







Adding Dynamic Simulation to Business Process Modeling via System Dynamics

H. A. Proper¹, Q. Zhu², J. P. P. Ravesteijn², and W. Gielingh³

¹ TU Wien, Vienna, Austria

henderik.proper@tuwien.ac.at

² HU University of Applied Sciences Utrecht, Heidelberglaan 15,
3584CS Utrecht, The Netherlands

quan.zhu@hu.nl, pascal.ravesteijn@hu.nl

³ Real Capital Concepts B.V., Tijgervinder 26, 3863GP Nijkerk, The Netherlands
wim.gielingh@circularbuilding.com

Abstract. Business process modeling and system dynamics are different approaches that are used in the design and management of organizations. Both approaches are concerned with the processes in, and around, organizations with the aim to identify, design and understand their behavior as well as potential improvements. At the same time, these approaches differ considerably in their methodological focus. While business process modeling specifically takes the (control flow of) business processes as its primary focus, system dynamics takes the analysis of complex and multi-faceted systems as its core focus. More explicitly combining both approaches has the potential to better model and analyze (by way of simulation) complex business processes, while specifically also including more relevant facets from the environment of these business processes. Furthermore, the inherent ability for simulation of system dynamics models, can be used to simulate the behavior of processes over time, while also putting business processes in a broader multi-faceted context. In this paper, we report on initial results on making such a more explicit combination of business process modeling and system dynamics. In doing so, we also provide a step-by-step guide on how to use BPMN based models and system dynamics models together to model and analyze complex business processes, while illustrating this in terms of a case study on the maintenance of building facades.

Keywords: System Dynamics · BPMN · Facade Maintenance

1 Introduction

Business process models play a central role in the design and analysis of business processes. As its name suggests, business process modeling specifically takes the (control flow of) business processes as its primary focus. This does not mean that other aspects are not considered, but rather that the (control flow of) business processes are taken as the primary focus. Many different languages are used

to model business processes, where some modeling languages are more generic, while others are more explicitly dedicated to the task of modeling business processes. Examples include: Petri Nets [2], UML's - Activity Diagrams (UML-AD) [7], ArchiMate's business layer [9, 12], and the Business Process Modeling Notation (BPMN) [5, 14].

While business process modeling specifically takes the business processes as its primary focus, System Dynamics (SD) [20], as a method, takes the analysis of complex and multi-faceted systems as its core focus with the aim of providing advanced simulation models of the complex and multi-faceted dynamics of such systems. As such, both approaches can contribute to the design and management of processes in an organizational context. At the same time, the fundamental difference in the primary focus of these two approaches points at a potential benefit of explicitly combining the two approaches. More specifically, such an explicit combination would have the potential benefit of better modeling and analyzing complex business processes, while also including other relevant (dynamic) facets from the environment of these business processes. Examples of the latter facets include energy consumption, produced waste, workload on workers, usage of equipment, etc. Even more, the inherent ability of SD models to provide simulations of complex dynamics, can be used to simulate the behavior of the different involved processes (covering the business process as well as other relevant dynamic systemic facets).

The authors of [3] also argue, in a broad sense, that existing model-driven development related modeling languages, such as BPMN, need to be complemented with dynamic simulation. The authors of [16] and [10] second this by more specifically arguing the need for a complementary (SD-based) perspective of SD next to business process modeling notation in terms of hybrid process simulations across different levels of detail.

The potential relationship between business process modeling and system dynamics has been touched upon before in [16–19, 21, 22], where the focus has essentially remained on the control flow aspects only. The work as reported in [19, 21, 22] pioneered the joint grounding of (high-level) SD models and (more specific) business process models on top of a general purpose domain modeling language, with the aim to produce higher quality (in terms of their conceptual fidelity) SD and business process models [22, Fig. 1].

