DOI: 10.22032/dbt.57902

Simulation in Produktion und Logistik 2023
Bergmann, Feldkamp, Souren und Straßburger (Hrsg.)
Universitätsverlag Ilmenau, Ilmenau 2023
DOI (Tagungsband): 10.22032/dbt.57476

Cross-Domain Simulation based on BPMN and the Process-Driven Approach

Domänenübergreifende Simulation basierend auf BPMN und dem Prozessgesteuerten Ansatz

Marvin Schobert, Pooria Lahooti, Jörg Franke, Institute for Factory Automation and Production Systems (FAPS), Erlangen (Germany), marvin.schobert@faps.fau.de, pooria.lahooti@fau.de, joerg.franke@faps.fau.de

Abstract*:* Simulation tools are currently facing two significant challenges: On the one hand, common applications are often focussed on a specific domain, which leads to a high synchronisation and integration effort in the investigation of cross-domain issues. On the other hand, the transition from simulated models to operations monitoring and continuous improvement requires a high further development effort or even a completely new implementation with other technologies. This paper aims to present a novel one-size-fits-all approach based on the Business Process Model and Notation (BPMN) standard and the Process-Driven Approach (PDA) for creating and executing process models for discrete event simulation. It combines the BPMN's potential to graphically represent any processes in arbitrary precision with those of PDA to sustainably integrate services, databases and tools via low-code. By means of two exemplary case studies, we would like to illustrate its practical implementation and potentials and challenge the scientific discourse.

1 Introduction, Motivation and Structure

Against the background of increasing complexity of variants and supply chains, rising global competitive pressure and continuous technological progress, interactions between different functional and technical business areas are intensifying. Simulation tools are currently facing two significant challenges in this respect: On the one hand, common applications are often focussed on a specific domain, such as kinematics, material flow or process technology and a specific discipline like production or business processes, which leads to a high synchronisation and integration effort in the investigation of cross-domain issues. The complexity and sophistication of each individual tool requires expert and application knowledge, potentially strengthening domain-based silo thinking. On the other hand, evolving from simulated models to operations monitoring and continuous improvement requires high further efforts or even a new set up with different technologies. (Yu and Zheng, 2021)

Consequently, research is still needed to enable a deeper integration of discrete event simulation (DES) within crossing domains as well as the operations of production and logistics systems, while lowering its implementation efforts. This paper aims to present a novel one-size-fits-all approach based on the Business Process Model and Notation (BPMN) standard (OMG, 2013) and the Process-Driven Approach (PDA) (Stiehl, 2014) for creating and executing DES experiments. Hereby, we expect three main advantages: (1.) Cross-domain investigation gets simplified and can be done by asset managers (or citizen developers) instead of system experts only, (2.) the comprehensive and precise language of BPMN as a *lingua franca* enables overarching processes of any domain in any level of detail and (3.) lastly the PDA reduces change efforts by applying state of the art distributed software and low-code concepts to the field of simulation. Our paper will be structured as follows: First, relevant basics about BPMN, the PDA and microservice architecture will be briefly explained. Afterwards, state of the art and how this paper fits in will be discussed. The main part forms the introduction and explanation of our concept, as well as following its illustration by the practical implementation and discussion of two differently characteristic use cases. Finally, future efforts and further questions are addressed in a conclusion and outlook.

2 Relevant Basics and State of the Art

VDI 3633 Part 1 (2014) defines the term of *simulation* as the "representation of a system with its dynamic processes in an experimentable model to reach findings which are transferable to reality; in particular, the processes are developed over time". In a deeper dive, two terms have greater significance: *Model* ("Simplified reproduction of a planned or existing system with its processes in a different conceptual or concrete system") and *Process* ("Full set of interacting operations in a system through which material, energy or information are processed, transported or stored"). Due to the large number of time-dependent and random system variables and the highly networked interactions, mathematical-analytical methods quickly reach their limits when investigating such systems. Simulation, on the other hand, can be used to examine the temporal behaviour of complex technical systems. (VDI, 2014)

