



HAL
open science

Anomaly Detection in Blockchain-Enabled Supply Chain: An Ontological Approach

Tahani Abu Musa, Abdelaziz Bouras

► **To cite this version:**

Tahani Abu Musa, Abdelaziz Bouras. Anomaly Detection in Blockchain-Enabled Supply Chain: An Ontological Approach. 18th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2021, Curitiba, Brazil. pp.253-266, 10.1007/978-3-030-94335-6_18 . hal-04186130

HAL Id: hal-04186130

<https://inria.hal.science/hal-04186130v1>

Submitted on 23 Aug 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Anomaly Detection in Blockchain-enabled Supply Chain: An Ontological Approach

Tahani Abu Musa ^[1][0000-0003-0661-4962] and Prof. Abdelaziz Bouras ^[2] [0000-0001-5765-1259]

¹ Department of Computer Science and Engineering, College of Engineering (Qatar University)
Doha, Qatar
ta090001@student.qu.edu.qa.

² Department of Computer Science and Engineering, College of Engineering (Qatar University)
Doha, Qatar
abdelaziz.bouras@qu.edu.qa

Abstract. In our work, we propose an anomaly detection framework, for detecting anomalous transactions in business processes from transaction event logs. Such a framework will help enhance the accuracy of anomaly detection in the global Supply Chain, improve the multi-level business processes workflow in the Supply Chain domain, and will optimize the processes in the Supply Chain in terms of security and automation. In the proposed work, Ontology is utilized to provide anomaly classification in business transactions, based on crafted SWRL rules for that purpose. Our work has been evaluated based on logs generated from simulating a generic business process model related to a procurement scenario, and the findings show that our framework was able to detect and classify anomalous transactions from those logs.

Keywords: Anomaly Detection, Business Process, Supply Chain, Blockchain.

1 Introduction

Supply Chain (SC) is defined as a network of organizations and entities involved in all the processes and activities related to manufacturing to transporting products and services from source to eventually the ultimate customers [1]. Nowadays, Supply Chain became more complex, due to the involvement of various stakeholders in the global context, and this may increase the complexity of interacting systems, which make them face several vulnerabilities such as downtime, security breaches and forgery, and hence raises concerns related to security, reliability, and treatability of information.

Such limitations of the existing Supply Chain Management systems can be mitigated by incorporating Blockchain, or distributed ledger technology, that was first introduced in 2008 [2], implemented in 2009 [3], and can be thought of as a tamper evident, tamper resistant digital ledger implemented in a distributed fashion and usually without a central authority. Hence, Supply Chain Management Systems can benefit from the Blockchain Technology when it comes to traceability, transparency, and data provenance [4].

The second generation of Blockchain, Blockchain 2.0 [5], allowed for programmable transactions, self-executing code that are triggered when certain conditions are met, known as Smart Contracts [6].

Smart contracts are programs that run on the Blockchain and add blocks to the network whenever certain conditions are met. They are often defined by translating actual legal contracts, or business processes of a given domain of interest into programs or code, to enforce the legal contracts onto the Blockchain. A smart contract can be thought of like stored procedures in databases, and usually stored on the Blockchain as scripts and executed in accordance with the data that is fed to them to produce output that is anticipated from the original contract. In other words, smart contracts manage the transactions over the network when executed either fully or partially according to the current input fed to them, acting as digital agreement among the transacting parties [7]. Incorporating Smart Contracts in the Supply Chain domain would be of a revolutionary solution, in which they will automate the several complex aspects in traditional Supply Chain systems, related to paperwork and manual workflows. In this context, Smart Contracts can contribute to lowering costs, reduce administration fees, streamline workflows, and securing the Supply Chain Management Systems.

On the other hand, a business process may be defined as a group of linked activities produced for a particular purpose [8]. A business process holds a number of attributes: has a particular goal, needs a certain input, produces a certain output, has a series of events or activities carried out in a predefined order, might need some resources and involve a number of owners [9]. There are several representations and notations of business processes, including: Unified Modeling Language (UML), Business Process Model Notation (BPMN), Event-driven process chain (EPC) and Petri Net Markup Language (PNML) [8].

Business process mining or process mining is an emerging research area that aims at analyzing Business Process Models (BPM), by extracting knowledge from logs generated by various information systems. Such knowledge can support auditing, monitoring and analysis of business activities for future improvements [10]. It is also useful when detecting potential exceptions or anomalies in the logged workflow, and hence help identify the source of such exceptions and modify them accordingly.

On the other hand, ontologies can contribute to the representation of the business process model of interest, in which they play a major role in representing the domain in a uniform and formal representation, that allows both humans and computers to access structured and well-defined knowledge effectively, as well as allows for automatic reasoning and inferring new knowledge easily [11].

