TOWARDS A MODEL FOR INTEGRATING PROCESS VERIFICATION IN PROCESS-DRIVEN APPLICATION DEVELOPMENT

HƯỚNG ĐẾN MỘT MÔ HÌNH TÍCH HỢP XÁC THỰC QUY TRÌNH TRONG PHÁT TRIỂN ỨNG DỤNG DỰA TRÊN QUY TRÌNH

Nguyen Thanh Tuan[1], Le Thanh Nhan[2], Hoang Thi Thanh Ha[3]*

[1]*The University of Danang - Vietnam-Korea University of Information and Communication Technology, Vietnam*
[2]*The University of Danang - Danang International Institute of Technology, Vietnam*
[3]*The University of Danang - University of Economics, Vietnam*

*Corresponding author: ha.htt@due.edu.vn

(Received: May 02, 2024; Revised: October 08, 2024; Accepted: November 21, 2024)

Abstract - In the era of Industry 4.0, companies must quickly adapt to market and technological changes to remain competitive. While monolithic software systems are still common, Process-Driven Applications (PDA) offer flexible, sustainable, and integrable solutions, relying on Business Process Model and Notation (BPMN) for process modeling. However, BPMN lacks formal semantics, leading to potential errors during process execution. This paper introduces a verification model that integrates Colored Petri Nets (CPN) into the PDA development lifecycle to improve process verification. By converting BPMN models into CPN, the method allows for early detection of errors such as deadlocks and inconsistencies. A real-world, loan application, case study demonstrates that the BPMN + CPN approach reduces errors by 80%, significantly enhancing the reliability and robustness of PDA systems.

Key words - Process-Driven applications; process Verification; Colored Petri Nets; BPMN; DMN.

Tóm tắt - Trong kỷ nguyên Công nghiệp 4.0, khả năng thích ứng nhanh chóng với thay đổi thị trường và công nghệ là yếu tố quan trọng giúp các công ty cạnh tranh. Mặc dù, các hệ thống phần mềm nguyên khối vẫn phổ biến, Ứng dụng Dựa trên Quy trình (PDA) cung cấp giải pháp linh hoạt, bền vững và tích hợp, sử dụng Mô hình và Ký hiệu Quy trình Nghiệp vụ (BPMN) để mô hình hóa quy trình. Tuy nhiên, BPMN thiếu ngữ nghĩa hình thức, gây ra các lỗi tiềm ẩn trong quá trình thực thi. Bài báo này đề xuất một mô hình xác thực tích hợp Mạng Petri màu (CPN) vào vòng đời phát triển PDA nhằm nâng cao khả năng xác thực quy trình. Bằng cách chuyển đổi các mô hình BPMN sang CPN, phương pháp này cho phép phát hiện sớm các lỗi như deadlock và các bất nhất trong cấu trúc. Thông qua nghiên cứu tình huống là Quy trình xử lý đơn vay, phương pháp BPMN + CPN giúp giảm 80% lỗi, đồng thời cải thiện đáng kể tính ổn định và độ tin cậy của hệ thống PDA.

Từ khóa - Ứng dụng dựa trên quy trình; xác thực quy trình; mạng Petri màu; BPMN; DMN.

1. Introduction

In today's rapidly evolving industrial landscape, driven by technological innovation and shifting market dynamics, companies are under immense pressure to enhance their agility and responsiveness. Legacy software systems, particularly those built on monolithic architectures, are often ill-equipped to meet these new demands due to their inherent rigidity and limited scalability. To address these limitations, Process-Driven Applications (PDA) have emerged as a modern solution, offering a flexible framework that supports the seamless integration of business processes across various platforms and systems [1].

PDA are typically constructed using Business Process Model and Notation (BPMN) [2], a widely recognized standard that enables the graphical representation of business workflows. BPMN provides a user-friendly way to model processes, making it easier for both technical teams and business stakeholders to collaborate in designing operational workflows.

Decision models employing Decision Model and Notation (DMN) may be added to further segregate decision logic from process logic [3]. The Object Management Group (OMG) maintains both BPMN and DMN, and they're especially suited together.

