Bridging the Gap between Business Process and Simulation: Transformation from BPMN to DEVS (I)

Mariane El Kassis, François Trousset, Grégory Zacharewicz, Nicolas Daclin

Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 56-2 (2023) 11888–11893

Bridging the gap between business process and simulation: transformation from BPMN to DEVS

Mariane El Kassis, François Trousset, Gregory Zacharewicz, Nicolas Daclin

Laboratory for the Science of Risks, IMT Mines Ales, Ales, France (e-mail: mariane.el-kassis@mines-ales.fr, francois.trousset@mines-ales.fr, gregory.zacharewicz@mines-ales.fr, nicolas.daclin@mines-ales.fr)

Abstract: Organizations are constantly looking for ways to improve the performance of their business processes (BPs) by making them more understandable, coordinated, streamlined, and effective. As a result, stakeholders are invited to take part in BP's management and improvement projects, which make use of modeling and simulation technologies. The influence of resources (human, monetary, ...) and their allocation in a business process has a significant impact on the simulation results, thus they must be portrayed accurately from the analysts' standpoint. By offering concepts, methodologies, and tools that combine business process modeling and simulation, this study bridges the gap between the two worlds. This paper presents perspectives on a metamodel-based transformation approach for Business Process Modeling and Notation (BPMN) models extended with Business Process Simulation Interchange Standard (BPSIM) into Discrete Event System Specification (DEVS) simulation models with an emphasis on resource allocation.

Keywords: Discrete event modeling and simulation, Model-Driven Engineering (MDE), Business Process Modeling and Notation (BPMN), Discrete Event System Specification (DEVS), Model Transformation, Business Process Simulation Interchange standard (BPSIM)

1. INTRODUCTION

Business processes (BPs) are essential to the daily activities of a company. They are applied and executed at three distinct levels throughout a large variety of sectors. At the descriptive level so that information can be shared and exploited, at the executable level so that BPs can be coordinated and automated, and at the analytical level to improve performance (OMG, 2013). BPs may be modeled using the Business Process Modeling and Notation (BPMN) standard which is in demand and very utilized in the corporate sector (Pasha and Pasha, 2013). It is typical practice for business users or developers to build BPMN processes using standalone apps that enable stakeholders to translate their knowledge into conceptual models that are independent of development, implementation, and environments. Modeling and Simulation (M&S) methodologies promote the development of simulation models based on conceptual models. However, this is not a current practice in most business process improvement projects, mainly because this practice necessitates the involvement of business stakeholders in the discrete event simulations while business stakeholders' may lack technical simulation skills. Complex system design can be enhanced with the use of models. Numerous significant industries utilize the Model-Driven Engineering (MDE) methodology to design complicated systems. It has demonstrated its efficacy in various business sectors, including aerospace, automobile, and telecommunications, among others (Whittle et al., 2014). The focus of this study is the Object Management Group (OMG) standard BPMN, which offers a standard notation for collaboration between business users, analysts, and implementers (OMG, 2013). BPMN is highly expressive, widely used, and may be utilized for the process design in a model-driven simulation engineering approach. Moreover, BPMN was not originally designed for simulation purposes, therefore a BPMN model is considered a high-level model that is used to make a process understandable and shareable. The Business Process Simulation Interchange standard (BPSIM), on the other hand, is a standardized specification that allows the enrichment of business models with scenario aspects having an impact on how operations are carried out. (WFMC, 2016). BPSIM aims to extend BPMN by detailing pertinent parameters and logic essential to the simulation model. In this study, we will examine BPMN extension with BPSIM Resource parameters. BPSIM resource parameters allow analysts to allocate resources explicitly which is not handled by BPMN, and which directly affects the simulation model. Once the model is extended with simulation parameters, mapping rules will be applied in order to transform it into a low-level simulation model using Discrete Event System Specification (DEVS) which allows the simulation of all types of processes independently from their application domain (Zeigler et al., 2011). This study suggests improving M&S activities with model-driven development approaches, techniques, and tools. The structure of the

10.1016/j.ifacol.2023.10.601

paper is as follows: after this brief introduction, the state of the art on business process modeling and simulation is presented in section 2. The transformation methodology is described in section 3. Section 4 concludes the paper and opens the outline.

