Studying Domain Dependence in BPMN Process Modeling: An Empirical Research Proposal

Thomas S. Heinze[1]

[1]Cooperative University Gera-Eisenach (DHGE), Weg der Freundschaft 4, 07546 Gera, Germany

Abstract

The Business Process Model and Notation (BPMN) language as a de facto standard is omnipresent in business process modeling. This not only applies to its vertical scope, ranging from pen-and-paper process sketches to fully-implemented process automation, but also to its horizontal spectrum, comprising differing domains, e.g., financial services, healthcare, e-government, or webshops. In this paper, we argue for the analysis of differences in the usage of BPMN in varying application domains using an empirical approach: Studying and contrasting features of BPMN models found in process model repositories of varying domains allows us to gain insights into the domains' modeling characteristics. The resulting findings may then be beneficial for defining modeling best practices, supporting future language standardization, or improving (data-driven) modeling tools.

Keywords

mining software repositories, business process modeling, e-government, BPMN

1. Introduction and Motivation

A business process defines a collection of interconnected tasks and activities which allow an organization to achieve a certain goal or objective. Modeling business processes is a crucial step in business process management and its resulting process models form the foundation of the various stages in the business process lifecycle [1] and are thus used for, e.g., documenting, analyzing, or improving business processes. The *Business Process Model and Notation (BPMN)* [2] is a widely accepted business process modeling language, both in industry and academia. BPMN not only supports the various stages of the business process lifecycle and covers the full vertical scope, ranging from sketched process drafts to automated process implementations, but is also the de facto standard in a wide spectrum of application domains, including, e.g., financial or business services, healthcare, e-government, and webshops. As a result, syntax and semantic of the BPMN modeling language is quite crammed and complex, as is illustrated well by the 500 pages of the current BPMN standard language specification [2].

Due to its versatility, we expect different modeling practices and styles in BPMN's day-to-day usage. Empirical research methods like *software repository mining* can help to learn more and understand about how a modeling language is used in practice [3, 4]. Using this knowledge then allows for, e.g., defining modeling guidelines and best practices, supporting future language standardization efforts, or improving modeling tools to better serve their user needs. In prior research, various BPMN language corpora have consequently been established by systematically searching *Github software repositories* and identifying BPMN models [5, 6], yielding extensive data sets comprising thousands of business process models from real open source projects. The resulting data sets were then used to empirically investigate on various research problems, including questions about the standard compliance of the found process models [3, 5], the adaptation of certain design choices and modeling practices [6, 7], or about the frequency of process model duplication and cloning [4, 8]. While this previous work allowed for a better understanding of the general usage of BPMN in open source projects, covering a wide range of differing application domains [4], it remains open whether the obtained findings similarly apply to domains. In particular, the question whether there are not only differences in BPMN's usage across the vertical scope but also across its various application domains, while having been addressed for certain singular domains like healthcare in previous related work [9], still remains open.

Vienna'25: 17th Central European Workshop on Services and their Composition (ZEUS), February 20–21, 2025, Vienna, Austria
thomas.heinze@dhge.de (T. S. Heinze)

S. Böhm and M. Vidgof (Eds.): 17th ZEUS Workshop, ZEUS 2025, Vienna, Austria, 20–21 February, 2025 published at http://ceur-ws.org

In this position paper, we argue for studying the *domain dependence* of BPMN usage. We believe this to be a worthwhile effort due to various reasons: First, such an analysis helps in providing a more thorough picture of the language's pragmatics. For instance, certain application domains may impose modeling styles and practices which deviate and even contradict common BPMN usage, though be important when defining modeling guidelines or best practices. Second, analyzing the characteristics of BPMN's usage in differing domains may help in identifying and prioritizing specific requirements which can be incorporated in future language versions or in defining language subsets or profiles for certain application domains. Third and finally, the advent of large language models (LLMs) promises effective applications of machine learning and data-driven approaches to business process management. This also comprises the use of LLMs for business process modeling, e.g., providing tools for autocompletion of activity labels or other BPMN model structures and transforming process specifications formulated in natural text into matching BPMN process models (cf. [10] for an example in e-government). However, the quality of a LLM's predictions is influenced by the quality of its training data and the model's generalization capabilities. This can become problematic if a certain domain features special characteristics, which are otherwise underrepresented in the training data and resulting models. This phenomenon, well-known as out-of-distribution problem [11, 12], may as well apply to the application of LLMs to the process modeling language BPMN across varying application domains.

