects, e.g., linear or rotational axes. The corresponding changes of the objects' positions and orientations, defined, e.g., by a 4x4 transformation matrix T, are streamed from the VC system to OV through the API. OV does not calculate the object's trajectory in architecture option A. The orchestration of the multiple DTs in OV is implemented in the OL, within which the operation sequence and the material flow in OV are synchronized.

C. IM Architecture Option B

IM architecture option B (Figure 6) differs from option A in the way the control logic is implemented. Option A handles the entire control logic inside the PLC, whereas option B splits the control logic into two parts:

1) Control Logic in the MES: This part of the control logic handles the product variants or product orders, e.g., control logic for a specific product feature that other product variants do not have.

2) Control Logic in the PLC: This part handles control logic focusing on the shop floor, such as safety functions or real-time control logic for, e.g., axis.

D. IM Architecture Option C

IM architecture option C (Figure. 7) implements the complete control logic inside the MES. In comparison to option A, option C allows the implementation of control logic in one instance concerning multiple cells, lines or even an entire factory. This option offers much more flexibility to change the control logic in one instance than would be required in multiple PLCs in options A or B. The parts DT, VC and OV are the same as in options A and B.

E. IM Architecture Option D

The main difference in the IM architecture of option D (Figure 7) is the implementation of the DT in OV. In options A-C, the DTs were located outside the OV in multiple VC systems. As the features of OV increase, e.g., Omnigraph, it will become possible in the future to integrate the single DTs of the automation equipment into OV to create a single DT of the factory. Option D abandons all the VC systems because their functionality is integrated into OV. Despite the integration of the single DTs into OV, an OL is still required inside OV for the purpose of synchronization. An individual OL can also be implemented inside or outside the OV using the OV API.

F. IM Architecture Option E and Option F

Both IM architecture options E and option F (Figure 8) are addressed in this paper for the sake of completing the architecture options available with regard to combining design criteria.

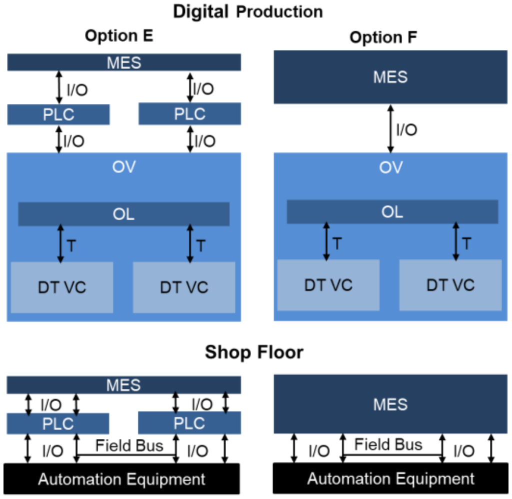


Figure 8: Architecture Options E and F

In options E and F, the DTs are implemented inside OV, similarly to option D. Also, the control logic of option E corresponds to the control logic of option B. Option F corresponds to option C with regard to the control logic.

G. Internal versus External Orchestration Layer

All diagramed options A-F show the OL inside OV. The advantage of an internal OL is the use OV internal functions to synchronize, e.g., the temporal sequence or the material flow among the DTs. The corresponding options A* to F* have an external OL with the same responsibility as their internal equivalents (Figure 9).

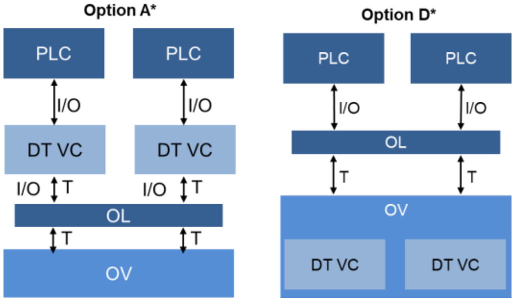


Figure 9: Option A* and D* with an external OL

With an external OL its tasks is getting more transparent. E.g., the OL of options A*-F* have to implement the I/O synchronization between the DTs to keep the transformations T of the 3D-objects correctly along the timeline. Additionally the OL has to know the ids of the DT objects to manage the transformations of the DT objects correctly.