Although BPMN and DMN provide a standardized understanding of process logic and decision logic, their specifications lack a formal language. This might result in misunderstandings or errors in the models, which can have a significant influence on the deployment from model to application in the PDA. Furthermore, one of the characteristics of PDA is their capacity to adapt to change; the first change will be in the model, which, if verified, will aid in the reduction of time and cost associated with application deployment. Verification of such process models can provide information about the model's correctness in terms of syntactic and structural irregularities.

To overcome these challenges, this paper proposes the integration of Colored Petri Nets (CPN) into the PDA development cycle. CPNs provide a mathematically grounded approach that allows for the precise validation of process models, enabling the identification of complex issues that are difficult to detect using BPMN alone. By leveraging the strengths of both BPMN for process visualization and CPN for formal verification, this method enhances the reliability and consistency of PDA.

2. Business Process Verification

2.1. BPMN and DMN

BPMN [2], offers a graphical representation for business process collaboration and design, enhancing both technical accuracy and stakeholder understanding.

Conversely, DMN [3] facilitates decision logic integration within these processes using decision tables for a clearer abstraction level, supported by the Simplified Friendly Enough Expression Language (S-FEEL). The synergy of BPMN and DMN streamlines operations by simplifying decision gateways into tasks, improving business rule clarity and efficiency.

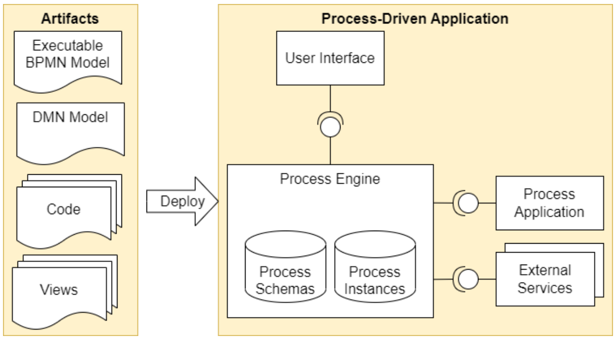

Figure 1. *Architecture of PDA modified from [4]*

PDA automate business process execution through process-aware IT, span ning from simple human tasks to intricate service orchestrations [1], [4]. As hybrid constructs, PDA blend source code with process models (e.g., BPMN, DMN), including both process and decision-making elements. To operationalize these models, technical integrations such as task assignment logic and system interaction mechanisms are essential. This includes coding automated task behaviors and creating interfaces for humancentric user tasks, ensuring comprehensive and executable process applications (Figure 1).

To execute a process based on such artifacts, a process engine is necessary. A process engine can be embedded as a library within the program or run as a separate service connecting with user interfaces, the implemented process applications, and external services.

2.2. Verification in business process

Verification in process building is a crucial step that ensures that a model is aligned with its intended characteristics before execution. This process identifies and rectifies potential issues early on, significantly enhancing system reliability. Despite the critical importance of such verification, many business process modeling tools and management systems often fall short, offering only basic syntax checks. This limitation leaves room for errors such as deadlocks and livelocks, which can severely impact the functionality and efficiency of the system.

CPN have emerged as an advanced tool for modeling and analyzing concurrent systems. They are particularly chosen for formalizing BPMN due to their support for formal verification methods like state-space and invariant analysis. CPN's hierarchical structure and colored tokens provide a robust framework for the detailed modeling of complex BPMN processes and data. This detailed modeling capability is essential to ensure the accuracy and reliability of process models.

For instance, in the context of Business Process Reengineering (BPR), where enterprises adapt to rapidly changing requirements to achieve new goals efficiently, the use of CPN for formal semantics of BPMN models can be invaluable. This approach allows for the detection of semantic errors in BPMN models, which are often difficult to identify and costly to fix later in the process. By mapping BPMN to CPN, organizations can leverage CPN analysis techniques to ensure the models are semantically correct before deployment, thereby enhancing the reliability and efficiency of the re-engineered business processes [5].

Moreover, CPN's combination of classical Petri-net and programming language functionalities enables the declaration of variables, data types, and data manipulation, including hierarchically structural representation. This feature is particularly useful in verifying the concurrent systems, as it allows for the computation of the reachability graph and the use of temporal logic queries to validate the model's properties [6].

