



A New IT Architecture for the Digital Factory

Christopher Lorch¹(✉) and Bernd Lüdemann-Ravit²(✉)

¹ Mercedes-Benz Operations Information Technology, 71059 Sindelfingen, Germany
christopher.lorch@mercedes-benz.com

² Institute for Production and Informatics, University of Applied Science Kempten, 87527
Sonthofen, Germany
bernd.luedemann-ravit@hs-kempten.de

Abstract. In general, current systems for the Digital Factory implement a product-process-resource (PPR) data model in a monolithic rich-client/server architecture with a single database persistence layer. Common data objects are the product bills of material, descriptions of the production processes, or the resource structure, e.g. bill of equipment. The main drawback of the current monolithic architecture is the slow rate of development, which prevents fast adoption of the software to the new production planning process (i.e., due to new technologies for the transformation of the automotive industry with the goal of electrification) is not possible. Furthermore, time-consuming and error-prone export-import operations characterize the collaboration of the engineering supply chain. Mercedes-Benz has created a new IT system architecture for their Digital Factory. The core idea of this architecture is a module-based approach. Each planning step has its own module, e.g. product analysis, layout planning or cost calculation. One single module consists of a server-based business logic, a web-based user interface and its own database. Each module is the source of master data objects that originate from the corresponding planning step and refers to data objects from predecessor planning steps. The single modules communicate mostly via KAFKA. The usage of a model based application engine allows the fast creation of different modules. Best-of-breed third-party systems for specific planning steps can be integrated into the system architecture. Web technologies allow suppliers to access the Mercedes-Benz systems directly for a fully integrated supplier collaboration. Roll-out has started and has already led to significant efficiencies.

Keywords: Digital Factory · Production Planning · IT Architecture

1 Introduction

1.1 Digital Factory

The term Digital Factory characterizes systems, data models, methods and business processes during the production planning [1]. The term is used for more than twenty years and received new attention within the paradigm of industry 4.0 as it is the origin of the digital twin of the production.

1.2 Data Model PPR – Product, Process, Resource

Data models in a Digital Factory consist of three main abstract data structures (business objects) [1, 2]: product, process and resource (PPR) objects. First, the product structure contains the outcome of the factory, e.g. described by the bill of material (BoM) including fasteners. The author of the product structure is the research and development department. The production planning uses the BoM in read-only mode. Second, the process structure represents the operations that are necessary to build the product including estimated and analyzed time attributes. For example, it contains processes to assemble, weld or glue product parts. Third, the resource structure contains all equipment inside the factory: e.g. robots, tools, carriers, building and humans. Both process and resource structure are created by the production planning (PP), in parallel to the product development process (simultaneous engineering). The PP also interconnects the business objects (product, process and resource) with predefined relations according to their usage to define which product is processed inside the factory with the help of the corresponding resources (see Fig. 1).

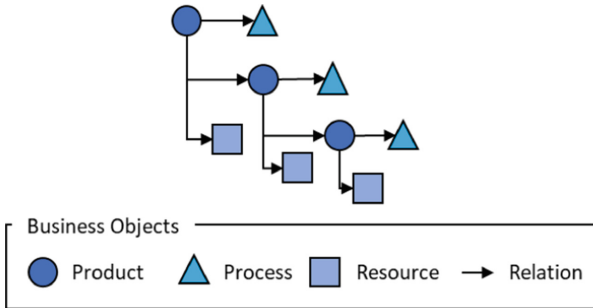


Fig. 1. Sample Data Model “Product Process Resource (PPR)” including relations to represent the Digital Factory

1.3 Scope

The scope of the current development activities of the new IT architecture focuses on body-in-white planning in an early stage starting from the preliminary planning and ending at the placing of the orders to the prime contractors for the automation equipment. The idea of this new IT architecture can also be used in other planning departments such as press shop, final assembly and intralogistics. Validation topics are out of scope in the current stage of the system design because they require extensive 3D simulation functionality. 3D visualization and modification functionality offered by this system is therefore not enough.

1.4 Main Requirements

The chance to build a new Digital Factory is also a challenge, because the corresponding IT architecture has to be suited for the requirements and upcoming ideas of industry 4.0 [3].

Requirements Due to Supplier Collaboration. Nowadays, original equipment manufacturers (OEMs) and their suppliers (Tier 1, 2, 3...) collaborate along the complete production planning process [2] until the automation equipment is ready to produce. This leads to the fact that supplier integration is one of the key aspects to establish efficient planning processes. The IT architects focused therefore on four approaches (Fig. 2).