The advantage of implementing an external OL is that it hides completely the OV from the above layers, e.g., the PLCs in option D* (Figure 9). Hence, an external OL offers the possibility to change the OV without a reimplementation of the OL and an adaption of the above layers.

VII. IM ARCHITECTURE ROADMAP

For simplification, we focus in this section on the internal OL options A-F, but the roadmap can easily transferred to the external OL options A*-F*.

Regarding IM architecture options A-F, one can find in most cases the implementation of the control logic as in option A or D, through the sole use of PLCs. The target architecture is implemented in option F with the following motivation: Firstly, because it offers the greatest flexibility in changing the control logic through its implementation in the MES system; secondly, the integration of the single DTs into OV offers the minimal number of systems required for implementing the DT of the entire factory. Practically, this reduces complexity concerning technical and organizational aspects and consequently license and maintenance costs.

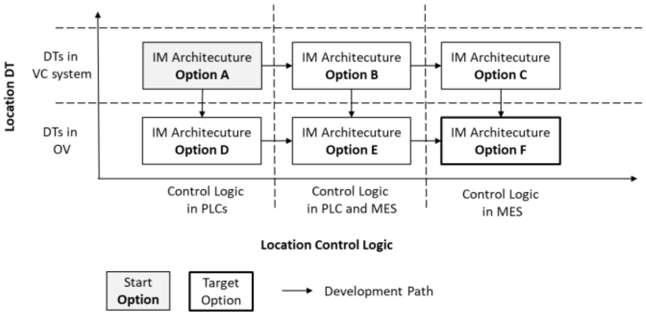


Figure 10: Roadmap for IM architecture

To implement an IM architecture, we propose a roadmap (Figure 10) starting with option A, because the regular control logic from the shop floor is not touched and the existing VC systems need not to be replaced, but only an interface to OV to stream the positions of the DT objects. For the synchronization of the DTs, an implementation of the OL is necessary that can be reused almost completely with some adaptations in options B-F.

The roadmap assumes that with one development path, only one of the design criteria “control logic” or “location of DTs” is changed. At the current stage, there is no preferred development path. Its realization speed and costs is significantly dependent on the single roadmaps of the systems MES and OV, because MES will integrate today’s PLC functions and OV will integrate present DT VC functions.

VIII. CONCLUSION

This paper presents for the first time different architecture options to realize an Industrial Metaverse based on the existing IT landscape. The two main assumptions are, firstly, that the existing control logic implementation found in current shop floors is based on PLCs, and, secondly, that to program these PLCs, current implementations of DTs for VC are based on separate VC systems within current IT landscapes. It was possible to define design criteria and to classify the architecture options into six categories.

Here, we have succeeded in defining a roadmap proposing a step-by-step implementation towards a target architecture. The presented architecture options are the prerequisite of the complete DT of the factory, which is the necessary to use OV internal AI algorithms to optimize and predict the behavior of production scenarios.

REFERENCES

[1] H. Pozniak, "Could engineers work in the metaverse?“, *Engineering & Technology*, vol. 17, no.4, pp. 99–106, May 2022.

[2] J. Lee, P. Kundu, "Integrated cyber-physical systems and industrial metaverse for remote manufacturing“, *Manufacturing Letters*, vol. 34, pp. 12–15, Oct. 2022.

[3] X. Yao et. al., "Enhancing wisdom manufacturing as industrial metaverse for industry and society 5.0“, *Journal Intelligent Manufacturing*, Nov. 2022.

[4] Z. Zheng *et.al.*, "Industrial Metaverse: Connotation, Features, Technologies, Applications and Challenges“, in *Methods and Applications for Modeling and Simulation of Complex Systems*, in Communications in Computer and Information Science, vol. 1712, Springer Nature Singapore, pp. 239–263. 2022.

[5] W. Kritzinger, M. Karner, G. Traar, J. Henjes, W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification“, *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018.