Typically, common simulation tools are good in investigating the interactions of technical systems within a single domain (e. g. Tecnomatix Plant Simulation for material flow analysis, NX MCD for kinematics simulation or Matlab for physics simulation). For questions involving multiple domains (like "How does a change request within the bill of material affect the return on invest of certain equipment?"), iterative adaption effort and the combined use of multiple distributed systems is incurred. To express the relations within a system containing multiple models, involved assets can be structured accordingly to VDI 3682 (2015): Products, energies or information (inputs) are converted to new products, energies or information (outputs) by a process operator, which is executed by a technical resource. More common, the generalized *product*, *process* and *resource* (also shortly known as PPR-model) approach is being used (Schleipen et al., 2015), which will also be referred to within this publication. For machine-executable simulation, it is important to consider the transfer of those principles to distributed software architecture (see Figure 1). Products and information objects can be resembled by stateful business objects, containing descriptive variables. Processes consist of chained activities (or method operations), which perform basic *create*, *read*, *update* or *delete* (CRUD) operations on the business objects. In terms of (business) process management, activities are executed through human input or by dedicated IT-services, representing resources like e. g. humans, organisations, machines or robots (Freund and Rücker, 2019).

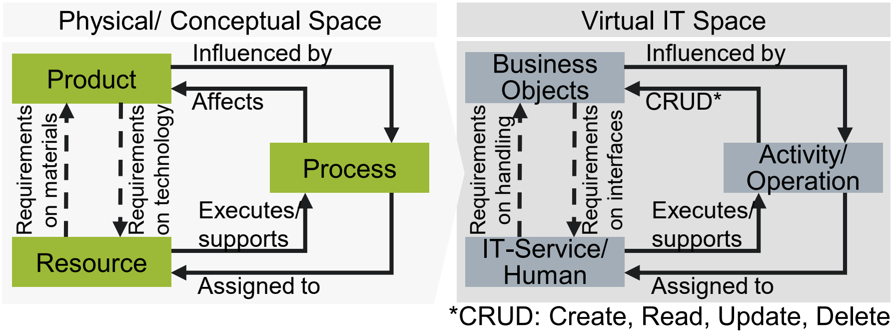


Figure 1: PPR-model within IT-context (left: based on VDI 3682)

Since processes' activities represent the connecting units and are mainly dominated by domain experts, a process definition language should be intuitive, standardized, complete, precise and have the capability to be quickly transferred into executable software. The Business Process Model and Notation (BPMN), (OMG, 2013) makes it possible to modularise complex behavioural models and break them down into its elementary activities, conditions, transitions and events. Unlike other process notations such as the event-driven process chain, BPMN is standardised, offers a comprehensive palette for the precise graphical description of processes and enables the direct execution of the models in a process engine (Freund and Rücker, 2019). This orchestrates, monitors, integrates, reports and manages all tasks (see Figure 2).

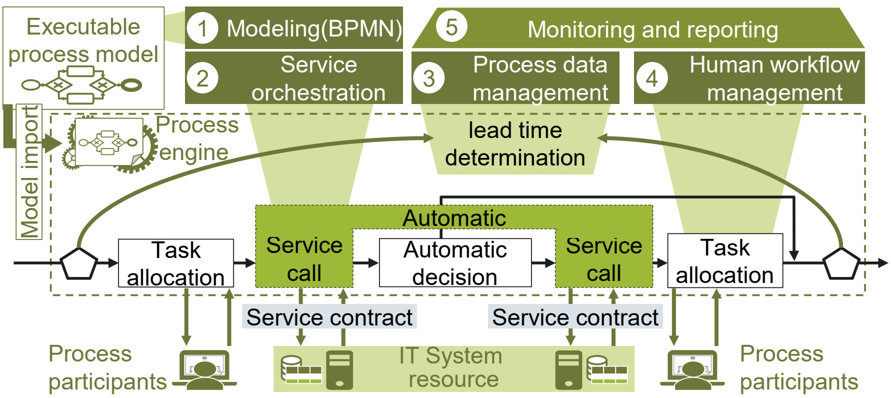


Figure 2: Features and Functionalities of a Process Engine (Schäffer et al., 2021b)

From a formal point of view, not only business relationships can be mapped in any abstraction, but also technical processes according to VDI 3682. The Process-Driven Approach (PDA) suggests an architecture, where process logic is executed within a process engine and external services can be accessed using a predefined service contract describing its interface structure via asynchronous messaging (Stiehl, 2014). As long as both interfaces stay with the exchange format defined by the service contract (variables in e. g. JSON or XML), the IT-systems can be developed, maintained, exchanged and executed independently from business logic. In the context of business process modelling, events represent transaction incidents causing the development of a process, like customer and production orders, change requests, timers or task acknowledgements. Within DES, events are more generally defined by VDI 3633 as an "atomic incident that causes a transformation of state […].".

