BEAR: BPMN and Environment AnimatoR

Flavio Corradini[1], Luca Mozzoni[1], Jessica Piccioni[1,*], Barbara Re[1], Lorenzo Rossi[1,*] and Francesco Tiezzi[2]

[1]School of Science and Technology, University of Camerino, Italy
[2]Dipartimento di Statistica, Informatica, Applicazioni, University of Florence, Italy

Abstract

Nowadays, BPMN is a de-facto standard for modeling multiple participants processes as collaboration diagrams. BPMN collaboration can be heavily influenced by the physical environment in which the involved participants act. Indeed, they can occupy a position, move in the space, make decisions according to the environment's status, and change it. However, BPMN lacks support for integrating the environment. To overcome this limitation, we present BEAR, a tool for modeling and animating together a BPMN collaboration and its environment in the form of a place graph. BEAR supports designers in precisely understanding the interplay between BPMN collaborations and the environment.

Keywords

BPMN collaboration, Physical Environment, Modeling, Animation.

1. Introduction

Business Process Model and Notation (BPMN) [1] has become a widely adopted standard for intuitively and expressively modeling business processes. Particularly, BPMN collaboration diagrams are very effective models for showing how participants e.g., humans, software systems, robots, and IoT devices, behave and interact by exchanging information to achieve a common objective [2].

The collaboration execution often depends on its working physical environment, in particular on the topology of the space and the environmental attributes, e.g., temperature, lighting, weather conditions, and network connectivity. Thus, in modeling such collaborations, it is crucial to consider the environment where participants operate [3]. Indeed, participants can occupy a position, move, and react to a particular environmental status. At the same time, activities, decisions, and events within a process can depend on environmental conditions and change the environmental status.

In this regard, the literature increasingly deals with the environment concept within BPMN [4, 5]. It focuses mainly on the effect of the environment on process execution while leaving the influence of process activities on the environment underspecified. Differently, in [6], we discuss such interplay, prescribing place graphs [7] and place attributes as an abstraction of the physical environment where the collaborations occur. We link the place graphs concepts with the BPMN meta-model, resulting in the definition of environmental BPMN collaboration diagrams, and we equip such models with formal operational semantics. Specifically, an environmental BPMN collaboration is the union of a BPMN collaboration and a place graph.

To enforce these findings and better guarantee a more precise understanding of the interplay between BPMN collaboration and the environment, we propose BEAR (BPMN Environmental Animator). BEAR is a novel BPMN animator tool that provides the user with a modeling environment to model environmental BPMN collaborations and observe their execution through the flow of tokens over the diagram and the evolution of environmental attributes.

It is indeed well known from the literature that animator tools play an important role in easing the understanding of the behavior of business processes [8]. In particular, different tools exist to model and

Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum co-located with 22nd International Conference on Business Process Management (BPM 2024), Krakow, Poland, September 1st to 6th, 2024.
*Corresponding author.
flavio.corradini@unicam.it (F. Corradini); luca.mozzoni@studenti.unicam.it (L. Mozzoni); jessica.piccioni@unicam.it (J. Piccioni); barbara.re@unicam.it (B. Re); lorenzo.rossi@unicam.it (L. Rossi); francesco.tiezzi@unifi.it (F. Tiezzi)