In this paper, we report on initial results towards a more explicit, and multi-faceted, combination of business process modeling and SD. More specifically, we will provide guidelines on how to use BPMN based models and SD models together to model and analyze complex business processes. These guidelines will be illustrated in terms of a case study in the context of maintenance of the facades of building, as part of a project in the Dutch construction sector aiming to improve the recycling of aluminum as used in building facades. This case will also illustrate how BPMN diagrams can be used as the starting point for creating SD models, as they provide a clear and comprehensive picture of the sequence of activities and events in a process. The resulting SD models, in turn, can then be used to simulate the behavior of the business process over time,

while relevant additional (dynamic) facets of the operational environment of the business process can be included as well.

In line with this, the remainder of this paper is structured as follows. In Sect. 2 we provide a short introduction to SD. In Sect. 3 we then briefly discuss BPMN, while also more clearly differentiating between BPMN and SD regarding the abilities for (complex) simulations. Section 4 then elaborates the idea of using BPMN in concert with SD in terms of a mapping and associated guidelines. Before concluding, we illustrate these in terms of the case regarding the maintenance of facades in Sect. 5.

The mapping as discussed in Sect. 4 is actually a first (humble) iteration of a design science cycle [23] towards the design of a more complete integration between business process modeling and system dynamics. The case as used in Sect. 5 currently serves both as a case for the initial evaluation (towards the design cycle), as well as to identify ‘use cases’ (towards the relevance cycle) for such an integration and resulting simulation potential.

2 System Dynamics Modeling and Simulation

System Dynamics (SD) was originally developed by Jay W. Forrester at MIT Sloan in 1956 to study the behavior of systems. Presently, SD is used primarily to analyze complex systems by means of simulations [20]. It involves the construction of a model of a system in terms of feedback loops and other causal relationships, and then simulating the behavior of the system over time to gain insights into its dynamics and performance. SD can be used to understand how different facets of a system interact and how changes in one area may impact other areas. SD uses two diagram types to capture SD models: Causal Loop Diagrams (CLDs) and Stock-and-Flow Diagrams (SFDs). CLDs show the main *variables*, the system *boundaries* and the overall *structure* of the SD model. In the process of analyzing a system, these diagrams help to scope the system, to quickly capture hypotheses about the causes of dynamics, elicit and capture the mental models of individuals or teams, and communicate important feedback loops. CLDs show how systemic variables influence each other in terms of a qualitative (positive or negative) influence. However, they do not operationalize this in quantitative terms. Therefore, SFDs complement CLDs in terms of stocks and flows of quantitative accumulations of ‘things’ (materials, value, information, tasks, CO2 emissions, energy, etc) as they ‘move’ through a system [20].

3 BPMN and Simulation

BPMN [14] provides a graphical notation to create diagrams that show the flow of activities, decisions, and interactions between different actors. As stated in [14], the main goal of BPMN is to provide a notation that is understandable to all stakeholders involved in the design and analysis of organizational processes such as managers, business analysts, information managers, software developers, and end-users. As BPMN is often used in process improvement initiatives, all these

roles are involved in identifying bottlenecks and inefficiencies in a process, while subsequently creating and implementing solutions.

Simulation is often used to help managers and analysts in understanding different solutions and deciding on which to implement. With the aim of assessing the effects of changes made to the processes and/or physical settings (e.g., the ability of resources to perform tasks), without disrupting current operations, simulation is a technique that may be used to understand the behavior of a system [10]. While there are BPM-tool suppliers that offer simulation functionality based on BPMN, Pereira and Freitas [15] found that, to be able to do so, specific elements for simulation needs to be incorporated, as BPMN is not designed for simulation as such. Based on a study into common simulation properties [4] and an analysis of 5 BPM-tools with simulation capabilities, several properties were found to be lacking [15]. None of the analyzed tools had the possibility to define periods of unavailability of resources. Also, functionality for defining transfer time, interruptions and execution priorities were only available in one tool (i.e., BPSim – Trisotech BPMN 2.0 Modeler for Visio version 4.2.0). Finally, the possibility to define an allocation plan for sharing of execution activities as well as stipulating the number of replications of the simulation were only provided by two of the five tools. Still, whether properties were found to be present or not does not say anything about the level of support by the tool, therefore these outcomes can only be taken as a first indication regarding the simulation capabilities and use of BPMN.