Various case studies have proven the operational use of BPMN in structured, organisational contexts like customer relationship management or request processing (Vom Brocke, J. et al. 2021). A previous publication involving the main author introduced an architecture and development method for process-driven web platforms within engineering, connecting users, systems and knowledge with operational planning processes (Schäffer et al., 2021a). In terms of asset lifecycle, those considerations exclusively map to operations, but will be adapted towards phase-independency within this paper. The idea of using BPMN for simulation generally exists in practice already: Some providers of workflow management systems (e.g. SAP Signavio, Trisotech) advertise functionalities for DES of BPMN process models in their portfolio. However, this simulation of process models is limited to the process engine's system context for validation and optimization. Due to the missing integration of stateful systems, only the process itself can be simulated without being able to consider the interaction with stateful assets like products or resources.

Comparable research is mostly assigned to business informatics and deals with the mapping and analysis of business processes, BPMN-driven software architectures for Industry 4.0 and extensions for dedicated application areas. Steindl and Kastner propose and demonstrate a service framework architecture aligned to the RAMI4.0 IT layers with a message-oriented middleware concept similar to ours (see chapter 3) (Steindl and Kastner, 2021; DIN, 2016). The discrete event simulation capability of BPMN has generally been proven by Guizzardi et al. (2011). The publications that deal with simulation either do not explicitly consider assets like products or resources, do not involve networked systems and therefore enable limited findings, or focus on organisational processes only. Some of the research papers worth mentioning will be briefly presented: Dorrer (2020) suggests a solution to forecast parameters like duration, labor intensity, material consumption and costs by translating a BPMN model describing a manufacturing process to an equivalent, parametrized graphical evaluation and review technique-(GERT)-model, a probabilistic model for discrete-event systems. Vlasov and Gonoshilov (2019) simulate electronics manufacturing systems with the use of BPMN without explicitly considering products or resources and staying within process engine's system as well. The diagrams show interactions between the different pools, but are not capable of doing further investigations. Durán et al. (2021) propose an extension to BPMN allowing to annotate models with resource, probability and duration information separately for the process execution, which enables automated parameter optimization. Ougaabal et al. (2019) distinguish process and resource types at different domains (IT, human/ organisation and physical means) and levels of abstraction (Business system, technology independent and technology specific models) in BPMN workflows during the simulation phase by using an extension enabling the consideration of resources. Respecting those publications, the simultaneous consideration of processes along with products and resources has not yet been considered satisfactorily. In contrast to some of these approaches, we do not want to modify the established BPMN standard. Given that BPMN itself is not intended as a modelling tool for technical systems or resource management, we think it makes sense to keep process logic separated from asset persistence and therefore see BPMN as more of an orchestrator of sub-processes, activities and other systems.

3 Concept Introduction

The concept is described below using the three axes of application control, asset lifecycle and domain (see Figure 3). The general structure and architecture are influenced by the PDA (Schäffer et al., 2021b) and RAMI4.0 (DIN, 2016). While application control describes the technical architecture, the asset lifecycle point of view contains the use of external systems depending on its state. The domain dimension represents a specific view on the asset. The novelty is created by separating the executed logic (BPMN within a process engine), transaction data (events and protocols) and asset handling (back-end-services, databases and control interfaces).