Potential Architectures. The first option is the classical monolithic architecture offered commonly by standard software: an installation with one central database in the data center of the OEM and corresponding installations in the data centers of the suppliers (Fig. 2a). Most OEMs still use this approach for their Digital Factory. Thereby manual effort and complex interfaces are necessary to update and exchange product and planning data between the OEMs and their suppliers, since the product is still in development. Additionally, the IT departments of all participants have to synchronize the same database scheme, which is a challenging organizational task. Some software vendors propose a second approach (Fig. 2b): a supplier client accesses the OEM database.

A web-based (Fig. 2c) and a cloud-based architecture (Fig. 2d) promise a more efficient approach regarding applicability and operation.

Requirements Due to the Speed of Change. An additional and essential requirement to the IT architecture is a fast adaption and an easy extension of the Digital Factory systems according to changes in the planning process. The transformation of the car industry due to digitalization and electro-mobility will cause changes in the current planning process that no one can predict today. Adaptions and extensions of the standard Digital Factory software that take several years on the roadmap of a software vendor will significantly slow down the speed of adaption of the Digital Factory IT landscape and the support of agile production planning processes. Therefore, service-orientation and modular organization of the Digital Factory system are also key requirements.

The implication of the new IT architecture to fulfill extensibility and modularization is the concept of a distributed data model (Fig. 3). What is the key idea of this requirement? The complete business process consists of single processes with a corresponding result out of each process, e.g. the production planning consists of a process “plan resources”. Its result is the recourse bill of material. The IT architecture itself must have one module for each process with its own GUI and its own master data. One module is referring to the data of another module via a reference object. The architecture then needs mechanisms to exchange data between the modules. For example, the resource module to plan the resources contains a data model with reference objects to the product data model of the module to analyze the product (Fig. 3). The resource module stores the business objects that the planner has created within the process “plan resources” (master data resource objects).

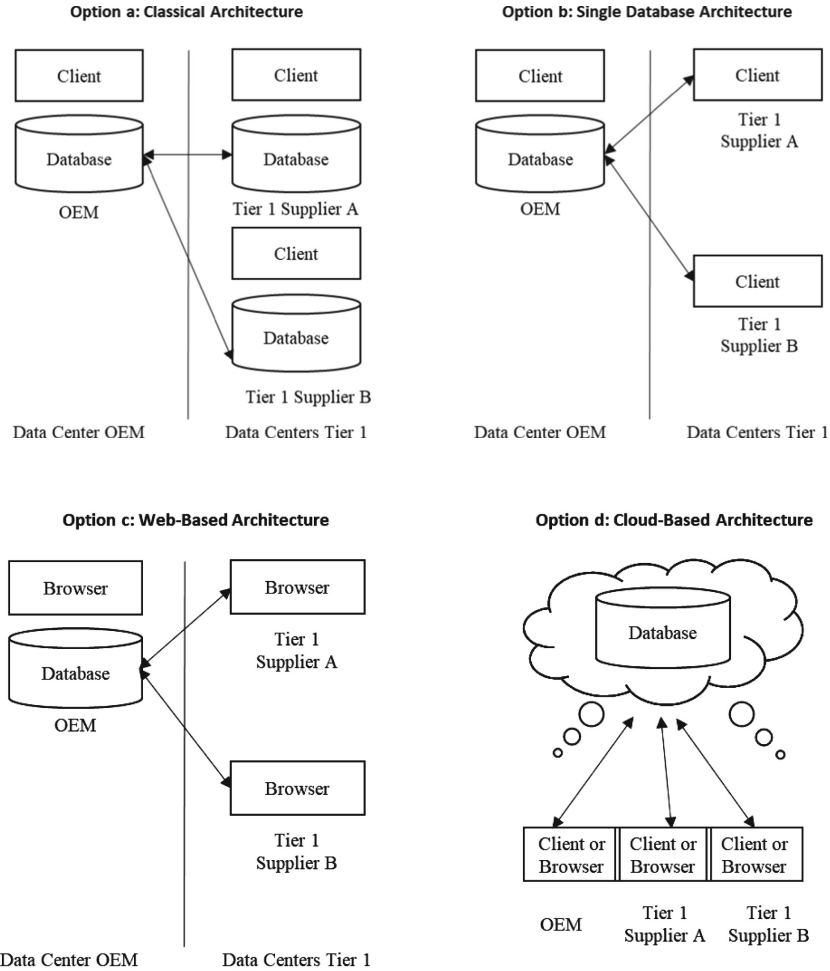


Fig. 2. a + b. Classical and Single Database IT architecture for the supplier collaboration. **c + d.** Web- and Cloud-Bases IT architecture for the supplier collaboration