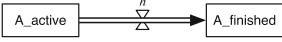
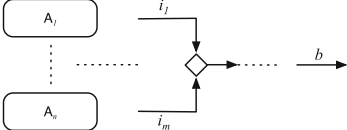
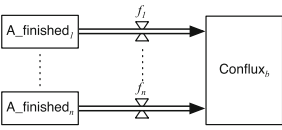
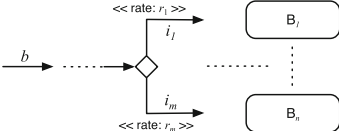
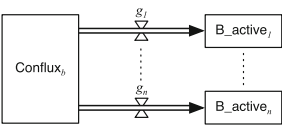
4 System Dynamics and Process Flows

SD is often used for high-level organizational analysis, such as strategy development and analysis of policy options, where capturing information flow and feedback are essential considerations [10]. Simulation models that are based directly on BPMN can only perform analyses and decision making at an operational (discrete) level, c.f. [13]. SD, however, can be used on a more abstract level than BPMN in terms of a more continuous approach to everything in the process. For instance, for managers with the responsibility to predict future resource needs and anticipating what various options might cost their organizations, SD offers more than conventional (business process) flow modeling [8]. A conventional flow plan and some simulation in any of the conventional process modeling analysis tools will identify many problems facing organizations. However, a conventional flow plan would not provide much insight into a supply chain problem in which shipping costs, resource costs and personnel costs are all varying in different ways, each affecting the other. SD, on the other hand, is very useful for its feedback options that can influence various related parameters of the process like learning and improvement during the process, communication overhead, error rates and even increasing experience of human resources [11].

In principle, BPMN and SD share a common logic, i.e. sequential progression of activities from start to finish [10]. BPMN's flow-and-gate based notation are conceptually similar to SD's stocks and flows. Based on this, Table 1 provides three key mapping patterns to translate (the control flow of) BPMN to SFD.

The mapping as shown in Table 1 provides the *requirements* on the valves and flows as identified on the system dynamics side. We are currently investigating the most effective way (in terms of representation and simulations) to represent the semantics of the different BPMN gateways in terms of system dynamics.

Table 1. BPMN to SD mapping

BPMN	System Dynamics (SFD)
 <p style="text-align: center;">$T \geq 0$</p>	 <p style="text-align: center;">$h(t) \triangleq \text{Inflow}(A_active, t - T)$</p>
 <p style="text-align: center;">no specific pre-conditions</p>	 <p>base: for each $i \in \{1, \dots, n\}$: $O_i(t) := \text{Stock}(A_finished_j, t)$ $R_i := 1$</p> <p>recuse: if AND-split, then for each $i \in \{i_1, \dots, i_m\}$: $O'_i(t) := \text{Min}_{j \in \{i_1, \dots, i_m\}} O_j(t)$ $R'_i := R_i / m$ if (X)OR-split, then for each $i \in \{i_1, \dots, i_m\}$: $O'_i(t) := O_i(t)$ $R'_i := R_i$</p> <p>final: for each $i \in \{1, \dots, n\}$: $f_i(t) \triangleq O_i(t) \times R_i$</p>
 <p>if AND-split: $\forall_{1 \leq i \leq m} [r_i = 1]$</p> <p>if OR-split: $\forall_{1 \leq i \leq m} [r_i > 0] \wedge \sum_{i \leq i \leq m} r_i \leq m$</p> <p>if XOR-split: $\forall_{1 \leq i \leq m} [r_i > 0] \wedge \sum_{i \leq i \leq m} r_i = 1$</p>	 <p>base: for each $i \in \{1, \dots, n\}$: $R_i := 1$</p> <p>recuse: for each $i \in \{i_1, \dots, i_m\}$: $R'_i := R_i \times r_i$</p> <p>final: for each $i \in \{1, \dots, n\}$: $g_i(t) \triangleq \text{Stock}(\text{Conflux}_b, t) \times R_i$</p>