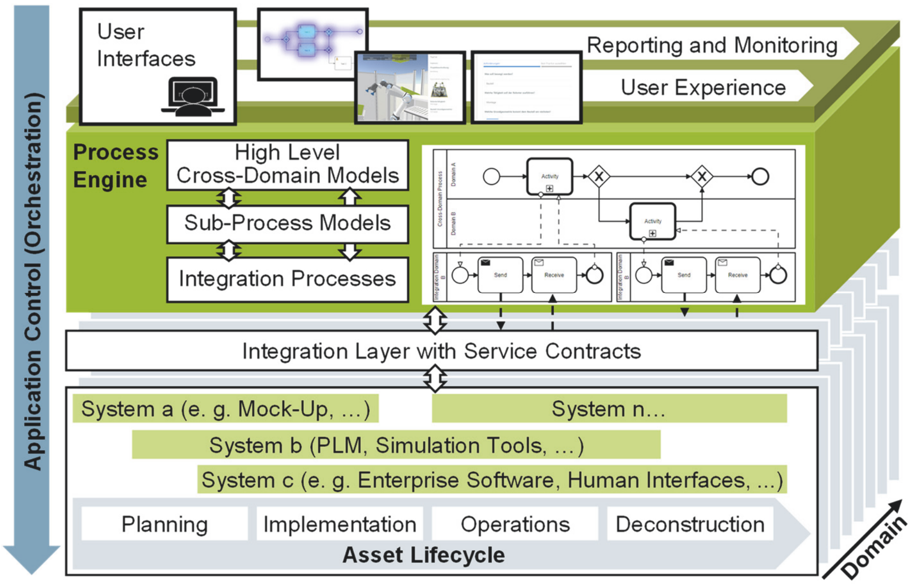


Figure 3: Concept composition of Process-Driven Cross-Domain Simulation

3.1 Application Control (Y-Axis):

The technical set up of a process-driven application in the context of simulation can be described following along the axis of application control, inspired by Stiehl (2014) and Schäffer et al. (2021b). It builds upon the concepts of distributed software architecture, execution logic expressed through BPMN and (micro-)service orchestration. The application is to be controlled and monitored via user interfaces containing especially dashboards and possibly user forms, if user tasks are to be completed selectively through dedicated front ends within the simulation. The core process layer contains the cross-domain and -departmental BPMN models, which are to be modelled according to the style guide proposed by Silver and Fischli (2012). It is intended that a process model only displays the logic that is necessary at the corresponding level of abstraction and the more detailed description is outsourced to sub-processes. An integration layer should be used to connect independent, stateful systems to the process models by the use of service contracts. The process engine coordinates all transactions between participating systems, thereby achieving increased transparency, maintainability and scalability. By doing so, connected systems can be exchanged throughout the applications lifecycle without the need to change execution logic. The services for activity execution are to be connected through asynchronous messaging with events for loose coupling. Events get instantiated by the activities defined within the process models and cause CRUD-operations on the asset's states. Those can be stored in databases or systems (e. g. inventory), dedicated simulation tools (e. g. plants and cells) or physical environment.

3.2 Asset Lifecycle (X-Axis):

Since task execution is separated from process definition, process and data models don't need to be changed throughout the asset's lifecycles. Given the PDA's orchestration architecture, traditional choreography having many mutual interconnections between corresponding systems can be avoided (see Figure 4). Multiple assets with different lifecycle states each can be handled as well by the same executable process model. The degree of autonomy of the simulation can be flexibly adjusted: Automatic *Service Tasks* for system interaction, *Send* and *Receive Event Tasks* in combination with asynchronous communication for resource allocation and *User Tasks* for desired user interaction during simulation.