2. STATE OF THE ART

2.1 Background

Conceptual modeling is defined as the process of abstracting a real-world problem or system (Robinson, 2010). The main purpose of conceptual modeling is to adequately represent a part of the real-world problem domain that will be the subject of a simulation study so that all possible research questions in this area can be investigated by first deriving appropriate simulation design models from the conceptual simulation model and then putting these design models into practice using one or more simulation platforms. Discrete event Simulation (DES) deals with the simulation of real-world systems and hence the use of conceptual modeling is necessary to represent discrete events in a comprehensible way. Standardizing business processes makes them easier to understand and maintain (Mens and Gorp, 2006). The capacity to interpret and orchestrate process models is provided by a variety of business process management tools that combine graphical notation and software, but the simulation capabilities of these technologies are currently quite limited (Pufahl et al., 2018). A few free and open-source business process management software are available for use by corporate stakeholders in the creation of business processes. Model simulation may be made possible by these technologies, but the modeling and simulation components have not yet been sufficiently integrated. For instance, it may be necessary to extend the model to include simulation elements (scenario parameters, resources, time, etc.). Only those who are qualified with simulation notions can use simulation because of the ongoing separation of these two worlds. The simulation of a BP model requires 2 steps: Enrichment with simulation parameters and transformation into a low-level simulation model. Once simulated, analysts would review the results of interest and improve the process. (Wagner, 2014).

2.2 BPMN enrichment

BP modeling aims to analyze the model at design time to identify bottlenecks and inconsistencies, as well as the performance of the process in terms of potential costs, resource requirements, locating and eliminating unproductive process areas, and presenting market shifts prior to their implementation. Therefore, to be able to represent a company's value chains and business operations using BPMN, a model extension is required. The model extension approach adds domain-specific concepts to the model for specific objectives. The Object Management Group identifies the BPMN metamodel's extensibility as one of its primary distinguishing features (OMG, 2013). If an extension's primary aim is to describe a domain, it is said to have a descriptive purpose. If the primary objective is to assist in the analysis of current BPMN models, then it is categorized as analytical. BPMN extensions may also be used in the purpose of assisting in the execution of the process. BPMN extension has been utilized in various contexts:

- Temporal perspective (Gagne and Trudel, 2009)
- Interoperability (Mallek et al., 2012)
- Resource performance (Lodhi et al., 2011) (Bocciarelli and D'Ambrogio, 2011)
- Risk management (Marcinkowski and Kuciapski, 2012)
- Key performance indicators (Friedenstab et al., 2012)
- Resource definition (Stroppi et al., 2015)
- Resource allocation constraints (Awad et al., 2009)
- Temporal constraints (Cheikhrouhou et al., 2013) (Pichler et al., 2017)