The first row in Table 1 shows the mapping of a BPMN activity A to an SFD. As an activity would require a time $T \geq 0$ to finish after it is started, the mapping to an SFD needs to take this into account. Since such a mapping from BPMN to SFD requires this delay to be known, we have added it also to the BPMN side in terms of (a slight ‘abuse’ of) the stereotype notation: <<delay : T>>. On the SFD side, we have to create two stocks for activity A. One representing the instances of A that are **active**, and one for those that are **finished**. The flow between

them is controlled by a valve h , which depends on the inflow of A_active in the sense that it releases, at point in time t , the entire inflow from $t - T$. The situation as shown in the first row is actually the basic situation in which there is a fixed waiting time of T for activity A to finish. More generally, however, one could also consider T as a probability distribution function over time, signifying how long it would take to complete an instance of task A , such that:

$$\int_0^\infty T(t) dt = 1$$

For a fixed delay, we would then have $\exists!_t [T(t) = 1]$, where that (unique) t is the fixed delay. For this generalized approach, we could define h (when using a continuous time axis) as:

$$h(t) \triangleq \int_{-\infty}^t \text{Inflow}(A_active, t - u) \times T(u) du$$

We take the view that the control flows between activities in BPMN essentially involve two kinds of trees: a *join-tree* with zero or more join-gateways, and a *split-tree* with zero or more split-gateways. In between a combination of such trees there will be one trigger b that bridges between the two trees. In the mapping to an SFD, the latter trigger leads to a stock we refer to as the Conflux_b . This is illustrated at the top of the second and third rows of Table 1. In the case of a join-tree with sources A_1, \dots, A_n , for some n , we have the SDF pattern with $A_finished_i$ stocks and a flow f_i to Conflux_b . In the case of a split-tree with targets B_1, \dots, B_n , for some (other) n , we have the SDF pattern right with A_active_i stocks and flows g_i from Conflux_b .

The lower parts of rows two and three define requirements for the flows contained in the SFD. For a join-tree (second row) we use a recursive definition following the structure of a join tree, involving O_i which defines the *outflow* from A_i into f_i at point in time t and ratio R_i with which the *outflow* from A_i is translated to an *inflow* for Conflux_b . The actual outflow O_i depends on the combination of the different join gateways from the sources to trigger b bridging between the join-tree and split-tree. Depending on the join gateway, the outflow and inflow ratio need to be computed differently.

For the split-tree (third row) we also use a recursive definition. In this case, this only involves the rate R_i in which the stock from Conflux_b is turned into an inflow of the target stocks. This rate reflects the division of the stock of Conflux_b based on the probabilities associated to the different options of the split-gateways. On the left, we see the conditions of on these probabilities (r_i) depending on the kind of split-gateway.

In the case of a join-tree without any gateway, i.e. $n = 1$, we can optimize the resulting SFD by merging f_1 and Conflux_b into $A_finished_1$. Similarly, when the split-tree does involve gateways, but the join-tree does not, we can merge Conflux_b , g_1 and B_active_1 .

It is important to stress that the mapping of the control-flow from BPMN to SFD as shown in Table 1 is only intended to provide the ‘backbone’ of an

SD model, which can then be further complemented with other facets such as energy consumption, produced waste, workload on workers, usage of equipment, etc. However, for the flows related to the consumption/production of such facets, the flow(s) through the business activities (as based from the original BPMN model) will be the main driver.

As an integrated procedure to create a BPMN and SD model together, to analyze complex business processes in conjunction with all relevant facets, we propose the step-by-step guide as discussed in the remainder of this section. This guide extends the work of [10], in particular in step 3 (i.e., the way of mapping) and steps 6 to 8 (i.e., new steps). The guide is visualized in Fig. 1.