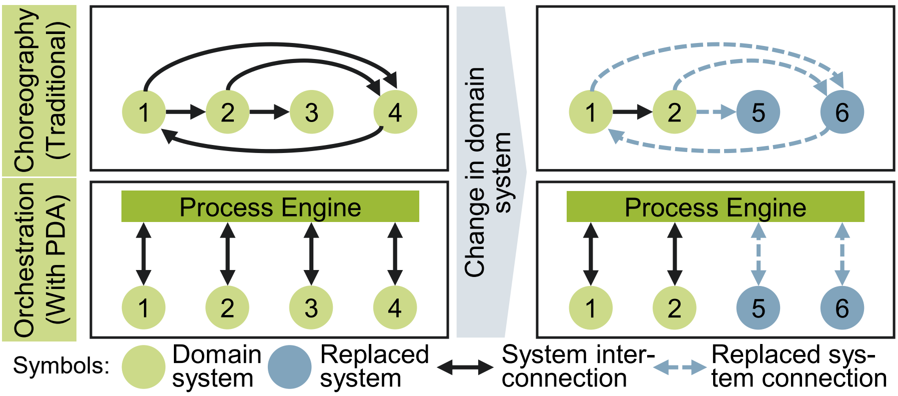


Figure 4: Interconnection between domain components

3.3 Domain (Z-Axis):

The concept of cross-domain involves the segmentation of interconnected systems, whereby each system is solely linked to the process engine (see Figure 4). Different domains can be various technical fields like mechanics, electrics and fluid dynamics as well as different business areas like production, purchasing or planning. Similar to the lifecycle point of view, the approach benefits from a process engine that remains constant regardless of whether we are in a simulation or operational phase and have changes in domain systems. If, for example, two systems undergo changes during their lifecycles, only the interconnections between the modified systems and the process engine need to be slightly adjusted, while keeping remaining systems unchanged. As a result, this new approach shows efficiency for systems with numerous interconnections, offering the advantage of reduced complexity.

4 Exemplary Case Studies

The following will briefly describe the two case studies that were examined as part of the research. The first one (a)) resembles an exemplary, abstracted order fulfilment (also known as order to cash) process as commonly executed within small and medium sized enterprises while simulating knowledge worker tasks as well as inventory and finances development. The second case study (b)) considers a material flow simulation scenario with separation of process control, production task execution and product handling. Figure 5 shows the composition of each case study. The light green fields represent the business and technical process models, while dark green depict databases and back-ended systems. In scenario a), task execution (dark grey fields) is simulated by a stateful node.js-server, whereas in scenario b) a stateful Unity material flow simulation application is responsible for task execution. For ease of readability, the different domains are explicitly modelled in one dimension deviating from the concept's representation in Figure 3 and one only time section is shown.

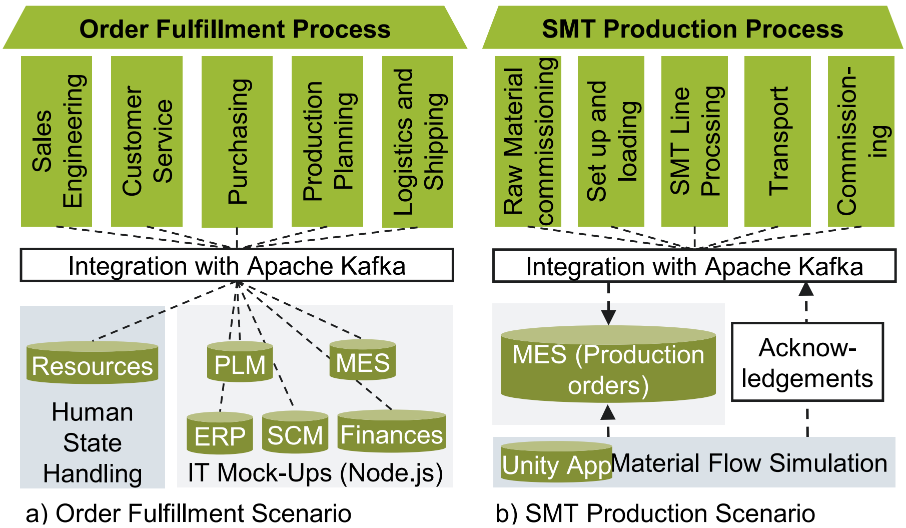
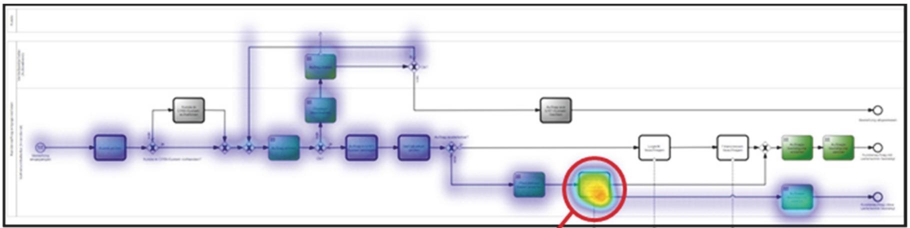


Figure 5: Technical composition of the two case study scenarios

4.1 Order Fulfilment Process

In case study one, an abstracted SME-inspired order fulfilment process was modelled in order to present the cross-domain idea, including the areas of sales, purchasing, customer service, production planning, production operations, logistics and finances.

