

Model Reader Preferences for Semantically Duplicate Elements in BPMN

Daniel Lübke^{1,2,*}, Volker Stiehl³

¹Digital Solution Architecture GmbH, Hannover, Germany

²Leibniz Universität Hannover, FG Software Engineering, Hannover, Germany

³TH Ingolstadt, Ingolstadt, Germany

Abstract

BPMN, which is the underlying modeling notation of many BPM endeavours and business information system development projects, is a rich modeling language, which also offers redundant constructs, i.e., different syntax can express the same semantics. We want to investigate which syntactical constructs are preferred by model readers if different ways to model message exchanges are offered by BPMN. In an empirical study we asked 77 participants which BPMN model they prefer for expressing eight situations. We found that send tasks and intermediate message catch events are significantly preferred. Also, event-based gateways are preferred over boundary events for many variants of the Deferred Choice pattern.

Keywords

BPMN, Empirical Study, Gateway, Boundary Event, Message, Subjective Preference, Event-based Gateway

1. Motivation

BPMN is THE standard for modeling business processes. Nowadays, business-critical applications based on BPMN and modern architectures [1, 2] are developed to digitize important business processes. Consequently, BPMN is used to communicate between a variety of stakeholders, e.g., developers and business analysts, and thus understandability is very important. While BPMN offers a wide set of modeling options for expressing many process details, it contains redundant constructs. For example, modeling message arrival time-outs can be modeled in different ways as explained in this paper. Allowing ambiguity how to model a certain situation allows for confusion and misunderstandings. Consequently, clarifying the usage of redundant syntax could standardize the current use of BPMN, streamline future versions of BPMN and thus make the notation easier to learn and understand. This paper presents a first step into this direction by investigating the subjective preferences of a) modeling message-based communication and b) representations of the deferred choice workflow pattern [3], when messages are involved. This paper is structured as follows. Within the next Section we present related

ZEUS'2023: S15th Central European Workshop on Services and their Composition, February 16–17, 2023, Hannover, Germany

*Corresponding author.

✉ daniel.luebke@digital-solution-architecture.com (D. Lübke); volker.stiehl@thi.de (V. Stiehl)

🌐 <https://www.digital-solution-architecture.com> (D. Lübke)

🆔 0000-0002-1557-8804 (D. Lübke)

© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

S. Böhm and D. Lübke (Eds.): 15th ZEUS Workshop, ZEUS 2023, Hannover, Germany, 16–17 February 2023, published at <http://ceur-ws.org>

work. In Section 3 the design of our empirical study is presented. The results are presented in Section 4 and an interpretation of those are given in Section 5. Finally, we conclude and give an outlook.

2. Related Work

Quality of business process models is multi-faceted. Lindland et al. [4] specified a framework that can be used to categorize different quality aspects of models, in which they distinguish between syntactic, semantic, and pragmatic qualities. This paper is concerned with subjective preference of certain model constructs. Because “[i]n general, researchers associate aesthetics with readability, and readability with understanding” [5] subjective preference is a part of *understandability* and thus a pragmatic quality. Or as Lindland et al. put it: Understandability is the main concern of pragmatic model quality, which “affects how to choose from among the many ways to express a single meaning” [4]. Comprehension of BPMN models is a vast research area: For example, there are studies concerning the influence of layout on understandability. Figl provides a good overview [6]. Scholz & Lübke [7] investigated subjective layout preferences and used the same research design as we do: By using a quiz-like study, in which participants choose one of the presented options, they have analyzed subjective preferences of different choices for BPMN layouts. Moody [8] has critiqued BPMN in general for failing to adhere to his “Physics of Notations” [9] – especially that BPMN has considerable semantic redundancy, e.g., the *Exclusive OR Gateway* has two visual representations. Genon et al. [10] found the same. The eCH-0158 modeling guidelines for BPMN [11] recognize the redundancy between *send/receive tasks* and *message catching/throwing events*. They standardize on *send tasks* and *message catch events*.

3. Study Design

3.1. Goals, Hypothesis & Variables

By following the Goal-Question-Metric (GQM) approach [12] we are defining our goal as

Understand the Subjective Preference
with regard to *Semantically Equivalent Elements in BPMN 2.0*
from the viewpoint of a *Model Reader*.

This goal is refined into (research) questions. While BPMN has many redundancies, we concentrate on the ones below. We want to answer, which construct for each of the following pairs of semantically equivalent BPMN constructs are preferred:

RQ1 Send Task vs. Intermediate Message Throw Event: BPMN offers two elements for sending messages: The *send task* and the *intermediate message throw event* both send a message.