2.3 BPMN transformation

Model driven engineering (MDE) is a method of software engineering that promotes the development of models at various levels of abstraction, shifting the focus from programs to models at the level of development. Two of the fundamental tenets of MDE are model use, which emphasizes the notions of meta-model and compliance (Mens and Gorp, 2006) and model transformation, which covers the concepts of mappings and transformation rules. The last two decades have seen a lot of academics concentrate on the transformation of complex models into simulation models such as the Discrete Event System Specification (DEVS) formalism which has been the focus of many researchers. Based on a solid mathematical foundation Zeigler et al. (2011), DEVS is modular and hierarchical which makes it possible to build models that may be coupled to one another and to simulate them. Unlike BPMN which has one specific metamodel (OMG, 2013), DEVS does not have a standardized specific representation. Numerous DEVS metamodels were designed when moving from the theoretical model into a practical implementation necessary to replicate a DEVS model. Many applications were created for the purpose of simulating DEVS such as ADEVS, CD++, VLE, Python PDEVS, DEVS-Suite and others. Nevertheless, other DEVS metamodels were developed introducing different concepts that were found to be important such as metadata information(St-Aubin and Wainer, 2022). In MDE, both the source models and the destination models have to conform with their respective metamodels (Bézivin and Briot, 2004). Researchers presented a set of rules for transforming a conceptual source model (BPMN) into a simulation model (DEVS) for the purpose of providing a model-driven development framework for M&S. The BPMN elements differ from one another in terms of behavior and functionality. As a result, they may be termed black boxes; additionally, DEVS' behavior is transparent and hence regarded as a white box. Before they can be simulated in DEVS, the components of a BPMN flow must be mapped to their proper DEVS representation. The initial mapping concepts were introduced in (Cetinkaya et al., 2011) and were later expanded to encompass other domains, such as additional BPMN task categories (Bazoun and al., 2014) and the concept of resource allocation and failure (D'Ambrogio and Zacharewicz, 2016). These attempts, however, only incorporate a small number of BPMN concepts. Several critical components, such as message flows, interrupting events, and intermediate events, are still missing. In addi-

tion to the conversion of BPMN to DEVS, many scholars also looked at the conversion to alternative discrete event simulation models, such as Petri nets (Mutarraf and al., 2018), however DEVS was selected as it covers properly the temporal dimension (Zeigler et al., 2011). Along with a larger framework for modeling and simulation systems, DEVS also provides a formal specification of the simulator(Zacharewicz and Hamri, 2007).

3. METHODOLOGY

BPMN was initially developed to express BPs without addressing simulation. Our objective is to extend BPMN processes with the necessary simulation parameters to be able to simulate them. The Workflow Management Coalition (WFMC) has established a standard that is called BPSIM (Business Process Simulation Interchange standard). This standard defines how to parameterize BP models from multiple views to enable process simulation, analysis, and optimization (WFMC, 2016). As shown in Fig. 1 and based on the MDE principles, our objective is to transform BPMN models which are extended with BPSIM into DEVS models that can be operated or simulated to identify issues and improve the business processes. The extended BPMN model (BPMN+BPSIM) is consequently transformed into a corresponding DEVS model according to a predetermined set of mapping and transformation rules. The transformation mapping also takes into account DEVS models that we have already developed in DEVS-Suite. We, therefore, propose to provide process modeling tools for two profiles of users as shown in Fig. 5: a business modeler whose responsibility is the modeling and enrichment of the process and an Advanced M&S Expert whose job is to manage the DEVS bank and the library of the mapping rules of BPMN+BPSIM to DEVS. The business modeler will also be reviewing the end results of the simulation in order to make changes to the business process when necessary. In our methodology, we consider that the DEVS bank contains models that are already implemented in DEVS-Suite and therefore these models may be used as part of a mapping rule. For instance, the basic resource model designed in (Zacharewicz et al., 2008) and the resource with behavior model represented in (D'Ambrogio and Zacharewicz, 2016) have been developed in DEVS-Suite and are considered as part of our initial load of the DEVS models Bank.

3.1 Resource allocation in BPSIM

Resource allocation has a great impact on tasks. In this study, we focus on the extension of BPMN using BPSIM's resource parameters. Given that the resource perspective in BPMN is limited, a process modeler usually represents a resource with a swimlane to determine which resource is responsible for the activity. Moreover, this technique does not indicate whether the needed resource is available at the time the activity is scheduled to be executed or which resource should be assigned when several relevant resources are available (Goel and Lin, 2021). BPSIM standard allows us to parameterize resource allocation in BPMN as it recognizes resource parameters and describes their criteria in terms of availability, quantity, selection, and roles. Expression parameters may also be parameterized at the