A process instance is started by the arrival of an incoming order message coming from a simulation server via Apache Kafka. Thereby, any statistical custom order models can be fed into the system. Throughout the process, mock-up ERP, PLM and financial systems were integrated and all tasks to be performed by humans were carried out by a stateful resource-simulation-webservice bases on node.js. As a result, all process variants with their time and cost values as well as the utilisation of the individual resources can be traced over the entire order processing. The large amount of generated data (activity-, process-, business object- and transaction-data) makes each transaction traceable and provides information about occurring bottlenecks. As shown in Figure 6, the critical path and bottlenecks can easily be identified with the use of a heat map and resulting data. In the fictional scenario presented, the purchasing department is the main bottleneck, since most of the bill of materials products were purchased parts and no process automation took place. The rather low utilization rate is due to the time needed to send the events, which was not eliminated here and thus distorts the statistics. With increasing model accuracy, the replacement of simulated tasks with user tasks providing a dedicated front-end was successfully tested showing challenges in synchronizing simulation time.



Abteilung	Gesamte Arbeitszeit [min]	Auslastung	Wertschöpfung	Ausgaben
Vertrieb	80	7 %	57.77777777777778 EUR	777.1111111111111 EUR
Produktionsplanung	289	27 %	256.8888888888889 EUR	956.4444444444445 EUR
Logistik	105	10 %	47.25 EUR	484.2 EUR
Produktionsmitarbeiter	42	4 %	18.9 EUR	484.2 EUR
Einkauf	651	61 %	578.6666666666666 EUR	956.4444444444445 EUR
Finanz	15	1 %	13.333333333333334 EUR	956.4444444444445 EUR

Figure 6: Critical path identification with heatmap (top) and values (bottom)

4.2 Combined Material Flow Simulation

The second case study aims to examine co-simulation potentials by integrating an outsourced material flow simulation into the process. For this purpose, the production process of a Surface Mount Technology (SMT) line of electronics production was modelled with BPMN, which accordingly writes orders into a MES. The material flow simulation tool (implemented with the game engine Unity for visualisation and web deployment purposes) queries orders from the MES via a TCP service and sends a completion notification event to the process engine once finished. Figure 7 shows the total production task list as also stored within the MES, the front-end of the web-based material flow system as well as the currently inspected process activity. Using this, the concept's general capability to function as an orchestrator for distributed simulation has been successfully proved. Simultaneously, different kinds of process logic (e. g. business, knowledge work, kinematic, integration, physical) show potential to be modelled in BPMN and executed in various custom systems.

Figure 7: Production task list, current task and dedicated simulation environment

5 Conclusion and Outlook

In order to face current challenges concerning cross-domain simulation as well as combined cross-lifecycle investigation and operation, this paper presents a concept based on the BPMN standard and the Process-Driven Approach (PDA) for the creation and execution of tailored individual solutions making use of DES. It combines the BPMN's potential to graphically represent any processes in arbitrary precision with those of PDA to sustainably integrate external services, databases and tools into the process via low-code modules. In more detail, cross-domain process models are created in various degrees of abstraction with BPMN and compiled as well as executed within a process engine. By separating business logic from systems integration processes making the use of subprocesses and event stream processing, external subsystems specialised in certain domains can be orchestrated and exchanged throughout the applications lifecycle as needed without having to change the logic. This enables Citizen Developers and Process Owners, as people with a lot of domain expertise but potentially little IT knowledge, to map their cross-domain workflows in a tailored way and divide the details among the appropriate experts. Within this paper, two case studies are presented: One deals with the order fulfilment process across multiple departments and IT-systems, the other one regards distributed material flow simulation by outsourcing product handling in an external simulation system. While the general functionality, user-friendliness and high expressiveness of applications created with this approach have been demonstrated, long-term studies and systematic comparisons and integration with common languages, such as STEP AP242, FMI or AutomationML for co-simulation still need to be performed. Future research conducted by the main author will feature two different aspects among others: The simulation and operational execution of flexible processes (e. g. engineering projects), which have partially unpredictable agile phases, will be considered. For this purpose, the mapping of state machines using the Case Management Model and Notation (CMMN) standard as a flexible variant of BPMN is considered. Furthermore, the interaction between process data, business objects, transaction data and state data will be researched. For this purpose, one approach pursues the creation of a knowledge graph, which is fed with correlations and transactions via the process engine and can be created by DES at early planning stages already. The research occurs within the project “PDA-RobE”, receiving funding by the VDI/VDE.