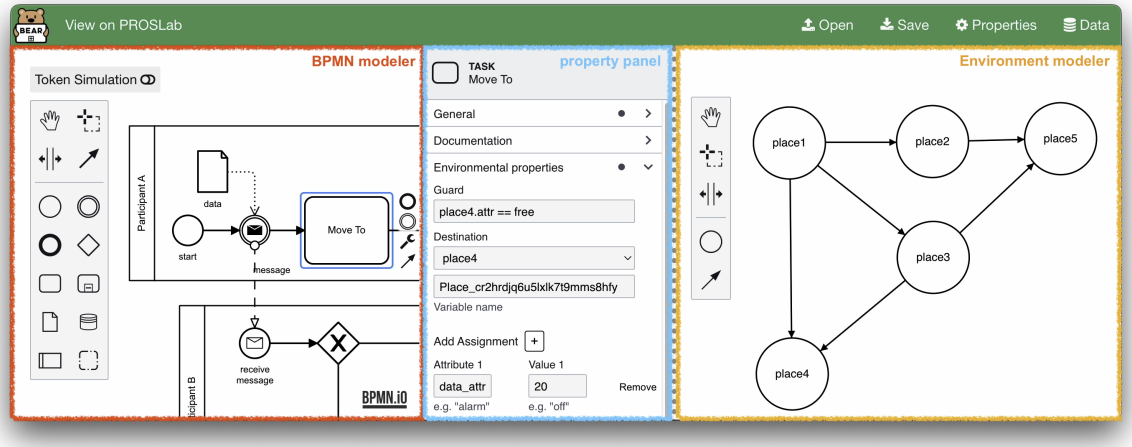

Figure 1: BEAR Modeler GUI

animate BPMN models. For instance, Allweyer and Schweitzer [9] and Signavio [10] provide tools to animate the control flow of BPMN processes; Corradini et al. [11] propose the tool MIDA to animate BPMN collaborations with multi-instance participants and data; Abdul et al. [12] present UBBA that creates and then animates a virtual 3D representation of a BPMN collaborations. Considering this landscape of animator tools, BEAR is the only one implementing environmental aspects within the BPMN notation.

The rest of the paper is organized as follows. Section 2 describes the functionalities of the BEAR tool. Section 3 assesses its maturity via case studies. Section 4 concludes the paper by providing links and information about downloading, installing, and using BEAR.

2. The BEAR tool features

This section presents the BEAR tool and its features for supporting the understanding of the interplay between collaboration business processes and the environment. BEAR provides *modeling* and *animation* functionalities. The former allows the designer to create environmental BPMN collaboration diagrams, while the latter is used to visualize the collaboration execution and the evolution of the environment.

The BEAR Modeler The interface of the BEAR modeler (Figure 1) is divided into two main parts i.e., the **BPMN modeler** on the left and the **environment modeler** on the right. The designer can use the BPMN modeler to create a collaboration diagram using the standard elements and can exploit the environment modeler to design a place graph representing the environment. A place graph consists of nodes that symbolize a predefined area and edges that define the connection between them.

In addition, using the button in the upper part of the interface, the designer can open the **property panel** that appears in between the two modelers. By clicking an element from one of the modelers, the property panel permits to add information related to the selected element and use them to drive the animation. Starting from the place graph, the designer can insert a list of attributes characterizing the environment in each place. These attributes are *key-value* pairs, e.g., `alarm = false` accessible by each process in the collaboration by using the place name in front of the attribute name, e.g., `place1.alarm`. All place attributes are accessible by the element of the BPMN collaboration in order to react and change them. Similarly, the property panel shows for each BPMN element, beyond the already existing properties like element name and identifier, it is possible to set new additional information related to the interplay with the environment and to process data. In particular, selecting a *pool* element, we added a new property called **position** that states the initial place of the participant represented by the pool. This property appears in BEAR as a select box that lists all the place names in the place