2. Prospective Case Study: German E-Government

In 2017, the German federal government released a legislation whose primary goal was to provide citizens digital access to 575 selected administrative services and processes over all administrative divisions by the end of 2022, (so-called *Onlinezugangsgesetz (OZG)*). While this objective was not met, the law has been a driver for various e-government initiatives in Germany which also address a more holistic end-to-end digitization of administrative processes. As part of these initiatives, the *Federal Information Management (FIM)* standard[1] is responsible for providing a streamlined methodology for supporting the translation of legal requirements as entailed by law into digital public services implemented by authorities. Due to the federal structure of Germany and since administrative services and processes can thus vary across the different authorities, e.g., federal/state agencies and municipalities, FIM follows a layered approach. At the top layer, master information is derived directly from law and defines administrative services and processes without characteristics of the implementing agency. The reference layer adds technical and organizational aspects, which is for instance required to provide the user-friendly digital services claimed by the OZG. Finally, the local layer allows implementing agencies to add even more details about their respective characteristics, e.g., information about concrete IT services used, etc. In this way, FIM supports reuse of service and process definitions (so-called *"one for all"* principle), but allows for customization according to the characteristics of implementing agencies.

On a technical note, the FIM standard includes three different modules: (1) *XZuFi*, (2) *XDatenfelder*, and (3) *XProzess*[2], which comprise XML-based specifications to define administrative services, data fields/forms, and processes, respectively. XZuFi defines overall information about administrative services, in particular including a unique identifier (*Leistungskatalog-ID (LeiKa-ID)*), which can be referenced in the other modules. XDatenfelder provides uniform structures for data forms and elements together with corresponding plausability rules, which are utilized in the administrative services and build the foundation of web forms for the respective digital services. Eventually, XProzess is used to define the processes modeling administrative services. To this end, the XProzess notation embeds process models in the BPMN 2.0 language. Note that XProzess therefore defines limited subsets of BPMN to be used in process definitions at the master and reference layer: *FIM-BPMN* and *OZG-BPMN*. All three modules are accompanied by libraries and common building blocks. The former provides a repository infrastructure and the latter lowers the implementation burden by providing reusable components for problems which occur frequently and are unspecific to a single administrative service.

[1] https://neu.fimportal.de/

[2] https://www.xrepository.de/details/urn:xoev-de:mv:em:standard:xprozess

3. Research Proposal

We propose the German e-government initiative as a starting point for our investigation into the domain dependence of BPMN usage. As a first step, a corpus of BPMN process models from this application domain has to be established. Mining the respective process repositories, e.g., the FIM master data library[3], allows for retrieving XProzess specifications, which can subsequently be parsed to extract BPMN process models. The process models of the resulting data set can then for example be analyzed in conjunction with the more general corpora of open source BPMN models, which have been scraped from Github [3, 5]. In a preliminary analysis, we are then interested in studying and contrasting aspects like process model size, used BPMN features and modeling tools, frequency of model clones, incidence of modeling styles and practices, etc., similar to prior work [6, 8]. A literature survey on the domain dependence usage of BPMN, also within other domains, e.g., healthcare [9], and including domain-specific extensions [13], will complement the insights gained in the preliminary analysis.

As a next step, in a more thorough analysis, certain modeling aspects can be further investigated. For instance, we assume activity labels to be tightly coupled to the application domain. As a result, metrics may emerge for differentiating BPMN process models from different application domains. Inferring such metrics is therefore another research step, which can, e.g., be tackled by training autoencoders that classifies the application domain using the process metrics as input. Note that the mentioned activity labels also play an important role for many process modeling and analysis tools. Algorithms for detecting model clones are for example using activity labels for assessing model similarity [4]. Thus, the tools' performance may be influenced by the application domain. In particular when it comes to tools employing machine learning, as outlined above, the out-of-distribution problem can become an issue. Considering similar experiences with similar tasks for label prediction in conventional programming languages [12], we expect interesting results.

Acknowledgments

The author thanks the reviewers for their helpful comments that improved the quality of the paper.

Declaration on Generative AI

The author has not employed any generative AI tools.

References

[1] M. Dumas, M. L. Rosa, J. Mendling, H. A. Reijers, Fundamentals of Business Process Management, Second Edition, Springer, 2018. URL: https://doi.org/10.1007/978-3-662-56509-4.

[2] Object Management Group (OMG), Business Process Model and Notation (BPMN), Version 2.0.2, OMG Document formal/2013-12-09, 2014. URL: https://www.omg.org/spec/BPMN/2.0.2/PDF.