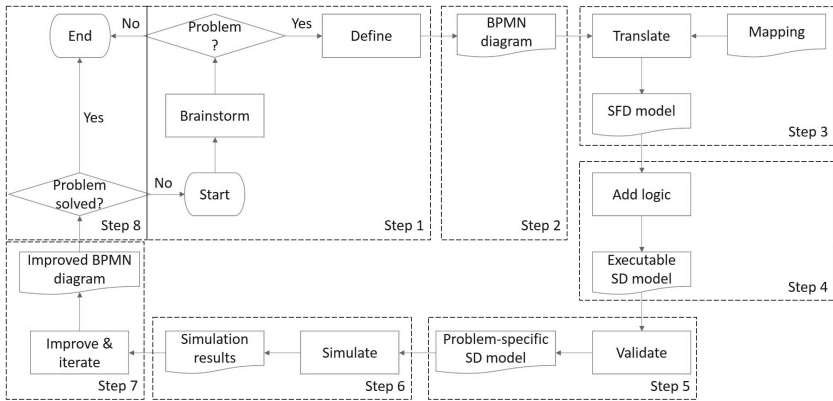


Fig. 1. Visualized step-by-step guide

Step 1: Define the scope and objectives of the process. Identify the key stakeholders, inputs, and outputs of the process, and define the performance metrics to evaluate the process.

Step 2: Create a BPMN diagram of the process using the standard notation by modeling the activities, decisions, and events involved in the process.

Step 3: Translate the BPMN model to the SFD via the mapping from Table 1.

Step 4: Add model logic and variables to the SFD to enable its execution as an SD model.

Step 5: Validate the SD model by comparing it to real-world data or historical performance data for the process. Ensure that the SD model accurately reflects the behavior of the process, and that it is consistent with the BPMN diagram. Note: in practice, this step should draw upon results from e.g. process mining in general [1], and process mining towards SD models in particular [18].

Step 6: Use the SD model to simulate the behavior of the process over time. Test different scenarios and strategies, and observe how the process responds to changes in key variables. Use the BPMN diagram to help interpret the results of simulations, and to identify opportunities for improvement.

Step 7: Use the insights gained from simulations to identify areas for improvement in the process. Modify the BPMN diagram as necessary to incorporate these changes, and test them using SD simulations. Iterate this process until identifying the optimal configuration for the process.

Step 8: Communicate the results of analysis to stakeholders using the BPMN diagram and other visualizations. Explain how the insights gained from simulations can be used to improve the performance of the process, and make recommendations for future improvements.

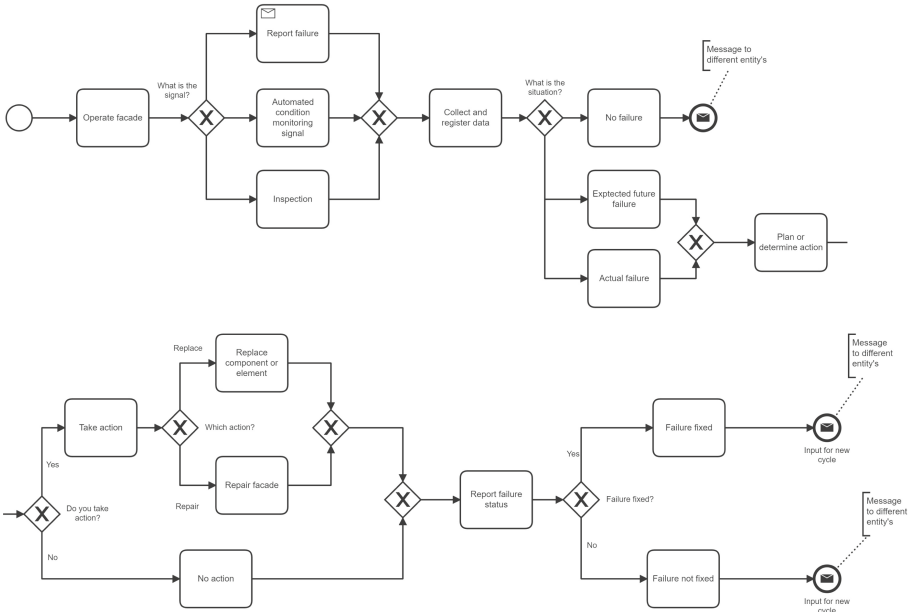


Fig. 2. Adjusted BPMN diagram for facade maintenance

5 The Facade Maintenance Case

The application of our proposed eight-step guide is illustrated by a case study from the project PerpetuAI, which is in the context of circular Aluminum chain. The aim of our case study is to demonstrate the management of facade panels of buildings. Given the confidentiality agreement, although our model is based on a real case, the presented structure and data is not case specific. The software to build our BPMN model was Camunda, while the software for our SD model was Vensim PLE.