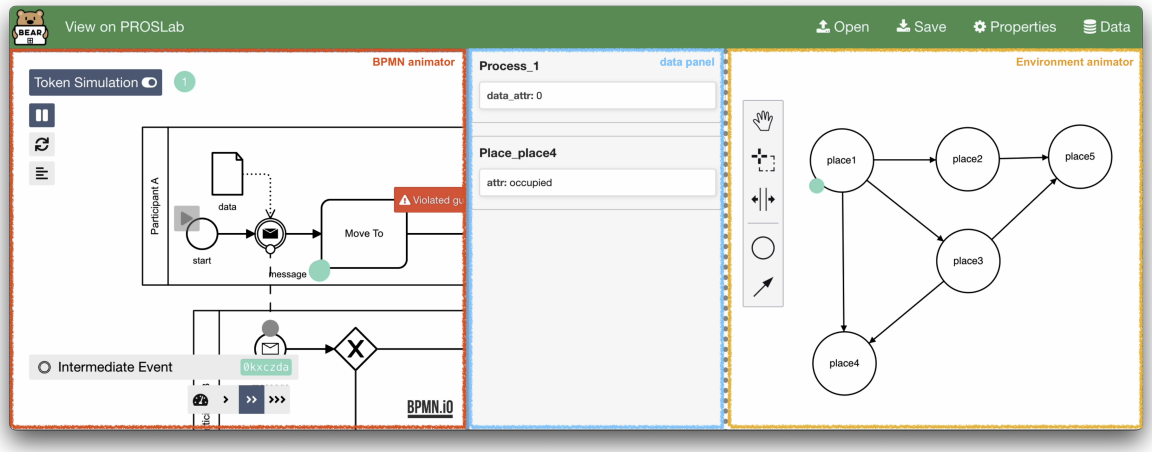

Figure 2: BEAR Animator GUI

graph. While selecting a *data object*, the designer can define process variables and instantiate them with a value e.g., `target = place1`. Such data are accessible only by process elements of the same pool like tasks, events, and gateways, making it possible to store, manipulate, and share information also related to the environment. About *tasks*, the designer can specify (i) a **guard** condition, used to constrain the execution of the task to specific values of data object and environmental attributes; (iii) a list of **assignments**, used to modify the environment topology (by adding and/or deleting nodes and/or places) or the values of data object and environmental attributes; and (iii) a **destination**, used to define a place in the place graph to reach from the current participant position. The destination can be static e.g., `place1` or dynamic using a data object attribute valorized at run-time with a place name. If the destination is `null`, the task does not involve any movement. For what concerns events the designer can define *message events* (start, intermediate, end) with value passing. Specifically, send events can exhibit a payload in the form of values e.g., `"Payload"` or `3`; or attribute names e.g., `place1.attr`. While receive events can exhibit an attribute name where to store the received payload. We also adopt, conditional events (start and boundary) to which the designer can add a boolean **condition** used to activate the event. A special condition for the conditional boundary event is the **unreachability**, used to react in case the participant can not reach its destination. Finally, boolean expressions can be assigned to the outgoing sequence flows of XOR gateways to drive decisions dynamically.

The BEAR Animator The key characteristic of the BEAR Animator (Figure 2) is its dynamic visual animation. It shows, in a step-by-step fashion, the evolution of the BPMN collaboration with its environment. By selecting the *Token Simulation* in the BEAR interface, a play button will appear over each fireable start event. Once this button is clicked, one process is activated. This creates a new token in the form of a small colored circle at the start event of the BPMN process and another token in the place of the place graph model corresponding to the value of the position property set on the pool. Then, the token starts to cross the BPMN collaboration model and the environment model according to the operational rules induced by our formal semantics [6]. A Data Panel in the BEAR interface allows the user to monitor the evolution of the values related to data objects and environmental attributes during the animation.

The animation terminates once all tokens cannot move forward. In case a token remains blocked due to data object or environmental attributes valorization, e.g. a guard condition violation, BEAR highlights it in red (Figure 2). Indeed, at any time, the animation can be paused by the user to check the distribution of tokens in the environment and into the BPMN collaboration. From a practical point of view, this allows designers to debug their environmental BPMN collaboration models. They can detect undesired executions, where e.g. a control flow is blocked, and deduce the cause beyond it by possibly

checking the tokens marking the BPMN model and the place graph.