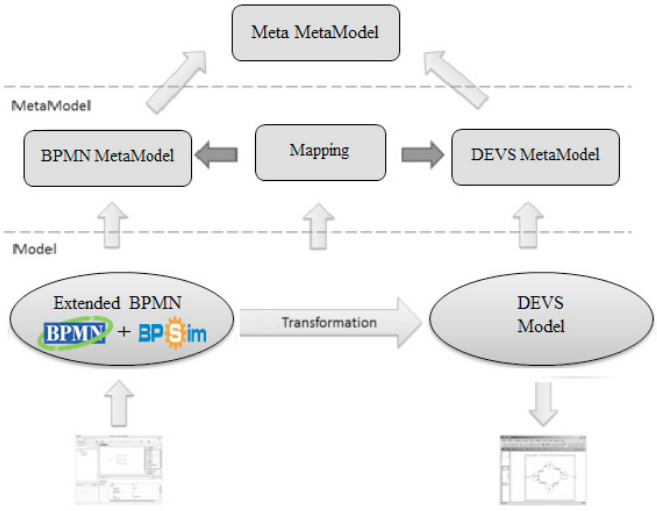

Fig. 1. Extension and Transformation architecture (El Kassis et al., 2022)

resource level in the form of explicit values, operators, and functions allowing runtime computations of values. The following BPSIM predefined functions are used in order to allocate resources based on resource role, name and quantities which are defined at the BPSIM resource level (WFMC, 2016). With these functions, we are able to identify 4 allocation patterns (Russell et al., 2005) that may be implemented as described in Table 1.

Table 1. Allocation patterns for BPSIM

Allocation type	BPSIM method
Direct Allocation	bpsim:getResource(name, qty)
Role-Based Allocation	bpsim:getResourceByRoles ([role,...],qty)
Deferred Allocation	bpsim:getResource(bpsim: getProperty(name), qty)
Escalation	bpsim:orResource([resources,...])

- The Direct Allocation is used whenever a resource is allocated to a defined task
- The Role-Based Allocation is used when the selection of the resources is based on a defined role
- The Deferred Allocation is used when the allocation of the resource is dynamic and where the selected resource is defined at simulation time
- Allocation by Escalation is used whenever a task may use a set of alternative resources

3.2 Transformation to DEVS

We intend to simulate the resulting BPMN+BPSIM model with DEVS-Suite simulator, therefore the enriched process model must be transformed into JAVA classes and methods. The extended model resulting from the enrichment of BPMN with BPSIM will be transformed into a DEVS coupled model. Both the source and destination metamodels are needed for this transformation, as well as a language for defining the transformation rules. Separate metamodels for BPMN and BPSIM are defined in the BPMN 2.0 OMG standard (OMG, 2013) and the BPSIM 2.0 WFMC specification (WFMC, 2016). First, we need to combine the BPMN and BPSIM metamodels to create

a new BPMN+BPSIM metamodel. A specific language must be established to implement the transformation rules between the source and target models. The definition and execution of transformation rules would be validated by test cases. Extensible Stylesheet Language Transformation (XSLT) and Atlas Transformation Language (ATL) were two of the transformation languages that were suggested for the model to model transformation from BPMN to DEVS (Cetinkaya et al., 2011) (Bazoun and al., 2014). In this paper, we have chosen to express the transformation rules using XSLT. Once a BPMN model is enriched with BPSIM parameters, the resulting XMI file is converted to XML using XSLT, and then the XML is processed in JAVA where it undergoes several iterations that aim to compare the XML with a defined mapping library. The DEVS models will then be created accordingly as JAVA code in order to be imported later on under DEVS-Suite simulator where they may be visualized and simulated. All new DEVS models are added to the DEVS models bank and may be used later on when adding new mappings. The generation and import of the JAVA classes into DEVS-Suite was also proposed in (Alshareef and Zeigler, 2020) as part of a transformation from Activity Diagram to DEVS, however, a user intervention was required to add simulation parameters, which in our case should be parameterized in the first stage using BPSIM which strives to extend the model with different aspects such as resources and time. Our objective is to provide a library of mapping for BPMN+BPSIM source and DEVS destination.