3. Literature of Verification for PDA

The literature overview provided by Suchenia et al [7] and the formal semantics for business processes and business rule models introduced by Tuan et al [8] offer a comprehensive understanding of process anomalies, especially in BPMN models, and the tools used for verifying these models. Suchenia's work focuses on identifying and categorizing anomalies in BPMN-based processes, highlighting the importance of using a universal standard like BPMN for modeling processes, decisions, and software. This standard, while precise and graphical, can contain inaccuracies and semantic errors that are challenging to detect and costly to fix later. The paper emphasizes the need for formal semantics to ensure accurate interpretation and execution of BPMN models, addressing the lack of a satisfactory BPMN interpreter and the ambiguity in the BPMN semantics [7].

The study of Tuan et al introduces a three-step strategy for creating a CPN model from a BPMN model, developing CPM ML functions from DMN rules, and validating the final model with CPNTools. This approach aims to bridge the gap between BPMN and CPN, providing a formal semantics for BPMN models by mapping them to CPNs. This mapping is implemented as a tool that, when used with CPN analysis techniques, can detect semantic errors in BPMN models, offering a more reliable and accurate method for verifying process models [8].

Volker Stiehl's work on PDA provides a comprehensive foundation for understanding the architecture and concepts of PDA, emphasizing the importance of a well-defined architecture for implementing processdriven applications effectively. Stiehl's contributions span from the basic architecture of PDA to advanced concepts for architecture support, offering a holistic view of the PDA paradigm [1].

The discussions on feature development, static analysis, and data-flow anomalies in the articles by Schneid [9]–[11] delve into specific aspects of PDA, highlighting the challenges and solutions in developing and maintaining process-driven applications. These works underscore the importance of static analysis techniques, such as data flow analysis, for identifying potential issues in the application's code without executing it.

Data flow analysis is a static analysis technique that proves facts about a program or its fragment, making conclusions about all paths through the program while taking control flow into account. This technique is crucial to ensuring the reliability and security of PDA by identifying potential vulnerabilities and anomalies early in the development process.

Schffer's work extends the process-driven approach in the engineering domain by building and prototyping a minimum functional PDA architecture using an engineering use case. This approach demonstrates the practical application of PDA in real-world scenarios, highlighting the potential of PDA to streamline the development of complex engineering systems [4].

However, these works lack a verification model and technique for embedding BPMN-based business processes into the PDA development lifecycle. There is an issue with how verification is currently integrated into this lifecycle, which is significant given the importance of ensuring the correctness and reliability of software systems. To address this deficiency, a new technique and paradigm are proposed to fully integrate verification processes into the PDA development lifecycle. This integration makes verification a critical component from the initial design phase through to the final deployment. Implementing this approach will enable developers to detect and address potential issues early, thereby enhancing the overall quality and reliability of PDA.

4. Verification for Process-Driven Applications

4.1. PDA Development Lifecycle

PDA are subject to a lifecycle of continuous business process improvement that consists of four phases: design, implementation, control, and analysis [12], [13]. However, to help in the development of PDA, we include a verification phase in which features such as deadlocks, live locks, and other process abnormalities can be verified before being implemented (see Figure 2).

The business process extension life cycle of PDA is described in detail with the following phases:

- Phase 1. Process Design: When introducing new processes, we begin with process design. In this case, the outcome is a target state process model.
- Phase 2. Process Verification: Verifying the process with verification tools enables the assessment of anomalies in the process, giving updated data on the current process state for redesign. Verification contributes to the development of a strong process by decreasing the weaknesses that appear during the process implementation phase.
- Phase 3. Process Implementation: As a software project, we implement the process model of the target state process. This procedure can be automated, software can be created or modified. The process execution results in a current-state process that matches the target-state process model. Implementing outside the business process requires additional artifacts, such as source code and views, as well as links to other services and software when deployed in a process engine (View Figure 1).
- Phase 4. Process controlling: Process monitoring is continuing, but it exposes more about the process's ongoing operation. Process control's most critical duties are to continually monitor individual process instances and evaluate key data so that weaknesses may be found as soon as feasible.
- Phase 5. Process analysis: We also performed process analysis, since continuous process control revealed process weaknesses that were difficult to correct. This enables a more thorough redesign of the new process.