Requirements Stemming from Legacy Environment. An additional challenge is the integration of the software into the existing legacy IT landscape (e.g. product documentation management (PDM) systems) or IT guidelines (e.g. identity access management infrastructure). Production planners need up-to-date product data. Therefore, an interface must provide hundreds of car configurations from the PDM system to the Digital Factory. The Digital Factory must be capable to provide or access this data and enrich downstream processes efficiently with production planning information.

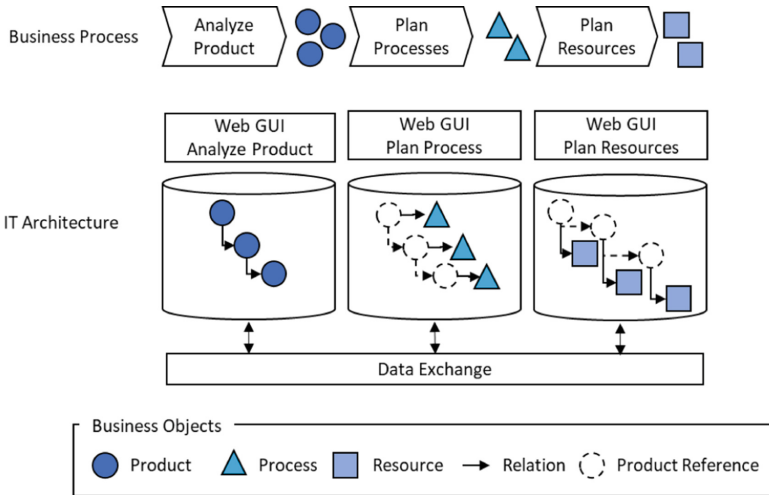


Fig. 3. Extensibility and modular organization of the system

1.5 State of the Art

An overview of commercial and scientific software tools for Digital Factories can be found in [4]. The market leaders [5, 6] started implementing either web-based architectures [7] or cloud-based architectures [8–10]. Nevertheless, all of them have one common characteristic in their architecture: the central database (Fig. 2a). In scenarios with suppliers one can find own central databases for the suppliers (Fig. 2b).

The Aras platform [11] and the ASCon Application Engine [12] have first implemented the idea of modularization of their Digital Factory solution. Both of them need additional 3D visualization, e.g. [13–15], and authoring functionality, e.g. [13].

Additionally, current IT landscapes for the domain of PLM (product lifecycle management) are dominated by few monolithic systems [21, 22]. New approaches for an IT architecture is necessary to implement the concept of a digital twin of the production [23–25].

2 System Design

2.1 General Approach

Advantages. We have chosen a modularized system design approach in order to fulfill the requirements listed above, using the ASCon Application Engine [12]. By this architecture we benefit from the potential of developing several modules in parallel and that we can focus on the use-cases most urgently needed without the necessity to implement all aspects which are needed in the implementation (or customization) of a monolithic system architecture. However, the open architecture approach is a sound basis for extending the ecosystem in order to match future use-cases like the realization of a digital twin

of the plant. Furthermore, from scratch design of the application allows a performant implementation of the product structure data model.

Disadvantages. Nevertheless, there are also drawbacks then creating such a modularized system architecture from scratch. On the one hand, the design of a shared data model and the modeling of the interactions between different applications are crucially important. On the other hand, this design makes it more difficult for the product owners of the individual modules to implement the correct features.

Web-Based Approach. The majority of the applications are web-based, which has several advantages. First, no sophisticated rich client installation on the end-user devices is necessary. The hardware requirements of the latter are lower for web-clients than for rich clients offering 3D modification features. In addition, less data needs to be transferred from the application server to the client if almost no processing is done on the client tier.

All of the above is very helpful to improve the collaboration with suppliers. OEM ask suppliers to work on the OEM's infrastructure (Fig. 2c, b). This shortens the overall planning time since no exchange of large data sets, complex export and import mechanisms and distribution of application software is needed. Furthermore, the suppliers do not need to purchase specific software or licenses. These advantages outnumber in the long run, the enlarged investment of the OEM in infrastructure due to a higher number of users.