Table 2. Mappings to DEVS

BPMN+BPSIM	DEVS
BPMN Task with Time parameter	Atomic DEVS model (Duration reflected in time advance function)
Resource Parameter with Selection attribute on one resource	Atomic DEVS representing basic resource extracted from the DEVS bank
BPMN task + Resource Parameter with Selection attribute on one resource	Atomic DEVS of the task coupled with that of the Resource
bpsim:orResource bpsim:getResourceByRoles	Resource Broker coupled with the list of resources resulting from the resource expression
BPMN task +Resource Parameter with Selection attribute set as either bpsim:orResource or bpsim:getResourceByRoles	The task is transformed to atomic DEVS model coupled with the resulting DEVS of the bpsim:orResource

Based on Table 1, we could identify the necessity to create a resource allocator (broker) in DEVS to manage the resource allocation. The broker will be responsible of the related group of resources that are either gathered under same BPSIM Role or linked statically inside a bpsim:orResource. A DEVS broker is to be implemented in DEVS-suite and hence it becomes part of the DEVS models bank. Using MDE and pattern analysis, we will be able to identify DEVS model to be coupled with the task in question which is either of type resource or a broker. As a rule, we initially start by analyzing the resources and then

```
<ElementParameters elementRef="ResourceR">
 <ResourceParameters>
  <Quantity>
     <NumbericParameter value="1"/>
  </Quantity>
 </ResourceParameters>
</ElementParameters>
<ElementParameters elementRef="TaskT">
 <TimeParameters>
  <ProcessingTime>
   <DurationParameter value="PT"/>
   </ProcessingTime>
 </TimeParameters>
 <ResourceParameters>
  <Selection>
     <ExpressionParameter
     value = "getResource(ResourceR,1)"/>
  </Selection>
 </ResourceParameters>
</ElementParameters>
```

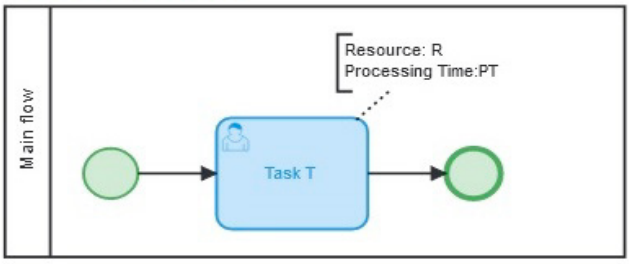

Fig. 2. BPMN Task Annotated with BPISM Resource and time

we create the brokers and their corresponding couplings. Table 2 shows the related mappings.

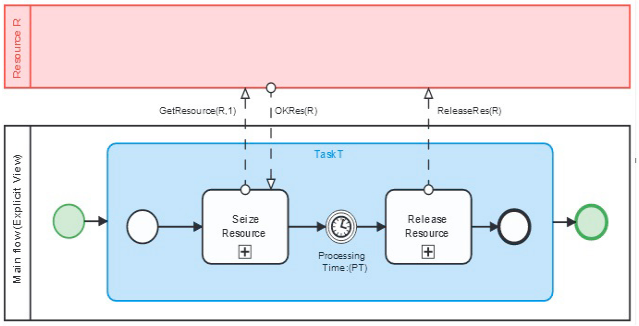

Fig. 3. BPMN Task Extended with Resource and time

Following the seize-release mechanism, the model of the task may be transformed in a BPMN only manner as illustrated in Fig. 3 by explicating the seize and release of the resource.