References

Deutsches Institut für Normung e. V.: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0): DIN, 2016.

Dorrer, M.G.: The digital twin of the business process model. Journal of Physics: Conference Series 1679 (2020) 3, 32096 (without page numbers).

Durán, F.; Rocha, C.; Salaün, G.: Resource provisioning strategies for BPMN processes: Specification and analysis using Maude. Journal of Logical and Algebraic Methods in Programming 123 (2021), 100711 (without page numbers).

Freund, J.; Rücker, B.: Real-Life BPMN: Using BPMN and DMN to analyze, improve, and automate processes in your company. Berlin: Camunda 2019.

Guizzardi, G.; Wagner, G.: Can BPMN Be Used for Making Simulation Models? In: Barjis, J. (Hrsg.): Enterprise and Organizational Modeling and Simulation. Berlin, Heidelberg: Springer Berlin Heidelberg 2011, pp. 100–115.

Object Management Group: ISO/IEC 19510: Information technology - Business Process Model and Notation (BPMN): OMG, 2013.

Ougaabal, K.; Ducq, Y.; Zacharewicz, G.; Tazi, S.: Distinguishing Resource type in Bpmn Workflows at Simulation Phase. In: 2019 Spring Simulation Conference (SpringSim), Tucson, AZ, USA, 29.04.2019 - 02.05.2019, 2019, pp. 1–12.

Schäffer, E.; Schobert, M.; Reichenstein, T.; Selmaier, A.; Stiehl, V.; Herhoffer, M.; Mala, M.; Franke, J.: Reference Architecture and Agile Development Method for a Process-Driven Web Platform based on the BPMN-Standard and Process Engines. Procedia CIRP (2021a).

Schäffer, E.; Stiehl, V.; Schwab, P.K.; Mayr, A.; Lierhammer, J.; Franke, J.: Process-Driven Approach within the Engineering Domain by Combining Business Process Model and Notation (BPMN) with Process Engines. Procedia CIRP 96 (2021b) 57, pp. 207–212.

Schleipen, M.; Lüder, A.; Sauer, O.; Flatt, H.; Jasperneite, J.: Requirements and concept for Plug-and-Work. at - Automatisierungstechnik (2015), pp. 801–820.

Silver, B.; Fischli, S.: BPMN Methode und Stil: Cody-Cassidy Press 2012.

Steindl, G.; Kastner, W.: Semantic Microservice Framework for Digital Twins. Applied Sciences 11 (2021) 12, 5633 (without page numbers).

Stiehl, V.: Process-Driven Applications with BPMN. Cham: Springer International Publishing 2014.

Verein Deutscher Ingenieure: VDI 3633 Blatt 1: Simulation von Logistik-, Materialfluss und Produktionssystemen Grundlagen: VDI, 2014.

Verein Deutscher Ingenieure: VDI/VDE 3682 Blatt 1: Formalisierte Prozessbeschreibungen: Konzept und grafische Darstellung: VDI, 2015.

Vlasov, A.I.; Gonoshilov, D.S.: Simulation of manufacturing systems using BPMN visual tools. Journal of Physics: Conference Series 1353 (2019) 1, 12043 (without page numbers).

Vom Brocke, J.; Mendling, J.; Rosemann, M. (Hrsg.): Business Process Management Cases Vol. 2: Digital Transformation - Strategy, Processes and Execution. Berlin, Heidelberg: Springer Berlin Heidelberg 2021.

Yu, F.; Zheng, C.: Tools, application areas and challenges of factory simulation in Small and Medium-Sized Enterprises – A Review. Procedia CIRP 104 (2021), pp. 399–404.