3. Maturity of the tool

The BEAR tool is a web application written in JavaScript. It extends the bpmn-js[1] toolkit and its token simulation plug-in [13]. This makes BEAR accessible from any browser without a server backend, and hence, it can be run in any operative system. To facilitate its usage, BEAR can be installed as a Node application or a Docker container; moreover, we provide it online without the need to install any software. By its nature, BEAR does not require significant computational power. Indeed, an animator executes only a single run of the model using the input configuration, differently from simulators and model checkers that execute many runs of the same model, even all possible runs if necessary. For this reason, BEAR performances are not affected by the dimension of the input environmental BPMN collaboration diagram, and thus, it does suffer from scalability issues.

To assess the maturity of BEAR, we designed and implemented eleven case studies belonging to different application domains (e.g. agriculture, transport, restaurant, library, sale of products, and logistics), three of which are inspired by the literature [5, 14]. We choose environmental BPMN collaborations covering various dimensions, topology, and elements. The BPMN collaborations range from 2 to 4 participants and from 11 to 53 elements. They are both structured and unstructured and contain all the BPMN elements formalized in [6]. The space models' dimensions range from 11 places and 22 transitions to 44 places and 150 transitions. Each case study comes with a short description and two environmental BPMN diagrams, one is correct, i.e., the animation terminates without errors and each process terminates successfully; the second contains a modeling error mentioned in the description and verifiable animating it.

The evaluation of these case studies has shown how environmental conditions can change the outcome of the collaboration process. For the sake of presentation, we briefly discuss one of the case studies concerning a *fire-extinguishing* collaboration in a student dormitory. The collaboration is performed by a *fire control system* and a *fire-fighting* autonomous robot. The fire control system is in charge of detecting fire in dormitory rooms through fire sensors and informing the extinguisher robot that will reach the fire to extinguish it. The dormitory, inhabited by students, comprises a kitchen, an entrance, three bedrooms, two bathrooms, and a living room connected by a corridor. Each room has a door and a fire sensor. The collaboration starts when the fire control system detects the presence of fire in one of the dormitory rooms. In this case, the fire control system notifies the fire-fighting robot of the position of the fire and then activates the alarm. As soon as the robot receives the notification, it moves toward the fire and, after extinguishing it, sends feedback to the control system, which turns off the alarm. Notably, apart from the fire control system and the fire-fighting robot, some students in the dormitory perform their daily routines interacting with the environment. Based on this scenario, the BEAR animation can show two slightly different outcomes. Indeed, if the doors in the dormitory are open, the *fire-fighter* can reach any place if the fire occurs. In the case the *student* closes a door, she/he can make a room unreachable by the *fire-fighter*, which may be unable to extinguish the fire (Figure 3). The environmental BPMN collaboration models exemplifying these two outcomes are made available in the repository referenced in the next section.

4. Resources

We provide the BEAR tool under the MIT license and in different forms, i.e., source code, online service, and a Docker image. Moreover, we provide a detailed user guide explaining how to install and use the tool, the environmental BPMN collaboration models and the description of each case study. Finally, we provide a screencast showing the modeling and animation of the *fire-extinguishing* collaboration described in Section 3. All the mentioned resources are available at https://pros.unicam.it/environmental-bpmn/.