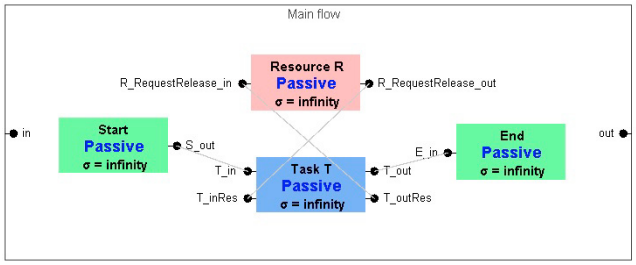

Fig. 4. Coupled DEVS model corresponding to Fig. 2

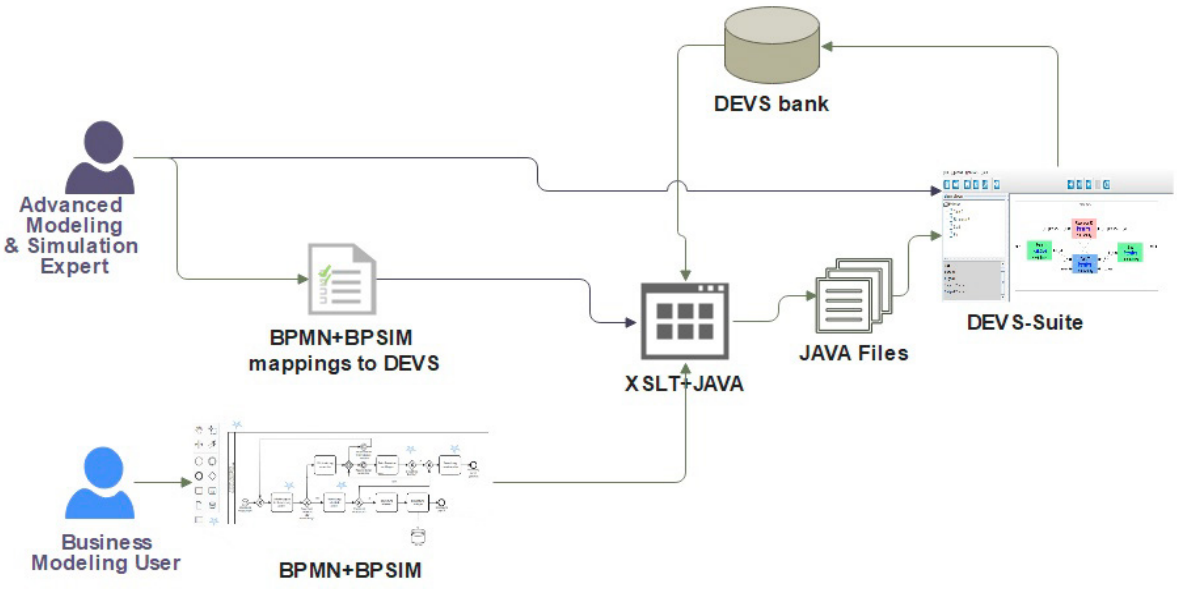

Fig. 5. Application flow

The resource parameters that are linked to a task affect the behavior their corresponding representation in DEVS. We will consider that the resources are managed following a seize-release mechanism. Fig. 2 illustrates a simple BPMN flow consisting of a start and an end event and a basic BPMN task TaskT. BPSIM extends this process by first creating a new element parameter with a reference RessourceR and a quantity. The resource is considered global however the allocation of this request to task is established using the Resource Parameter selection element. We differentiate in this example 3 main elements: the task, the resource, and the resource to task allocation.

The explicit BPMN group in 3 is to be represented by its equivalent DEVS atomic model which translates the seize and release tasks into states and where the time is handled with DEVS' time advance function. We interpret the seize and release tasks as subprocesses, horeover, more detailed models are implemented in (Goel and Lin, 2021) which elaborates more on the behavior of the seize and release. Fig. 4 illustrates the DEVS target model that was produced following the transformation of the explicit BPMN model. The target model consists of 4 atomic DEVS. The Start and End DEVS models which represent the Start Event and End Event of BPMN. The Resource and Task DEVS models are reusing atomic models defined in (D'Ambrogio and Zacharewicz, 2016). These models have been simulated and integrated in more complex models. The use of resource brokers is not presented in this paper but is under study to be integrated in the platform. As well BPMN XOR and AND gateways have been developed in the DEVS models bank.