The notion of ontology was defined in [12] as a “formal, explicit specification of a shared conceptualization”. It captures the concepts of a given domain, along with their relations and associated axioms, which allows for interoperability and performing automatic reasoning and inferring new knowledge about that domain [13, 14, 15, 27]. Ontology is developed with “OWL”, Ontology Web Language, in which the expressivity and reasoning capabilities can be expressed in terms of SWRL (Semantic Web Rule Language), by providing high-level syntax for Horn-like rules, that are expressed in

terms of Ontology concepts (i.e. classes, properties, and individuals) [16]. In our framework, SWRL rules were defined to classify the types of anomalies in each Business process, by analyzing the relationships between the events of that process. An example of a SWRL Rule to classify a consistent process is given below:

$$\begin{aligned} & \text{Process}(?P), \text{StartEvent}(?ST), \text{beginsWith}(?P, ?ST), \text{followedBy}(?ST, ?S), \\ & \text{EndEvent}(?EE), \text{endsWith}(?P, ?EE) \rightarrow \text{ConsistentProcess}(?P) \end{aligned} \quad (1)$$

In this paper, we would like to propose an anomaly detection framework, for detecting anomalous transactions in business processes from transaction event logs. The proposed work utilizes Ontologies to represent the domain of interest, that will help provide anomaly classification in business transactions, based on crafted SWRL rules for that purpose. Such framework will help enhance the accuracy of anomaly detection in the global Supply Chain, improve the multi-level business processes workflow in the Supply Chain domain, and will optimize the processes in the Supply Chain in terms of security and automation. The three main steps of this research are:

1. Convert the BPMN Model into an ontology.
2. Simulate the business process and generate event logs that will be adopted and used for evaluating the approach.
3. Build the SWRL rules needed to identify both consistent and anomalous processes.

The structure of this research is organized as follows: Section 2 survey some related work in literature, Section 3 describes our proposed approach, Section 4 demonstrates results and analysis, and Section 5 presents a discussion of our research findings and future work to be done in this area.

2 Literature Review

In this section, we survey some of the related work in the domain of Business Process Modeling using ontologies, anomaly detection and classification in business processes.

2.1 Ontologies for Business Process Modeling

Two BPMN ontologies were researched before in literature [17]. The first, sBPMN that incorporated the last release of BPMN 1.0. The classes within this ontology represent the elements of BPMN, examples include: Swimlanes, Artifacts, Processes, Flow Objects and Connecting Objects are examples of such classes. Such ontology includes around 95 defined classes, and 50 defined axioms [18]. The second BPMN ontology was presented by Francescomarino, et al., that was influenced by the BPMN 1.1 elements [19]. Such ontology includes around 95 defined classes, 108 object properties, 70 data properties in addition to 439 axioms [19]. Another ontology that serves for the purpose of BPMN modeling in BMNO, presented in [20], but according to [21] such ontology does not provide a full description of all the BPMN specification in terms of asserted properties. This was justified by the author because it focused more on the execution behavioral aspects of the business processes [21].

Thus, the old releases of BPMN influenced the creation of BPMNO and sBPMN, but with the release of BPMN 2.0, the BPMN metamodel served as a basis for a more powerful ontology that benefits from the expressivity of such new notation and helps better representing the BPMN specifications and standards.

In a parallel research, another ontology was presented, the Publish Workflow Ontology (PWO), which was implemented mainly to satisfy workflow specifications and requirements. PWO focuses on the logical steps of a given workflow and uses the notion of patterns or building blocks in representing such Workflow [22]. Table 1. summarizes the patterns used in PWO, that influenced our work.

In [23], the work of modeling industrial business processes for querying and retrieving was presented. It is a combination of hybrid program that presents modular ontology design, consistent ontology design of each module and cohesive control flow for process and its sub-processes, using both OWL-DL and SWRL, which also influenced our work when developing the ontology.

2.2 Anomaly Detection in Business Processes using ontologies

In the area of utilizing ontologies in anomaly detection in business processes, [24] presented an ontology-based process modeling to capture fraud in credit application business process, with the aid of Multi-Level Class Association Rule Learning (ML-CARL).

[25] also presented an anomaly detection framework in business processes via both process mining and fuzzy association rule learning. The fuzzy multi-attribute decision making is utilized to detect anomalies, and process mining is used to analyze the conformance between both the recorded event logs and the typical operational procedures of the business model.