RQ2 Receive Task vs. Intermediate Message Catch Event: Similarly to sending a message, BPMN also offers a *receive task* and an *intermediate message catch event for receiving*

a message.

RQ3 Send Task vs. End Message Throw Event: For modeling the sending of a message at the end of a process execution, a *send task* and a *none end event* can be used. Alternatively, an *message throw end event* can be used.

RQ4 Deferred Choice between two messages (diff. prob.): A Deferred Choice [3] between two incoming messages can be modelled via an *event-based gateway* or a *receive task* with an *interrupting message boundary event*. Because one participant in [7] indicated that he/she would model splits and joins differently depending on the probability of the branch taken, we differentiate between the probability of events. This question is concerned with messages that have different probabilities, i.e., the top *event* after the *event-based gateway* and the message caught by the *receive task* are more likely to occur than the bottom *event*, which is more exceptional, after the *event-based gateway* and the message caught by the *boundary event*.

RQ5 Deferred Choice between two messages (same prob.): This question is similar to RQ4. However, the incoming messages have the same probability, i.e., both events following the *event-based gateway* and both messages occur equally often.

RQ6 Deferred Choice between message and timer (diff. prob.): This question is similar to RQ4 but this time the Deferred Choice is not between two messages but instead resembles a deadline situation with a *message event* and a *timer event*. It is more probable to receive the message than to time-out. This pattern is presented as an *event-based gateway* with two following *events* or with a *receive task* with an *interrupting timer boundary event*.

RQ7 Deferred Choice between message and timer (same prob.): This question is similar to RQ6. However, the incoming message and the time-out have the same probability.

RQ8 Deferred Choice between two messages and a timer: The last question is concerned with a Deferred Choice between two messages and a timer, i.e., a scenario in which one of two messages must be received within a certain time. This can – again – be modeled as an *event-based gateway* followed by two *message events* and one *timer event*, or by a *receive task* with two *boundary events*.

3.1.1. Measurements & Hypothesis

We measure the subjective preferences of study participants as the only metric for all research questions. For all research questions the null hypothesis H_0 is that there is no preference for one of the two alternatives. Accordingly, H_1 is that one of the two alternatives is preferred.

3.2. Objects

The study setup is similar to a previous study by Scholz & Lübke [7]: Participants take part in an online survey in which two diagrams modeling the same process are shown which only

Table 1
Description of the Participants Groups of our Study

Group	Experience	Description	Count
LUH1	Students	MSc./CS, Software Architecture Lecture	20
LUH2	Students	BSc./CS, Software Engineering Seminar	3
LUH3	Students	MSc./CS, Software Methodologies Lecture	4
THI1	Students	BSc./IS, 4th semester	11
THI2	Students	BSc./IS, 6th semester	6
THI3	Students	MSc./IE, 2nd semester	9
THI4	Students	BSc./IE, 6th semester	11
Prof.	Professionals	recruited from different organizations	13
Total			77

differ in one point. In this study different but semantically equivalent BPMN diagrams were used as shown in Appendix A. Both options were shown side by side and participants had to choose the preferred one by clicking it. Descriptive text was shown to convey the probability of some branches. Since branching probabilities cannot be modeled in BPMN directly, it was necessary to convey this information textually.

3.3. Participants

Participants were a) recruited from lectures of the authors and b) professionals were asked to participate. We tracked the group to which a participant belongs to by using different invitation links. Participation was voluntary and no incentives were given. The number of participants per group and a more detailed description is shown in Table 1. All in all, we had 87 participants in total. After removing those, who did not complete the quiz or changed their answers in between, 77 participants remained.

3.4. Validity Procedure

As a first step we performed a power test: For a two-sided hypothesis test with $\alpha = p = 0.05$ and confidence $\beta = 0.95$ for a medium effect of $h = 0.5$ yields that we required at least 52 participants. As described above we recruited more participants than required. For eliminating extraneous variables we took following measures: We randomized the order in which questions (i.e., diagram pairs) were shown. Thereby, we try to eliminate learning and fatigue effects. We also randomized the order in which diagrams are shown.

4. Analysis

The statistical evaluation of the gathered data is shown in Table 2. The statistical significance indicated by the p-values is marked by asterisks (*: $p \leq 0.05$, **: $p \leq 0.01$, ***: $p \leq 0.001$). Similarly, the effect is denoted by pluses (+: $h \geq 0.2$, ++: $h \geq 0.5$, +++: $h \geq 0.8$).