4. CONCLUSION AND FUTURE WORKS

This article describes a model-driven approach for building DEVS-based simulation models from BPMN models that have been extended using BPSIM with concentration on BPSIM resource parameters. Business analysts are given the ability to specify the optimal distribution of task resources using BPSIM. The extended model is then utilized as an input for a model transformation, which eventually leads to the development of the matching DEVS model as an output. Once the target DEVS model is produced, a DEVS simulator, such as DEVS-Suite, is used for simulation. Mappings of BPMN and BPSIM components to DEVS atomic models were presented. The logic of the process control flow is applied to the coupling of these DEVS atomic models so that the final DEVS coupled model may be created. This study reviewed the literature on BPMN and business model simulation. Even though it shows that simulation tools exist, most BPMN-compatible business processes simulation programs are proprietary (black boxes). For future work, our objective is to offer a free, open-source web platform for business process modeling and simulation using BPMN for modeling and DEVS for simulation with the addition of BPSIM standards for distributed simulations. The BPSIM simulation standard will be added to BPMN, and DEVS will be utilized for operational simulation. The definition of a BPMN + BPSIM metamodel, the definition of a general DEVS metamodel, and the expression of transformation rules from (BPMN + BPSIM) to DEVS are the three main focuses of our work. The platform that results from this is meant to be open source (white box), incremental, and easy to customize so that the community can contribute to future developments.

REFERENCES

Alshareef, A. and Zeigler, B. (2020). Integration of activity specification into devs modeling & simulation development environment. 46–55. doi: 10.5753/mssis.2020.12494.

Awad, A., Grosskopf, A., Meyer, A., and Weske, M. (2009). Enabling Resource Assignment Constraints in BPMN. 16.

Bazoun, H. and al. (2014). Business Process Simulation: Transformation of BPMN 2.0 to DEVS Models. In *SCS/ACM/IEEE Symposium*

on Theory of Modeling and Simulation part of SpringSim 2014. Tampa, United States. URL https://hal.archives-ouvertes.fr/hal-00990758.

Bézivin, J. and Briot, J.P. (2004). Sur les principes de base de l'ingénierie des modèles. *Obj. Logiciel Base données Réseaux*, 10(4), 145–157.

Bocciarelli, P. and D'Ambrogio, A. (2011). A bpmn extension for modeling non functional properties of business processes. In *Proceedings of the 2011 Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*, 160–168.

Cetinkaya, D., Verbraeck, A., and Seck, M. (2011). MDD4MS: A model driven development framework for modeling and simulation. In *Proceedings of the 2011 Summer Computer Simulation Conference*, 113–121.

Cheikhrouhou, S., Kallel, S., Guermouche, N., and Jmaiel, M. (2013). Toward a time-centric modeling of business processes in bpmn 2.0. In *Proceedings of International Conference on Information Integration and Web-Based Applications & Services*, IIWAS '13, 154–163. Association for Computing Machinery, New York, NY, USA. doi:10.1145/2539150.2539182. URL https://doi.org/10.1145/2539150.2539182.

D'Ambrogio, A. and Zacharewicz, G. (2016). Resource-based modeling and simulation of business processes. In *SCSC '16. Summer Computer Simulation Conference*. Montreal, Canada. URL https://hal.archives-ouvertes.fr/hal-02465026.

El Kassis, M., Trousset, F., Daclin, N., and Zacharewicz, G. (2022). Business Process Web-based Platform for Multi modeling and Distributed Simulation. In *I3M 2022 - International Multidisciplinary Modeling & Simulation Multiconference*. Rome, Italy. URL https://hal.archives-ouvertes.fr/hal-03822888.

Friedenstab, J.P., Janiesch, C., Matznerand, M., and Müller, O. (2012). Extending BPMN for Business Activity Monitoring. *Proceedings of the Annual Hawaii International Conference on System Sciences*, 4158–4167. doi:10.1109/HICSS.2012.276.

Gagne, D. and Trudel, A. (2009). Time-bpmn. In *2009 IEEE Conference on Commerce and Enterprise Computing*, 361–367. doi:10.1109/CEC.2009.71.

Goel, A. and Lin, M.B. (2021). Resource requirements in business process modelling from an operations management perspective. *SSRN Electronic Journal*. doi:10.2139/ssrn.3869592.