Table 1. The main patterns of the PWO Ontology

PWO entity	Pattern entity
Workflow	Plan (basic plan description, via basic plan)
WorkflowExecution	PlanExecution (basic plan execution, via basic plan)
Step	Task (task role, via control flow)
Action	Action (task execution and plan execution) TimeIndexedSituation (time-indexed situation)
hasStep	definesTask (basic plan description)
hasFirstStep	definesTask (basic plan description)
executes	satisfies (basic plan)
hasNextStep	directlyPrecedes (sequence)
hasPreviousStep	directlyFollows (sequence)
needs	isDefinedIn o describes (basic plan description)
produces	isExecutedIn (o hasParticipant) (task execution and participation)

In [26], an Ontology-based Approach for Failure Classification in Predictive Maintenance Using Fuzzy C-means and SWRL Rules, in which the employed clustering techniques used to decide on the level of criticality of such failures based on historical data of machines, and ontologies use the results of those techniques for predicting both the time and criticality of such failures.

3 Proposed Approach

This section illustrates the proposed approach for modeling the business process and defining the SWRL rules for detecting and classifying the anomalous processes. As stated in section 1, the proposed approach is composed of three steps, illustrated in **Fig. 1**.

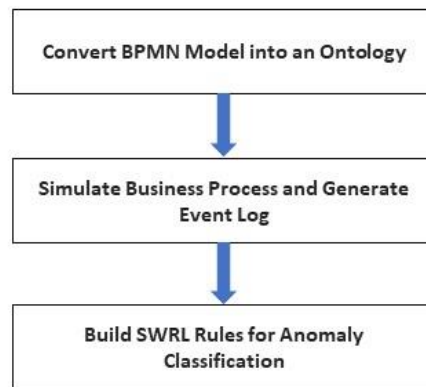


Fig. 1. Proposed Approach

3.1 Convert the BPMN Model into an Ontology

For demonstration purpose in this paper, we focus on a generic business process for a procurement scenario illustrated in **Fig 2**. (The real work is also performed on similar scenarios provided by our project industrial stakeholders). In this context, the class hierarchy of the constructed ontology is illustrated in **Fig. 3**, using the popular Ontology Editor Protégé 5.5.

To better represent and control the sequence of events in each process in our ontology, the following constraints must be fulfilled:

Constraint 1: *Every process must have a clear start and end events.*

This constraint will ensure that all process are consistent and conform to the business model.

Constraint 2: *In each process, each event must have at least one connection to other events.*

This constraint will ensure that a given process is built from connected events.

Constraint 3: No Cyclic dependency (i.e. loops) between events within a process.

Otherwise, such a process will be classified as anomalous with one or more events reworked.

Constraint 4: To ensure simplicity of events within a business process, an event must have one successor and one predecessor.

In case of multiple successors and/or predecessors, inclusive or exclusive gateways (AND, XOR gates) must be specified either to join or split those events.

Table 2. describes both object and data properties defined for the Procurement Process Ontology.