Step 1: A group of four researchers visited our focal firm and interviewed two customer service (maintenance) managers to identify inputs and outputs

of facade construction and maintenance processes. The initial BPMN diagram was then created.

Step 2: Another researcher with multiple years of experience with BPMN modeling in practice then checked and adjusted the initial BPMN diagram, according to the standard notation. The adjusted BPMN diagram was then validated by the fourth Author of this paper. The adjusted BPMN diagram is shown in Fig. 2.

Step 3: The adjusted BPMN diagram was then translated according to the mapping provided in Table 1 by the second author of this paper. The resulting SFD is shown in Fig. 3.

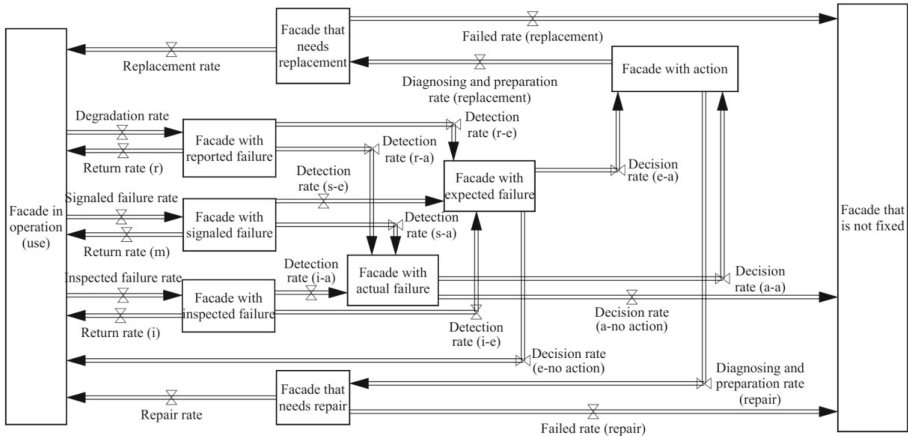


Fig. 3. Translated stock-and-flow diagram for facade maintenance

Step 4: Besides stocks and flows, other variables (mostly related to time/delay) and relationships (i.e. arrows) were added to the SD model (Fig. 4) to enable the simulation runs.

Step 5: Authors two and four acquired historical performance data from above-mentioned customer service (maintenance) managers and validated the SD model.

Step 6: Four scenarios were simulated. The base scenario, Scenario 1, is corrective maintenance (CM), which only repairs facades after failure. Scenario 2 is to combine CM with time-based maintenance (TBM). Scenario 3 is to combine corrective maintenance with condition-based maintenance (CBM). While Scenario 4, i.e. the full model, is to combine all three maintenance practices. The scenario simulation results are summarized in Fig. 5. Please note that “Dmnl” is the short term of “dimensionless”.

Step 7 and 8: will be done in the near future. Based on the scenario simulation results (Fig. 5), Scenarios 2 and 3 will be further checked with investment/cost data to calculate whether the investment on TBM or CBM can break even within an acceptable period (e.g., two years).