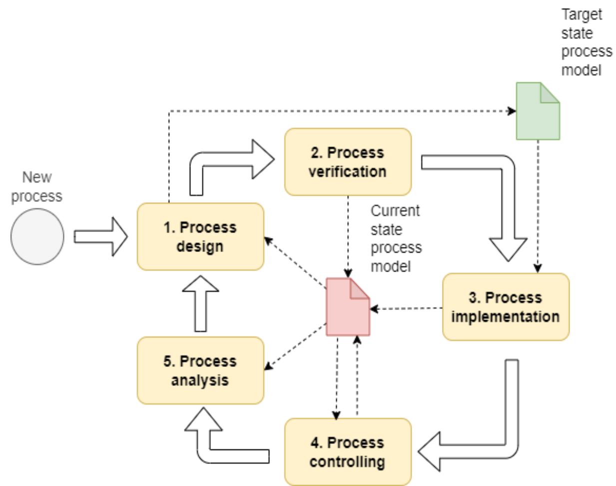

Figure 2. *PDA Development Lifecycle*

4.2. Model for Integrating Process Verification in PDA development

In the development of PDA, verification is crucial for ensuring that the processes and rules function both correctly and efficiently. To support this, we have developed a comprehensive technical model, which is detailed in Figure 3. This model represents the primary contribution of our paper. It is a comprehensive model to integrate process verification within the development lifecycle of PDA, using a combination of BPMN for process modeling and DMN for decision logic. Using tools like Camunda Modeler for initial modeling and custom developed tools for translating and verifying these models into Colored Petri Nets, this approach enhances the accuracy and efficiency of PDA by ensuring rigorous validation before deployment.

Initially, the Camunda Modeler [14] is used for the creation of business process and decision rule models. This step produces BPMN and DMN files that encapsulate the desired processes and business logic. Following this, the BP2CPN tool translates these models into CPN, represented in .cpn file format, enabling a more formal verification of the business logic and process flow.

To verify these CPN models, CPNTools [15] is used in conjunction with our custom-developed BPVerification tool. This combination allows for an in-depth analysis of the process models, identifying potential issues such as deadlocks or live locks. The outcome of this analysis is a detailed verification report, highlighting areas that may require refinement.

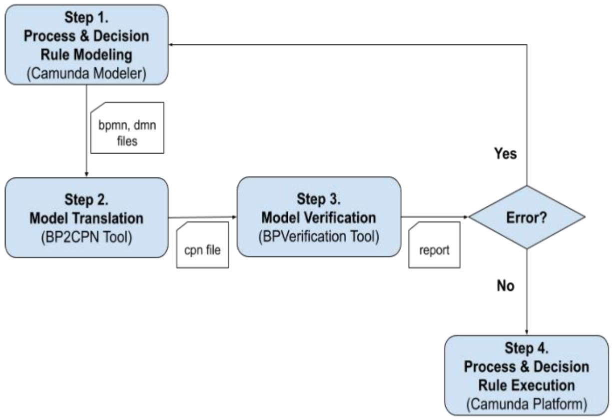

Figure 3. *Technical Model for Verification*

Based on the insights from the verification report, necessary adjustments are made to the process models and business rules. The final, verified models are then deployed on the Camunda Platform [16], bringing the PDA closer to execution. This step marks the culmination of the verification process, transitioning from model validation to the actual deployment and operation of the application.

Throughout this process, two bespoke tools, BP2CPN and BPVerification [8], [17], are integrated directly into the Camunda Modeler via a specifically designed plugin, streamlining the verification workflow. This integration not only enhances the efficiency of the verification process but also ensures that the PDA developed are robust, reliable, and ready for deployment in a real-world setting. The model will be implemented in detail through the following steps:

• Step 1: The business requirements of the application will be modeled into process models and decision rules by the Camunda Modeler tool, bpmn/dmn files containing processes and business rules are generated at this step.

• Step 2: The business processes and rules in bpmn/dmn files will be translated into a CPN using the BP2CPN tool, which is saved in the .cpn file format.

• Step 3: Business process and rules represented by CPN, will be connected to CPN Tools through BPVerification tool to perform verification, the result of these steps is a verification report including deadlock, live-lock and other problems will be generated.

• Step 4: The business process and rules are verified and adjusted based on the report from step 4, which will be deployed on the Camunda platform to build the application.