[6] N. Kshetri, "The Economics of the Industrial Metaverse“, *IT Prof.*, vol. 25, no. 1, pp. 84–88, Jan. 2023.

[7] Y. Jiang, O. Kaynak, H. Luo, M. Liu, S. Yin, "Industrial Metaverse: Solutions from a higher-dimensional world“, Nov. 2022 (*preprint*).

[8] T. Lechler, E. Fischer, M. Metzner, A. Mayr, J. Franke, "Virtual Commissioning – Scientific review and exploratory use cases in advanced production systems“, *Procedia CIRP*, vol. 81, pp. 1125–1130, 2019.

[9] Y. Chen, S. Huang, W. Gan, G. Huang, Y. Wu, "Federated Learning for Metaverse: A Survey“, 2023.

[10] G. S. Day, P. J. H. Schoemaker, "Avoiding the Pitfalls of Emerging Technologies“, *California Management Review*, vol. 42, no. 2, pp. 8–33, Jan. 2000

[11] J. Duda, S. Oleszek, K. Santarek, "Product Lifecycle Management (PLM) in the Context of Industry 4.0“, in *Advances in Manufacturing III*, in Lecture Notes in Mechanical Engineering. Cham: Springer International Publishing, pp. 171–185, 2022.

[12] C. Lorch und B. Lüdemann-Ravit, "A new IT Architecture for the Digital Factory“, 2nd International Stuttgart Conference on Automotive Production, Stuttgart, Nov. 2022 *(in press)*.

[13] S. Suss, A. Strahilov, C. Diedrich, "Behaviour simula-tion for virtual commissioning using co-simulation“, in *IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*, Luxembourg: IEEE, pp. 1–8, Sep. 2015.

[14] P. Bartz, M. Decottignies, Y. Montagnol, K. Kouiss, F. P. Terra, C. L. D’Elia Machado, "Manufacturing machine virtual commissioning: Automated validation of the control software“, in *IEEE International Conference on Electrical, Computer and Energy Technologies (ICECET)*, Cape Town, South Africa:, pp. 1–5, Dec. 2021

[15] S. K. Jagatheesaperumal M. Rahouti, "Building Digital Twins of Cyber Physical Systems With Metaverse for Industry 5.0 and Beyond“, *IT Prof.*, vol. 24, pp. 34–40, Nov. 2022.

[16] M. A. I. Mozumder, M. M. Sheeraz, A. Athar, S. Aich, H.-C. Kim, "Overview: Technology Roadmap of the Future Trend of Metaverse based on IoT, Blockchain, AI Technique and Medical Domain Metaverse Activity“, in *IEEE 24th International Conference on Advanced Communication Technology (ICACT)*, Pyeongchang, Korea, pp. 256–261, Feb. 2022

[17] H. Ning *et. al.* "A Survey on Metaverse: the State-of-the-art, Technologies, Applications and Challenges“, 2021.

[18] *NVIDIA Omniverse - Designing, Optimizing and Operating the Factory of the Future*, (12. April 2021). [Online Video]. https://www.youtube.com/watch?v=6-DaWgg4zF8&t=24s

[19] M. Tiegelkamp, K. H. John, "SPS-Programmierung mit IEC 61131-3: Konzepte und Programmiersprachen, Anforderungen an Programmiersysteme, Entscheidungs-hilfen", Springer, Berlin, Germany, 2009.

[20] G. Shapira, T. Palino, R. Sivaram, K. Petty, "Kafka: the definitive guide: real-time data and stream processing at scale", Second Edition. Sebastopol, CA: O’Reilly, 2022.

[21] S. Quincozes, T. Emilio, J. Kazienko, „MQTT Protocol: Fundamentals, Tools and Future Directions“, *IEEE Latin Am. Trans.*, vol. 17, no. 09, pp. 1439–1448, Sep. 2019

[22] J. Aro *u. a.*, "Praxishandbuch OPC UA: Grundlagen, Implementierung, Nachrüstung, Praxisbeispiele". Würzburg: Vogel Communications Group, Germany, 2020.