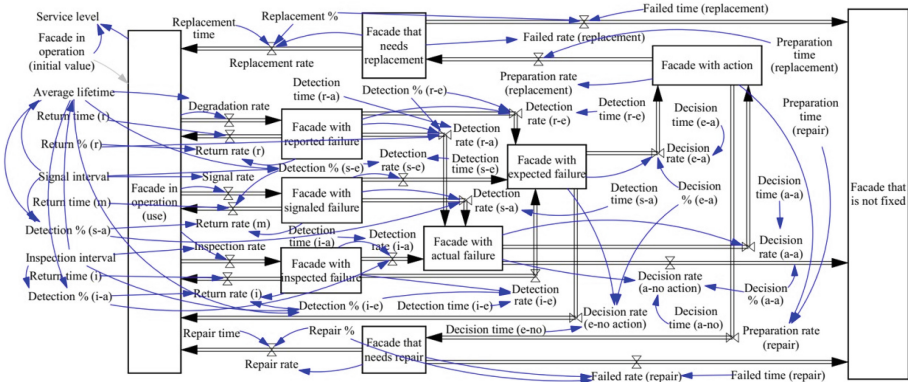


Fig. 4. System dynamics model for facade maintenance

The above reported case study is a part of the PerpetuAL project, which is a feasibility study for closing the circular material loop of aluminum facades. It was decided, within the project, to use BPMN for the AS-IS analysis of the existing processes, as well as for the design of the TO-BE optimized processes. The fact that BPMN enables easy process automation played an important role in this decision. Another factor that influenced this decision was the possibility to use BPMN models for process simulation. However, the standard BPMN tools do not allow us to append the models with user defined process parameters. In such a context, SD is more suited for process simulation. As reported in [8], it can be a valuable tool for greening the company’s processes, such as the modeling of facade wear, maintenance and repair, as also illustrated in the case study above. Predictive maintenance schemes (i.e., TBM and CBM) could replace traditional maintenance schedules (i.e., CM), anticipating the presence of a failure based on different data. This information can be used to feed continuous simulation models to extend the facade service life [6].

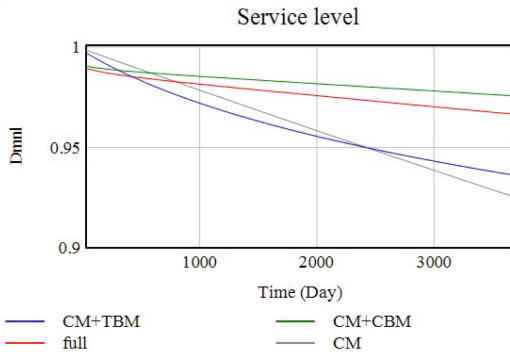


Fig. 5. Scenario simulation results

6 Conclusion and Next Steps

In this paper, we reported on initial steps towards a more explicit combination of business process modeling and system dynamics. We provided a step-by-step guide on how to use BPMN based models and system dynamics models together to model and analyze complex business processes, while illustrating this in terms of a case study on the maintenance of building facades.

As a next step, we aim to (1) further elaborate and validate the mapping as provided in Table 1, and the associated modeling guidelines, (2) develop experimental tool support to provide automatic support for this mapping and associated modeling activities, (3) develop standardized patterns to include specific facets such as energy consumption, CO2 emissions, etc., in the SFDs, (4) use these in further case studies, e.g. in relation to the PerpetuAL project and other circular economy efforts.

References

1. van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer, Heidelberg (2011). <https://doi.org/10.1007/978-3-642-19345-3>
2. Caballero-Villalobos, J.P., Mejía-Delgado, G.E., García-Cáceres, R.G.: Scheduling of complex manufacturing systems with petri nets and genetic algorithms: a case on plastic injection moulds. *Int. J. Adv. Manuf. Technol.* **69**(9–12), 2773–2786 (2013)
3. Fayoumi, A., Loucopoulos, P., Fayyoumi, A.: Hybrid enterprise modelling: integrating modelling mechanisms for socio-technical systems analysis and design. *J. Softw. Eng. Appl.* **7**(1), 6–13 (2014)
4. Freitas, A.P., Pereira, J.L.M.: Process simulation support in BPM tools: the case of BPMN. In: 5th International Conference on Business Sustainability, Povia de Varzim, Portugal, pp. 1–9 (2015)
5. Freund, J., Rüdiger, B.: Real life BPMN. Camunda (2012)
6. Giovanardi, M., Konstantinou, T., Pollo, R., Klein, T.: Internet of Things for building façade traceability: a theoretical framework to enable circular economy through life-cycle information flows. *J. Clean. Prod.* **382**, 135261 (2023)
7. Gopalakrishnan, S., Krogstie, J., Sindre, G.: Adapted UML activity diagrams for mobile work processes: experimental comparison of colour and pattern fills. In: Halpin, T., et al. (eds.) BPMDS/EMMSAD -2011. LNBP, vol. 81, pp. 314–331. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21759-3_23
8. Harmon, P.: System Dynamics and Business Processes (2008). <https://www.bptrends.com/bpt/wp-content/publicationfiles/advisor20080909.pdf>
9. Iacob, M.E., Jonkers, H., Lankhorst, M.M., Proper, H.A., Quartel, D.A.C.: ArchiMate 2.1 Specification. The Open Group (2013)
10. Jnitova, V., Joiner, K., Efatmaneshnik, M., Chang, E.: Modelling workforce employability pipelines for organisational resilience. *Int. J. Eng. Bus. Manag.* **13**, 1–19 (2021)
11. Kožusznik, J., Štolfa, S., Ježek, D., Kuchař, Š: Comparison of system dynamics and BPM for software process simulation. In: Snasel, V., Platos, J., El-Qawasmeh, E. (eds.) ICDIPC 2011. CCIS, vol. 189, pp. 1–15. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22410-2_1