Table 1. Overview of implemented modules.

Application Name	Description of the application
<i>DiFaProject</i>	Central project structure; user management; rights and roles;
<i>DiFaLibrary</i>	Life cycle and information of single equipments; definitions of equipment modules
<i>DiFaCost</i>	Costs of the different equipments; overhead cost
<i>DiFaProduct</i>	Analysis of the product and fasteners; comparison of different product configurations
<i>DiFaPlanning</i>	Planning of the mechanical part of the resource structure incl. Layout
<i>DiFaControls</i>	Planning of the electrical part of the resource structure

2.2 Interface Design

In general, the architecture uses two different technologies for the exchange of information between modules.

Streaming. Firstly, KAFKA [16] is the technology to implement use cases with an asynchronous communication based on an event-driven approach. This technology has several benefits for our modularized architecture. Applications can release events and

any number of applications can consume the information from the events. This makes the scenario easily extendable for the implementation of further use-cases, since the application providing the information does not need to be changed. Furthermore, different applications in the downstream processes can directly use the very same information provided by one source.

An example for a use case implemented via KAFKA is the release of a new equipment in the *DiFaLibrary* (Fig. 4b). An event containing the equipment information triggers a new calculation task in *DiFaCost* to plan costs. Additionally, the central planning application, *DiFaPlanning* consumes the message and provides the new equipment for resource planning. In the latter use case, *DiFaPlanning* also automatically checks if this new equipment replaces an older one, and, in this case, provides the information to the users so that they can update their resource planning with the new version of the equipment. Future use cases which could be implemented directly in the event of a new released equipment could be either the trigger for the purchasing department to start negotiations with the supplier or the contract department to start with the detailed design phase of the new equipment.

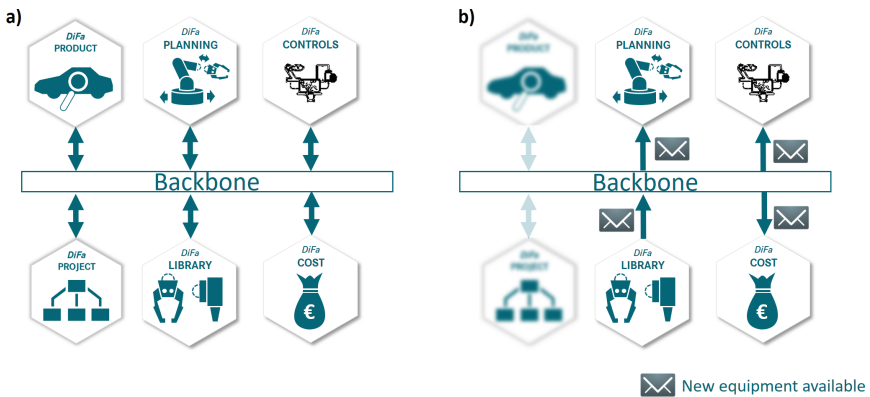


Fig. 4. a) Schematic overview of the six main applications generated in the first iteration of the implementation. Table 1 summarizes the idea behind the single applications. b) Example flow of information shared by *DiFaLibrary*, which is processed by several other applications

REST APIs. On the other hand, additional use cases were implemented using Representational State Transfer (REST) APIs [17, 18] for synchronous communication needs. Some data is just too big or changes happen too frequently for an efficient use of the KAFKA bus. A good example for this is the simultaneous mechanical and electrical resource planning. Traditionally, different user groups work in these two domains, however they are mutually dependent. For this use case both planning teams work on the same resource structure, using also the same equipment library, but with different optimized tools for the requirements of the individual user group.

2.3 Example Application

One typical and rather complex application is the resource planning application *DiFaPlanning*. We will highlight the most interesting parts. We created *DiFaPlanning* on a web software framework provided by the company ASCon Systems [12] and the 3D rendering service Uber provided by the company Netallied [13].

This application is used to create and modify the resource structure of the production plants and to provide a 3D layout of the production lines. Figure 5 shows a screenshot of a typical scenario. On the left, a tree structure of the resources is visible to navigate to the desired planning area and to list the planned resource bill of material. The middle section contains a view of reference objects to the equipment library originating in *DiFaLibrary*. The window on the right contains a 3D representation of the current planning scenario. The 3D scene is rendered on the server side and the information is shared with the user via WebSockets. Information is exchanged between the web framework and the 3D scene using REST APIs.