Table 2
Results and Hypothesis Test Results for all Questions

Question	#A	#B	p	*	h	+
Q1 Send Task vs. Message Throw Event	53	24	0.0013	**	0.39	+
Q2 Receive Task vs. Message Catch Event	29	48	0.0395	*	0.25	+
Q3 Send Task vs. Message End Event	39	38	1.0000		0.01	
Q4 Deferred Choice, 2 messages, diff. prob. Gateway vs. Boundary Event	53	24	0.0013	**	0.39	+
Q5 Deferred Choice, 2 messages, same prob. Gateway vs. Boundary Event	69	8	0.0000	***	0.91	+++
Q6 Deferred Choice, message+timer, diff. prob. Gateway vs. Boundary Event	36	41	0.6488		0.06	
Q7 Deferred Choice, message+timer, same prob. Gateway vs. Boundary Event	43	34	0.3620		0.12	
Q8 Deferred Choice, 2 messages+timer Gateway vs. Boundary Events	58	19	0.0000	***	0.53	++

5. Interpretation

5.1. Evaluation of Results & Implications

The *send task* is significantly preferred over a *message throw event* (RQ1). It seems that participants see the sending of a message more as a task, i.e., an active action, and therefore prefer the task instead of an event.

In contrast to RQ1, participants significantly prefer a *message catch event* for waiting on a message receive (RQ2). Interestingly, it is inconsistent to use different syntax for sending and receiving messages. This can mean that perhaps participants differentiate between active and passive/waiting elements.

There is no significant difference for sending a message at the process end (RQ3). In contrast to a significant preference for a *send task* during the process, there is no clear preference for a *send task* with an *end event* or a *message end event*. It seems that the additional penalty of a second symbol and its associated space requirements is not worth to keep up the semantic difference experienced in RQ1.

When modeling a Deferred Choice between two messages which arrive with different probabilities, participants prefer the use of an *event-based gateway* (RQ4). It may be that the visuals of two white envelopes – one in the *receive task* and one in the *boundary event* – is not attractive. Participants have an even stronger preference for the *gateway* if the probability of the messages are the same (RQ5).

When modeling a time-out, i.e., a Deferred Choice between a message and a timer, neither the *gateway* nor the *boundary event* is preferred – regardless of whether the timer is as likely to occur (RQ6) or is only triggered as an exception (RQ7). This contrasts with the results from RQ4/5, which are structurally the same but use a different second event. While more participants liked the *gateway* for same probabilities of events and more participants liked the *boundary event* for exceptional cases, these differences were not significant. More research has to further

clarify whether there is a difference with a small effect or not.

If the Deferred Choice is between two messages and a timer event (RQ8) there is a strong, significant preference to the *event-based gateway*. However, we cannot attribute to why this is: While in our study planning we wanted to examine the effect of a larger number of boundary events, another possible explanation is that a send task with a message boundary event is disliked as RQ4 and RQ5 have shown.

5.2. Limitations of Study

Because we only measured subjective preferences no quantitative data on model comprehension could be measured. This study still gives insights into model perception, especially with different variants of the Deferred Choice pattern. Like all studies which include students, the question of generalizability arises. However, we have seen that no differences between our groups exist – this also means that the group of professionals does not behave significantly different from the students. While we had a considerable amount of participants, some research questions gave non-significant results with a small effect size in the range of $0.1 \leq h \leq 0.2$. To have adequate power in the statistical tests, more participants (approx. 350) are required.

6. Implications for Practitioners

Following from these results practitioners should amend existing modeling guidelines by the following rules: 1) Use *Send Tasks* for sending messages during process execution, 2) use *Message Catch Events* for receiving messages during process execution, and 3) use *Event-based Gateways* when implementing the Deferred Choice pattern when receiving multiple messages. Modelers should keep in mind that this is the first study to examine these constructs. Hopefully, future studies will strengthen or refute these results and thus these proposed modeling guidelines.

7. Conclusions & Outlook

Within this paper we presented our empirical study with students from two universities and professionals on the subjective preference of syntactically redundant, message-related constructs in BPMN. We found a strong subjective preference for *send tasks* over *message throw events* within the process-flow and for *message catch events* over *receive tasks*. We also found that Deferred Choices in *event-based gateways* are preferred over *boundary events* in the case of two message events or three events. We could find no significant preference for Deferred Choices with a message and a timer (“time-outs”) or for the sending of a message on process completion. While the results are interesting in themselves, this study lays the foundation for further empirical inquiries: Follow up studies, especially experiments, can investigate and compare understandability of redundant BPMN message-related constructs. This way, especially eye-tracking experiments, can be used to gather quantitative data to evaluate whether the subjective preferences match the differences in objective understandability in the future, and further developing modeling guidelines for BPMN.