12. Lankhorst, M.M., et al.: Enterprise Architecture at Work - Modelling, Communication and Analysis. The Enterprise Engineering Series, 3rd edn. Springer, Heidelberg (2012). <https://doi.org/10.1007/978-3-642-29651-2>
13. Messi, L., Spegni, F., Carbonari, A., Ridolfi, L., Vaccarini, M.: Process-based simulation models using BPMN for construction management at runtime. In: Proceedings of the Conference CIB W78 2021, Luxembourg, pp. 519–528 (2021)
14. OMG: Business Process Modeling Notation, V2.0. Technical Report. OMG Document Number: formal/2011-01-03, Object Management Group (2011)
15. Pereira, J.L., Freitas, A.P.: Simulation of BPMN process models: current BPM tools capabilities. In: New Advances in Information Systems and Technologies. AISC, vol. 444, pp. 557–566. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31232-3_52
16. Pourbafrani, M., van der Aalst, W.M.P.: Hybrid business process simulation: updating detailed process simulation models using high-level simulations. In: Research Challenges in Information Science, pp. 177–194. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-05760-1_11
17. Pourbafrani, M., van Zelst, S.J., van der Aalst, W.M.P.: Scenario-based prediction of business processes using system dynamics. In: Panetto, H., Debruyne, C., Hepp, M., Lewis, D., Ardagna, C.A., Meersman, R. (eds.) OTM 2019. LNCS, vol. 11877, pp. 422–439. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33246-4_27
18. Pourbafrani, M., van Zelst, S.J., van der Aalst, W.M.P.: Supporting automatic system dynamics model generation for simulation in the context of process mining. In: Abramowicz, W., Klein, G. (eds.) BIS 2020. LNBIP, vol. 389, pp. 249–263. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-53337-3_19
19. Proper, H.A.E., Hoppenbrouwers, S.J.B.A., van der Weide, T.P.: A fact-oriented approach to activity modeling. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2005. LNCS, vol. 3762, pp. 666–675. Springer, Heidelberg (2005). https://doi.org/10.1007/11575863_86
20. Serman, J.D.: Business Dynamics: Systems Thinking and Modeling for a Complex World. McGraw-Hill, New York (2000)
21. Tulinayo, F.P., van Bommel, P., Proper, H.A.: Enhancing the system dynamics modeling process with a domain modeling method. *Int. J. Cooper. Inf. Syst.* **22**(2), 1350011 (2013)
22. Tulinayo, P., Hoppenbrouwers, S.J.B.A., van Bommel, P., Proper, H.A.: Integrating system dynamics with object-role modeling and petri nets. In: Enterprise Modelling and Information Systems Architectures: Proceedings of the 3rd International Workshop on Enterprise Modelling and Information Systems Architectures; Ulm, Germany, 10–11 September 2009. LNI, vol. 152, pp. 41–54. GI (2009)
23. Wieringa, R.J.: Statistical difference-making experiments. In: Design Science Methodology for Information Systems and Software Engineering, pp. 295–317. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-43839-8_20