This model for integrating process verification in the development of PDA marks a significant enhancement over traditional approaches. Unlike conventional models that typically progress linearly from Step 1 to Step 4, this revised model introduces critical verification stages at Steps 2 and 3. These stages ensure rigorous verification of the process models before their final deployment. By incorporating these intermediary verification steps, the model allows for early detection and correction of any errors or inconsistencies, significantly reducing the risks associated with direct implementation.

5. Case Study: Loan Application Process

This section presents a case study of a loan application process within a financial institution to demonstrate the practical implementation and advantages of the proposed BPMN + CPN hybrid approach. The case study highlights the steps involved in developing, verifying, and deploying the model, followed by an analysis of the performance metrics and overall results.

5.1. Steps of Development and Verification

• Step 1: Modeling the Loan Application Process

The loan application process was initially modeled using BPMN in Camunda Modeler. The process included key stages such as "Submit Loan Application," "Perform Credit Check," and "Evaluate Loan Eligibility" (Figure 4) Decision points related to credit score and financial background were modeled using DMN to ensure transparency in decision logic and separation from process control flows (View Figure 5).

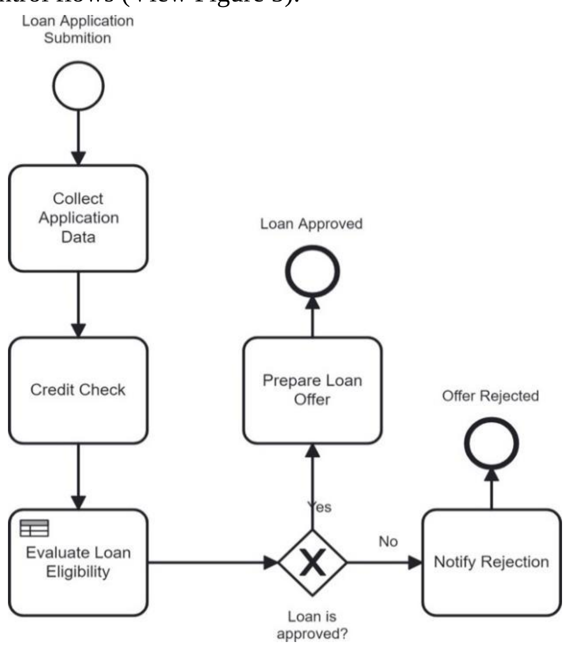

Figure 4. *BPMN/DMN Model of Loan Application Process*

#	Credit Score	Annual Income	Requested Loan Amount	Eligibility
1	>= 700	>= $50,000	<= $200,000	Eligible
2	>= 650	>= $75,000	<= $150,000	Eligible
3	< 650	Any	Any	Not Eligible
4	Any	< $50,000	> $200,000	Not Eligible

Figure 5. *DMN decision table of "Evaluate Loan Eligibility" Business Rule Task*

• Step 2: Translating to Colored Petri Nets (CPN)

Following the creation of the BPMN/DMN models, they are translated into a Colored Petri Net (CPN) model with the BP2CPN tool. This CPN model represents the loan process in a formalized manner, enabling verification of its logic and flow (Figure 6). The translation to a CPN model allows a detailed examination of the process under various scenarios, ensuring the robustness of the logic applied throughout the loan application process. The result is a.cpn file format of the loan application process.

• Step 3: Verification with CPNTools

The CPN model was analyzed and verified using CPNTools. This step was essential for identifying and correcting potential errors before deployment. The verification process focused on ensuring the correctness of control flows and decision logic, identifying unreachable tasks and improper data handling. The results allowed for refinements to be made to the BPMN model, ensuring it was error-free before proceeding to the deployment phase.

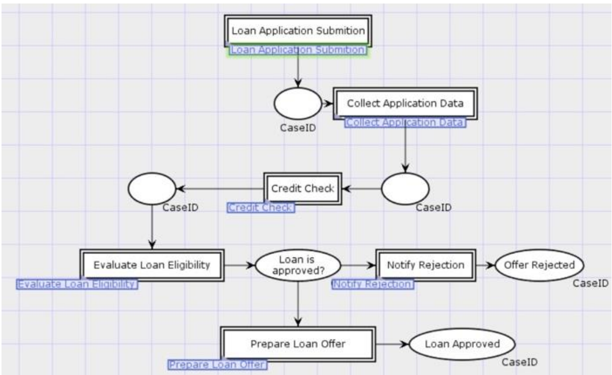

Figure 6. *CPN Model of loan process*

Analysis Summary
- **Total States**: 156
- **Total Transitions**: 255
- **Total Arcs**: 410

Boundedness Properties
- **Status**: Bounded

Home Properties
- **Home States Identified**: Yes

Liveness Properties
- **Live Transitions**: All transitions are live.