Acknowledgments

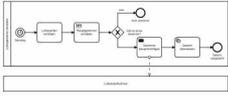
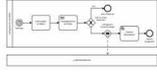
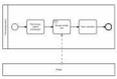
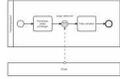
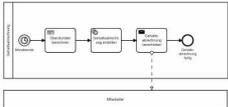
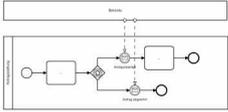
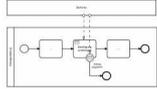
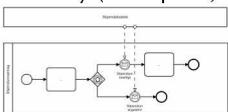
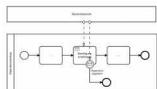
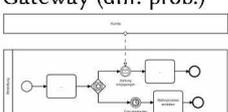
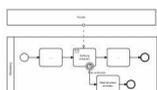
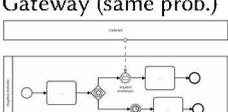
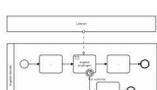
We'd like to thank all participants who took part in our study. Additionally, we like to thank Kurt Schneider, Dieter Kähny, and Barbara Ulrich for distributing the quiz within their classes and organizations.

References

- [1] V. Stiehl, *Implementing the Basic Architecture of Process-Driven Applications*, Springer, 2014. URL: http://link.springer.com/chapter/10.1007/978-3-319-07218-0_4.
- [2] B. Rücker, 3 common pitfalls in microservice integration and how to avoid them, WWW: <https://berndruecker.io/3-pitfalls-in-microservice-integration/>, last access: 2021-02-18, 2018.
- [3] W. M. van Der Aalst, A. H. Ter Hofstede, B. Kiepuszewski, A. P. Barros, *Workflow patterns, Distributed and parallel databases* 14 (2003) 5–51.
- [4] O. I. Lindland, G. Sindre, A. Solvberg, *Understanding quality in conceptual modeling*, *IEEE Software* 11 (1994) 42–49. doi:10.1109/52.268955.
- [5] C. Bennett, J. Ryall, L. Spalteholz, A. Gooch, *The aesthetics of graph visualization*, *Proceedings of Computational Aesthetics in Graphics, Visualization, and Imaging* (2007) 57–64. doi:10.2312/COMPAESTH/COMPAESTH07/057-064.
- [6] K. Figl, J. Recker, *Exploring cognitive style and task-specific preferences for process representations*, *Requirements Engineering* 21 (2016) 63–85. URL: <http://dx.doi.org/10.1007/s00766-014-0210-2>. doi:10.1007/s00766-014-0210-2.
- [7] T. Scholz, D. Lübke, *Improving automatic bpmn layouting by experimentally evaluating user preferences*, in: Á. Rocha, H. Adeli, L. P. Reis, S. Costanzo (Eds.), *New Knowledge in Information Systems and Technologies*, Springer International Publishing, Cham, 2019, pp. 748–757.
- [8] D. L. Moody, *Why a Diagram is Only Sometimes Worth a Thousand Words: An Analysis of the BPMN 2.0 Visual Notation*, 2011.
- [9] D. L. Moody, *The physics of notations: toward a scientific basis for constructing visual notations in software engineering*, *Software Engineering, IEEE Transactions on* 35 (2009) 756–779.
- [10] N. Genon, P. Heymans, D. Amyot, *Analysing the cognitive effectiveness of the bpmn 2.0 visual notation*, in: B. Malloy, S. Staab, M. van den Brand (Eds.), *Software Language Engineering*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 377–396.
- [11] A. Birchler, E. Bosshart, M. Märki, P. Opitz, J. Pauli, B. Rigert, Y. Sandoz, M. Schaffroth, N. Spöcker, C. Tanner, K. Walser, T. Widmer, *eCH-0158 BPMN-Modellierungskonventionen für die öffentliche Verwaltung*, WWW: <https://www.ech.ch/dokument/fb5725cb-813f-47dc-8283-c04f9311a5b8>, 2014. URL: <https://www.ech.ch/dokument/fb5725cb-813f-47dc-8283-c04f9311a5b8>.
- [12] V. R. Basili, *Applying the goal/question/metric paradigm in the experience factory*, *Software Quality Assurance and Measurement: A Worldwide Perspective* (1993) 21–44.

A. Diagrams

Table 3
Diagrams used in our Study

RQ1	Send Task	Message Throw Event
		
RQ2	Receive Task	Message Catch Event
		
RQ3	Send Task	End Message Event
		
RQ4	Gateway (diff. prob.)	Boundary Event (Msg.)
		
RQ5	Gateway (same prob.)	Boundary Event (Msg.)
		
RQ6	Gateway (diff. prob.)	Boundary Event (Timer)
		
RQ7	Gateway (same prob.)	Boundary Event (Timer)
		
RQ8	Gateway	Boundary Event (Timer+Event)
	