[1] https://bpmn.io/toolkit/bpmn-js/

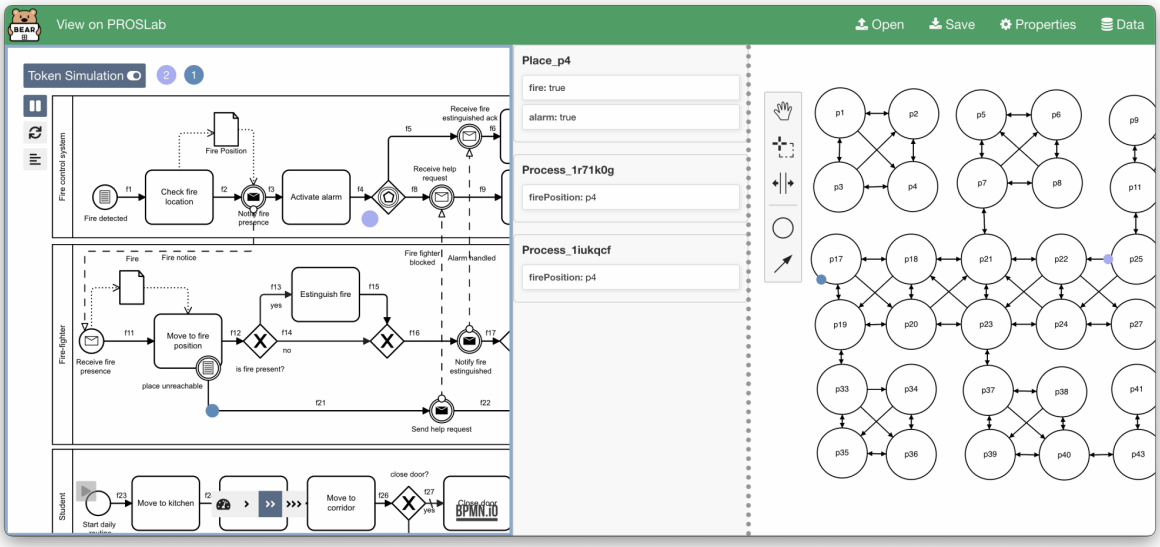

Figure 3: Unreachable position

References

[1] OMG, Business process model and notation, BPMN V2.0 (2011).

[2] C. Muzi, L. Pufahl, L. Rossi, M. Weske, F. Tiezzi, Formalising BPMN Service Interaction Patterns, in: The Practice of Enterprise Modeling, volume LNBIP, Springer, 2018, pp. 3–20.

[3] L. Poss, S. Schönig, A generic approach towards location-aware business process execution, in: Enterprise, Business-Process and Information Systems Modeling, volume 479 of *LNCS*, Springer, 2023, pp. 103–118.

[4] L. Poss, L. Dietz, S. Schönig, Labpmn: Location-aware business process modeling and notation, in: Cooperative Information Systems, volume 14353, Springer, 2024, pp. 198–216.

[5] R. Saddem-Yagoubi, P. Poizat, S. Houhou, Business processes meet spatial concerns: the sbpmn verification framework, in: Formal Methods, Springer, 2021, pp. 218–234.

[6] F. Corradini, J. Piccioni, B. Re, L. Rossi, F. Tiezzi, On the interplay between bpmn collaborations and the physical environment, in: BPM2024, Springer, 2024. To Appear.

[7] I. Afyouni, C. Ray, C. Christophe, Spatial models for context-aware indoor navigation systems: A survey, Journal of Spatial Information Science 1 (2012) 85–123.

[8] I. Maslov, S. Poelmans, Facilitating the comprehension of business process models for unexperienced modelers using token-based animations, Information and Management 61 (2024) 103967.

[9] T. Allweyer, S. Schweitzer, A Tool for Animating BPMN Token Flow, in: Int. Workshop on BPMN, volume 125 of *LNBIP*, Springer, 2012, pp. 98–106.

[10] S. GmbH, Signavio, 2018. URL: http://www.signavio.com/.

[11] F. Corradini, C. Muzi, B. Re, L. Rossi, F. Tiezzi, Formalising and animating multiple instances in bpmn collaborations, Infomation Systems 103 (2022).

[12] B. M. Abdul, F. Corradini, B. Re, L. Rossi, F. Tiezzi, Ubba: unity based bpmn animator, in: CAiSE Forum, volume 350 of *LNBIP*, Springer, 2019, pp. 1–9.

[13] P. Fromme, S. Warnke, P. Dehn, bpmn-js token simulation, 2017. URL: https://github.com/bpmn-io/bpmn-js-token-simulation.

[14] F. Corradini, S. Pettinari, B. Re, L. Rossi, F. Tiezzi, A technique for discovering bpmn collaboration diagrams, Software and Systems Modeling (2024) 1–21.