Fairness Properties
- **Fair Transitions**: All

Deadlocks and Livelocks
- **Deadlocks Detected**: No
- **Livelocks Detected**: No

Figure 7. *Verification Result Report of loan process*

• Step 4: Deployment on Camunda Platform

After the verification process, the refined BPMN and DMN models were deployed on the Camunda platform. The Camunda BPM engine was used to execute the verified loan application process in a live environment, handling real-time data for loan applications. By integrating BPMN and DMN models into a workflow engine like Camunda, the organization was able to manage, monitor, and optimize the process in real-time. The deployment also included continuous monitoring of the process performance to ensure that the workflow ran smoothly without encountering execution errors or bottlenecks.

5.2. Revised Performance Metrics

To evaluate the performance of the proposed BPMN + CPN approach, we applied four key metrics: Error Detection Rate (EDR), Model Execution Time (T_m), Reduction in Errors (R_e), and Scalability (S_k).

• EDR: The Error Detection Rate measures the percentage of errors detected during the verification process [17]. It is calculated as:

$$EDR = \frac{E_d}{E_t} \times 100\%$$

Where:

○ E_d is the number of detected errors.

○ E_t is the total number of errors present in the model.

• Model Execution Time (T_m): The Model Execution Time measures the time required to verify the model [18]. It includes both translation time and verification time:

$$T_{\text{m}} = T_{\text{translation}} + T_{\text{verification}}$$

• Reduction in Errors (R_e): The Reduction in Errors [19] measures the percentage decrease in errors after verification:

$$R_{\text{e}} = \frac{E_{\text{initial}} - E_{\text{final}}}{E_{\text{initial}}} \times 100\%$$

Where:

○ E_{initial} is the initial number of errors before verification.

○ E_{final} is the number of errors remaining after verification.

• Scalability (S_k): The Scalability metric evaluates the system's ability to handle increases in model complexity, measured by the growth in verification time relative to the increase in model size.

Table 1. *Performance Results of Case Study: Loan Application Process*

Metric	BPMN-only	BPMN + CPN
Error Detection Rate (EDR)	40%	85%
Model Execution Time (T_m)	0.8 sec	1.2 sec
Reduction in Errors (R_e)	20%	80%
Scalability (S_k)	Low	High

This case study validates the BPMN + CPN approach as an effective method for developing robust, error-free Process-Driven Applications. The combination of BPMN's intuitive process modeling and CPN's formal verification significantly improved the reliability, scalability, and overall performance of the loan application process (View Table 1). The empirical results demonstrate the approach's capacity to enhance error detection, reduce operational risks, and ensure successful process deployment.

6. Conclusion

This paper presents a novel approach to integrating formal verification into the development lifecycle of PDA by combining BPMN with CPN. The proposed method addresses the limitations of BPMN, which, despite its widespread use for modeling business processes, lacks formal semantics and is prone to errors such as deadlocks, livelocks, and structural inconsistencies. By introducing CPN into the verification phase, our approach ensures early detection of errors, leading to more reliable and robust process models.

The primary contribution of this paper lies in the integration of BPMN and CPN for process verification, offering a structured framework that enhances error detection and improves the overall reliability of PDA. The application of this method to a real-world loan application process demonstrated its effectiveness in reducing errors by 80% and increasing the error detection rate to 85%.

These results show that the BPMN + CPN hybrid model significantly outperforms traditional BPMN-only verification in both accuracy and scalability, providing a viable solution for businesses seeking to improve the quality and reliability of their process-driven systems.

Additionally, the paper introduces revised performance metrics to quantify the efficiency and reliability of the proposed method, providing a solid foundation for future research and further optimization of process verification techniques. These metrics allow for a more comprehensive evaluation of both the correctness and scalability of business processes.

In summary, this work makes a meaningful contribution to the field of process verification by offering a robust, scalable, and efficient method to enhance the reliability of PDA. The integration of BPMN and CPN sets a new standard for formal verification in business process management, ensuring more reliable applications with fewer errors at deployment.