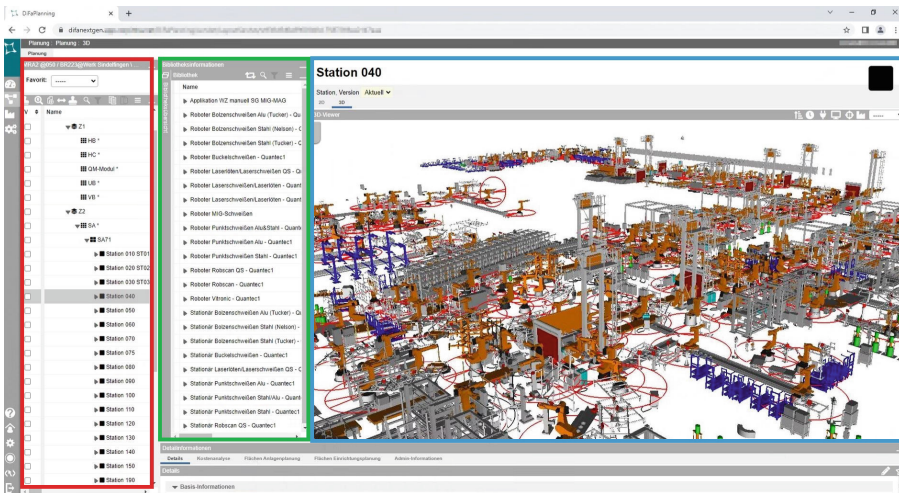


Fig. 5. Screenshot of *DiFaPlanning*. On the left hand side (marked with red), one can see the tree representation of the resource structure. The middle column (marked with green) shows only equipment that can be used for planning out of *DiFaLibrary*. On the right (marked with blue), 3D planning of the resources is shown.

3 Outlook

The implementation of the Digital Factory of Mercedes-Benz is by no means done. With the architecture described above, the fundamentals for the implementation of future use cases is set.

Additional Applications. One could think in various directions for the enhancement and expansion of the application landscape. So far, the three branches of the PPR are not

completely linked. Even though this results in some liberties for the users, a closer link between the branches will help to analyze necessary changes in the process or resource structure that lead to changes of the product structure. In addition, this is necessary when implementing an approach to create automatically the process and resource structures from a given product structure [19].

Integrated Change Management. Another interesting aspect to follow is a fully integrated change management over the whole planning and production cycle. This would be the basis for creating a full digital twin of the production facilities. In order to realize this, a complex branching concept of the different structures is necessary which allows, on the one hand, to see the evaluation in time, and, on the other hand, different planning alternatives representing different possibilities to realize the planning.

Online Validation. During the collaboration process, Mercedes-Benz requests usually different deliverables from the engineering partners. The integration of online validation tools in the architecture will further improve the data quality and reduce the realization time for new production facilities.

4 Conclusion

Implementation Speed. After the successful go live of the first applications, we have seen that the chosen approach is fruitful and sustainable. The implementation was approximately two times faster compared to the approach of customizing a commercial software solution to fit the OEM-specific requirements. We achieved this faster development speed mainly because we could develop several modules in parallel and we have implemented or customized only parts necessary for our processes. By setting up the already existing application engine ASCon Map [12] we have benefited from the already available functionality, such as user management, monitoring, logging and default user control features.

Performance. A great benefit of the architecture is the enormous speed of loading and visualization of 3D data. For a fully configured product structure, both the loading of the product structure and the visualization takes approximately 30 s, thanks to the visualization engine Uber [13]. In comparison, loading and visualization took usually around 30 min within the former Digital Factory system of Mercedes-Benz.

Additionally, we achieved the requirement of loading hundreds of car configurations from the PDM system into the Digital Factory with this architecture approach.

Integration with Existing Applications. We were able to add further, earlier excised, applications to the ecosystem and we are very positive that additional applications will be part of the Digital Factory in the future.

Acknowledgment. The authors thank the whole project team at Mercedes-Benz AG, Mercedes-Benz Research and Development India, ASCon Systems GmbH and NetAllied Systems GmbH for the effort put into the realization and the fruitful collaboration.