[3] T. S. Heinze, V. Stefanko, W. Amme, Mining BPMN Processes on GitHub for Tool Validation and Development, in: S. Nurcan, I. Reinhartz-Berger, P. Soffer, J. Zdravkovic (Eds.), Enterprise, Business-Process and Information Systems Modeling - 21st International Conference, BPMDS 2020, 25th International Conference, EMMSAD 2020, Held at CAiSE 2020, Grenoble, France, June 8-9, 2020, Proceedings, volume 387 of *Lecture Notes in Business Information Processing*, Springer, 2020, pp. 193–208. URL: https://doi.org/10.1007/978-3-030-49418-6_13.

[4] M. S. Nikoo, S. Kochanthara, Ö. Babur, M. van den Brand, An empirical study of business process models and model clones on GitHub, Empir. Softw. Eng. 30 (2025) 48. URL: https://doi.org/10.1007/s10664-024-10584-z.

[3] https://neu.fimportal.de/

[5] J. Türker, M. Völske, T. S. Heinze, BPMN in the Wild: A Reprise, in: J. Manner, D. Lübke, S. Haarmann, S. Kolb, N. Herzberg, O. Kopp (Eds.), Proceedings of the 14th Central European Workshop on Services and their Composition (ZEUS 2022), Bamberg, Germany, February 24-25, 2022, volume 3113 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 68–75. URL: https://ceur-ws.org/Vol-3113/paper11.pdf.

[6] E. Baalmann, D. Lübke, Validation of Algorithmic BPMN Layout Classification (short paper), in: S. Böhm, D. Lübke (Eds.), Proceedings of the 15th ZEUS Workshop, Hannover, Germany, February 16-17, 2023, volume 3386 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 13–20. URL: https://ceur-ws.org/Vol-3386/paper3.pdf.

[7] D. Lübke, D. Wutke, Analysis of Prevalent BPMN Layout Choices on GitHub, in: J. Manner, S. Haarmann, S. Kolb, N. Herzberg, O. Kopp (Eds.), Proceedings of the 13th European Workshop on Services and their Composition (ZEUS 2021), Bamberg, Germany, February 25-26, 2021, volume 2839 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 46–54. URL: https://ceur-ws.org/Vol-2839/paper9.pdf.

[8] R. Laue, M. Läuter, Beobachtungen und Einsichten zu Repositorys von BPMN-Modellen, in: J. Michael, M. Weske (Eds.), Modellierung 2024, Potsdam, Germany, March 12-15, 2024, volume P-348 of *LNI*, Gesellschaft für Informatik e.V., 2024, pp. 157–173. URL: https://doi.org/10.18420/modellierung2024_015.

[9] L. Pufahl, F. Zerbato, B. Weber, I. Weber, BPMN in healthcare: Challenges and best practices, Inf. Syst. 107 (2022) 102013. URL: https://doi.org/10.1016/j.is.2022.102013.

[10] S. T. Bachinger, L. Feddoul, M. J. Mauch, B. König-Ries, Extracting Legal Norm Analysis Categories from German Law Texts with Large Language Models, in: H. Liao, D. Duenas-Cid, M. A. Macadar, F. Bernardini (Eds.), Proceedings of the 25th Annual International Conference on Digital Government Research, DGO 2024, Taipei, Taiwan, June 11-14, 2024, ACM, 2024, pp. 481–493. URL: https://doi.org/10.1145/3657054.3657277.

[11] D. Berend, X. Xie, L. Ma, L. Zhou, Y. Liu, C. Xu, J. Zhao, Cats Are Not Fish: Deep Learning Testing Calls for Out-Of-Distribution Awareness, in: 35th IEEE/ACM International Conference on Automated Software Engineering, ASE 2020, Melbourne, Australia, September 21-25, 2020, IEEE, 2020, pp. 1041–1052. URL: https://doi.org/10.1145/3324884.3416609.

[12] B. Gruner, T. Sonnekalb, T. S. Heinze, C. Brust, Cross-Domain Evaluation of a Deep Learning-Based Type Inference System, in: 20th IEEE/ACM International Conference on Mining Software Repositories, MSR 2023, Melbourne, Australia, May 15-16, 2023, IEEE, 2023, pp. 158–169. URL: https://doi.org/10.1109/MSR59073.2023.00034.

[13] R. Braun, W. Esswein, Classification of Domain-Specific BPMN Extensions, in: U. Frank, P. Loucopoulos, O. Pastor, I. Petrounias (Eds.), The Practice of Enterprise Modeling - 7th IFIP WG 8.1 Working Conference, PoEM 2014, Manchester, UK, November 12-13, 2014. Proceedings, volume 197 of *Lecture Notes in Business Information Processing*, Springer, 2014, pp. 42–57. URL: https://doi.org/10.1007/978-3-662-45501-2_4.