REFERENCES

[1] V. Stiehl, *Process-driven applications with BPMN.* Switzerland: Springer Cham, 2014.

[2] OMG, "About the Business Process Model and Notation Specification Version 2.0.2", *omg.org*, June 2023. [Online]. Available: https://www.omg.org/spec/BPMN/ [Accessed April 25, 2024].

[3] OMG, "About the Decision Model and Notation Specification Version 1.3", *omg.org*, June 2023. [Online]. Available: https://www.omg.org/spec/DMN/ [Accessed April 25, 2024].

[4] E. Schaffer *et al*., "Process-driven approach within the engineering domain by combining business process model and notation (bpmn) with process engines", *Procedia CIRP*, vol. 96-2021, pp. 207-212, 2021. https://doi.org/10.1016/j.procir.2021.01.076.

[5] T. Krauter *et al*., "A higher order transformation approach to the formalization and analysis of bpmn using graph transformation systems", *ArXiv*, vol. abs/2311.05243, 2023. https://doi.org/10.48550/arXiv.2311.05243.

[6] C. Dechsupa *et al*., "Compositional formal verification for business process models with heterogeneous notations using colored petri net", in *Proceedings of the International Multi Conference of Engineers and Computer Scientists.* Hong Kong, 2019, pp. 565-570.

[7] A. Suchenia *et al*, "Overview of verification tools for business process models," in *Proceedings of Federated Conference on Computer Science and Information System.* Poland, 2017, pp. 295-302.

[8] N. T. Tuan *et al*, "Formalization of Business Processes and Business Rules Model using Colored Petri Nets", in *The 5th International Conference on Future Networks & Distributed Systems (ICFNDS 2021)*, Dubai, United Arab Emirates, 2021, pp. 42-47.

[9] K. Schneid *et al*, "Static analysis of bpmn-based process-driven applications" in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, Limassol, Cyprus, 2019, pp. 66-74.

[10] K. Schneid *et al*, "Feature development in bpmn-based process-driven applications", in *Business Process Management Forum. BPM 2020. Lecture Notes in Business Information Processing, vol 392*, Seville, Spain, 2020, pp. 35-50, https://doi.org/10.1007/978-3-030-58638-6_3.

[11] K. Schneid *et al*, "Uncovering data-flow anomalies in bpmn-based process-driven applications", in *Proceedings of the 36th Annual* ACM *Symposium on Applied Computing*, Korea, 2021, pp. 1504–1512.

[12] J. Freund and B. Rucker, *Real-Life BPMN*, 4th edition. USA: Independently Published, 2019.

[13] M. Weske, *Business Process Management: Concepts, Languages, Architectures*. Germany: Springer Berlin, 2019.

[14] CAMUNDA, "Camunda Modeler", *Camunda.com*, [Online]. Available: https://camunda.com/products/camundaplatform/modeler/ [Accessed April 25, 2024].

[15] CPNTools, "CPN Tools A tool for editing, simulating, and analyzing Colored Petri nets", *CPNTools.org*, [Online]. Available: https://cpntools.org/, [Accessed April 25, 2024].

[16] CAMUNDA, "Camunda Platform", *Camunda.com*, [Online]. Available: https://camunda.com/products/camunda-platform/, [Accessed April 25, 2024].

[17] N. T. Tuan *et al*, "VeBPRu: A toolchain for formally verifying business processes and business rules", in *Proceedings of the 2023 8th International Conference on Intelligent Information Technology*, Danang, Vietnam, 2023, pp. 15–19, https://doi.org/10.1145/3591569.3591572.

[18] L. Li and F. Dai, "Transformation and visualization of BPMN models to Petri nets", *IOP Publishing*, vol. 186, no. 5, p. 012047, 2018.

[19] L. Zonghua and Y. Zhengwei, "A Petri Nets Evolution Method that Supports BPMN Model Changes" *Scientific Programming*, vol. 2021, pp. 1-16, 2021. doi: 10.1155/2021/6610795.

[20] M. M. Ibrahim, "Formal Semantics of BPMN Process Models Using CPN", *International Review on Computers and Software*, vol. 18, no. 2, pp. 16-56, 2023. doi: 10.15866/irecos.v18i2.24401.