Lodhi, A., Köppen, V., and Saake, G. (2011). An extension of bpmn meta-model for evaluation of business processes. *J. Riga Technical University*, 43, 27–34. doi:10.2478/v10143-011-0004-7.

Mallek, S., Daclin, N., and Chapurlat, V. (2012). The application of interoperability requirement specification and verification to collaborative processes in industry. *Computers in Industry*, to be appear. URL https://hal.archives-ouvertes.fr/hal-00804253. Publisher: Elsevier.

Marcinkowski, B. and Kuciapski, M. (2012). A business process modeling notation extension for risk handling. In *IFIP International conference on computer information systems and industrial management*, 374–381. Springer.

Mens, T. and Gorp, P.V. (2006). A Taxonomy of Model Transformation. *Electronic Notes in Theoretical Computer Science*, 152, 125–142. doi:https://doi.org/10.1016/j.entcs.2005.10.021.

Mutarraf, U. and al. (2018). Transformation of Business Process Model and Notation models onto Petri nets and their analysis. *Advances in Mechanical Engineering*, 10(12). doi:10.1177/1687814018808170. URL https://hal.archives-ouvertes.fr/hal-02475788.

OMG (2013). *Business Process Model and Notation (BPMN) 2.0.2*. URL http://www.omg.org/spec/BPMN.

Pasha, M.A.u.r. and Pasha, S. (2013). Bloom's Taxonomy for Standardizing BPM Education of IT Under-Graduates Students. *Journal of Computer Applications*, 6–13.

Pichler, H., Eder, J., and Ciglic, M. (2017). Modelling Processes with Time-Dependent Control Structures. In H.C. Mayr, G. Guizzardi, H. Ma, and O. Pastor (eds.), *Conceptual Modeling*, 50–58. Springer International Publishing, Cham.

Pufahl, L., Wong, T.Y., and Weske, M. (2018). Design of an Extensible BPMN Process Simulator. In E. Teniente and M. Weidlich (eds.), *Business Process Management Workshops*, 782–795. Springer International Publishing.

Robinson, S. (2010). Conceptual modeling for simulation: definition and requirements. In *Conceptual modeling for discrete-event simulation*, 19–46. CRC Press.

Russell, N., Ter, A., Edmond, D., and Aalst, W. (2005). Workflow resource patterns.

St-Aubin, B. and Wainer, G. (2022). Metadata specification for devs simulation models. In *2022 IEEE/ACM 26th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, 160–163. IEEE.

Stroppi, L.J.R., Chiotti, O., and Villarreal, P.D. (2015). Defining the resource perspective in the development of processes-aware information systems. *Information and Software Technology*, 59, 86–108. doi:10.1016/j.infsof.2014.10.006.

Wagner, G. (2014). Tutorial: Information and process modeling for simulation. In *Proceedings of the Winter Simulation Conference 2014*, 103–117. doi:10.1109/WSC.2014.7019882.

WFMC (2016). *Business Process Simulation Specification (BPSIM)*. URL https://www.bpsim.org/specifications/2.0/WFMC-BPSWG-2016-01.pdf.

Whittle, J., Hutchinson, J., and Rouncefield, M. (2014). The State of Practice in Model-Driven Engineering. *IEEE Software*, 31(3), 79–85. doi:10.1109/MS.2013.65.

Zacharewicz, G., Frydman, C., and Giambiasi, N. (2008). G-DEVS / HLA Environment for Distributed Simulations of Workflows. *SIMULATION*, 84(5), 197–213. doi:10.1177/0037549708092833. URL https://hal.archives-ouvertes.fr/hal-00204203.

Zacharewicz, G. and Hamri, M.E.A. (2007). Flattening G-DEVS / HLA structure for Distributed Simulation of Workflows. In *AIS-CMS International modeling and simulation multiconference*, 11–16. Buenos Aires, Argentina. URL https://hal.archives-ouvertes.fr/hal-00173659.

Zeigler , B.P., Praehofer, H., and Kim, T.G. (2011). *Theory of Modeling and Simulation*. Elsevier, Academic Press, second edition edition.