References

1. Digital Factory – Fundamentals, VDI 4499 Part 1, Verein Deutscher Ingenieure e.V. 8, Beuth, Berlin (2008)
2. Bracht, U., Geckler, D., Wenzel, S.: *Digitale Fabrik*, 2nd edn. Springer Vieweg, Berlin (2018)
3. Duda, J., Oleszek, S., Santarek, K.: Product lifecycle management (PLM) in the context of industry 4.0. In: Trojanowska, J., Kujawińska, A., Machado, J., Pavlenko, I. (eds.) *MANUFACTURING 2022*. LNME, pp. 171–185. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-99310-8_14
4. Chen, D., Kjellberg, T., von Euler, A.: Software tools for the digital factory – an evaluation and discussion. In: Huang, G.Q., Mak, K.L., Maropoulos, P.G. (eds.) *Proceedings of the 6th CIRP-Sponsored International Conference on Digital Enterprise Technology. Advances in Intelligent and Soft Computing*, vol. 66. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-10430-5_62
5. Top 10 PLM and Engineering Software Vendors, Market Size and Market Forecast 2020–2025, <https://www.appsruntheworld.com/top-10-product-lifecycle-management-engineering-software-vendors-and-market-forecast>. Accessed 27 May 2022
6. Product Lifecycle Management (PLM) Software Market - Growth, Trends, COVID-19 Impact, and Forecasts (2022–2027). <https://www.mordorintelligence.com/industry-reports/product-lifecycle-management-software-market>. Accessed 27 May 2022
7. Active Workspace | Siemens Software. <https://www.plm.automation.siemens.com/global/en/products/collaboration/active-workspace.html>. Accessed 27 May 2022
8. Liu, L.: Application of Dassault system 3D experience platform in enterprise digitalization. In: Hung, J.C., Yen, N.Y., Chang, J.W. (eds.) *Frontier Computing*. FC (2021)
9. Sharma, S., Segonds, F., Maranzana, N., Chasset, D., Frerebeau, V.: Towards cloud based collaborative design – analysis in digital PLM environment. In: Chiabert, P., Bouras, A., Noël, F., Ríos, J. (eds.) *PLM 2018*. IAICT, vol. 540, pp. 261–270. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01614-2_24
10. Niemann, J., Pisl, A.: Teamcenter data management. In: *Life-Cycle Management of Machines and Mechanisms*. MMS, vol. 90, pp. 381–396. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-56449-0_19
11. Pfenning, M.: *Breaking Down the Silos - How Systems Architecture Enables Interdisciplinary Collaboration*. Aras Corporation, White Paper (2020)
12. ASCon Digital Twin Product Suite - Application Engine. <https://ascon-systems.de/product-suite/application-engine>. Accessed 27 May 2022
13. NetAllied Systems GmbH. <https://netallied.de/>. Accessed 17 May 2022
14. b.ReX GmbH. <https://b-rex.de>. Accessed 17 May 2022
15. Kisters AG. <https://viewer.kisters.de>. Accessed 17 May 2022
16. Shapira, G.: *Kafka - The Definitive Guide: Real-Time Data And Stream Processing at Scale*, 2nd edn., O'Reilly, Sebastopol (2021)
17. Fielding, R.: *Architectural Styles and the Design of Network-Based Software Architectures*. University of California, Irvine (2000)
18. Wikipedia. https://en.wikipedia.org/wiki/Representational_state_transfer, Accessed 17 May 2022
19. Hagemann, S., Stark, R.: An optimal algorithm for the robotic assembly system design problem: an industrial case study. *CIRP J. Manuf. Sci. Technol.* **31**, 500–513, Elsevier (2022)
20. Stark, R.: *Virtual Product Creation in Industry*, 1st edn. Springer, Berlin, Heidelberg (2022)

21. Hooshmand, Y., Resch, J., Wischnewski, P., Patil, P.: From a monolithic PLM landscape to a federated domain and data mesh. In: Proceedings of the Design Society, vol. 2, pp. 713–722 (2022)
22. Bleisinger, O., et al.: Killing the PLM Monolith—the Emergence of Cloud-Native System Lifecycle Management (SysLM) (2022)
23. Aheleroff, S., Xu, X., Zhong, R., Lu, Y.: Digital twin as a service (DTaaS) in industry 4.0: an architecture reference model. *Adv. Eng. Informat.* **47** (2021)
24. Talkhestani, A., et al.: An architecture of an Intelligent digital twin in a cyber-physical production system. *At - Automatisierungstechnik* **67**(9), 762–782 (2019)
25. Eric Harper, K., Ganz, C., Malakuti, S.: Digital twin architecture and standards, *IIC J. Innov.* **12**, 72–83 (2019)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