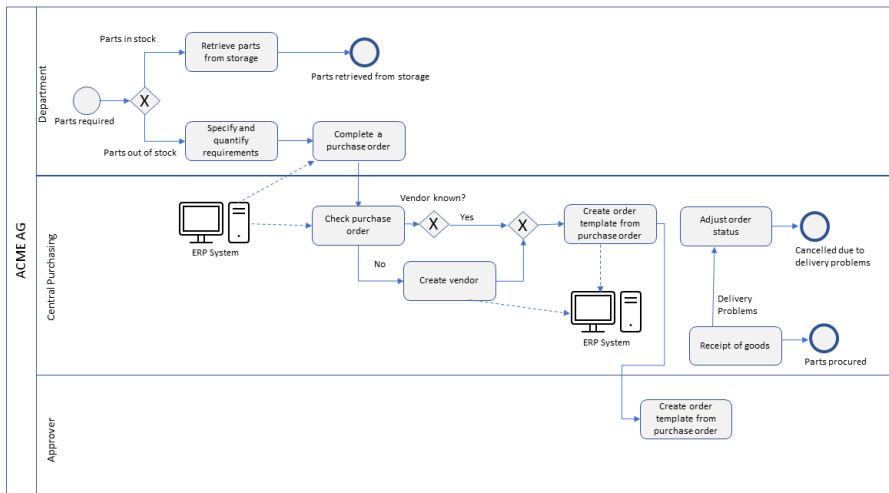


Fig. 2. Procurement Business Process

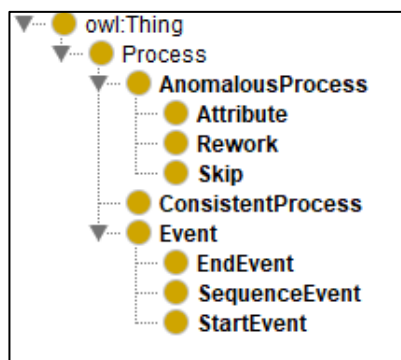


Fig. 3. Class Hierarchy of the Procurement Ontology

Table 2. Asserted Object and Data Properties for the Procurement Ontology

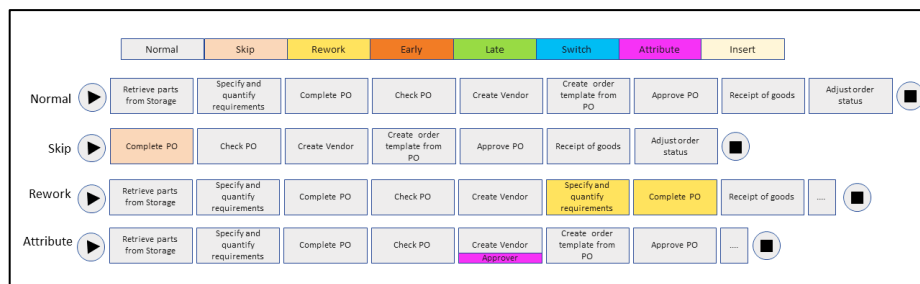
Property	Type	Description
beginsWith (P1, E1)	Object	Process P1 begins with start event E1
containedIn (SE1, P1)	Object	Sequence event SE1 is contained in Process P1
Contains (P2, E2)	Object	Process P2 contains sequence event E2
endsWith (P1, EE)	Object	Process P1 ends with end event EE
followedBy (E1, E2)	Object	Event E1 is followed by Event E2
PrecededBy (E2, E1)	Object	Event E2 is preceded by Event E1
hasEventName (E1, "ProcureParts")	Data	Event E1 is named "ProcureParts"
hasOwner (E1, "Department")	Data	Owner of Event E1 is "Department"
hasPName (P1, "ProcurementProcess")	Data	Process P1 is named "ProcurementProcess"
hasTimestamp (P1, "2021-01-26T21:08:18.849+03:00")	Data	Process P1 is timestamped with value of "2021-01-26T21:08:18.849+03:00"
pairsWith (E1, "Yes")	Data	Event E1 is associated with the yes part of the condition of XOR gateway.

Anomaly types:

The proposed model will classify the following types of anomalies in the workflow:

- **Skip:** An event in the sequence was not executed.
- **Rework:** An event was executed more than once.
- **Attribute:** An event with incorrect attribute value.

Fig. 4. Illustrates these types, compared to the normal workflow.

**Fig. 4.** Types of Anomalies to detect and classify**3.2 Simulate Business Process and Generate Event Logs**

For this purpose, we used ProM 6.10¹, that is a well-known Process Mining tool developed by Developed by Process Mining Group, Math and CS department, Eindhoven

¹ <http://www.promtools.org/doku.php?id=prom610>

University of Technology, and has several capabilities related to simulation of business processes, as well as log generation capabilities. Table 3. provides a snapshot of a generated event log resulting from the simulation.

Table 3. Snapshot of a generated Event Log

ID	Process	Activity	Timestamp
1		Parts Required	2010-12-30T11:02:00.000+01:00
1		Retrieve Parts from Storage	2010-12-31T10:06:00.000+01:00
1		Parts Retrieved from Storage	2011-01-05T15:12:00.000+01:00
2		Parts Required	2021-01-26T21:06:18.849+03:00
2		Specify Quantify Requirements	2021-01-26T21:12:18.849+03:00
2		Specify Quantify Requirements	2021-01-26T21:13:18.849+03:00
2		Complete PO	2021-01-27T21:23:03.707+03:00
2		Check PO	2021-01-27T21:25:03.707+03:00
2	
3		Parts Required	2021-01-30T22:18:32.555+03:00
3		Specify Quantify Requirements	2021-01-30T22:19:32.555+03:00
3		Complete PO	2021-01-31T22:28:32.555+03:00
3		Check PO	2021-02-01T22:14:41.262+03:00
3		Specify Quantify Requirements	2021-02-02T22:27:41.262+03:00
3	
4		Retrieve Parts from Storage	2021-02-22T23:03:01.460+03:00
4		Parts Retrieved from Storage	2021-02-22T23:04:01.460+03:00
4	
5		Parts Required	2021-03-26T23:00:32.002+03:00
5		Retrieve Parts from Storage	2021-03-26T23:00:38.002+03:00
5	

3.3 Build SWRL Rules for Anomaly Classification

In this sub-section, we present the constructed SWRL Rules that will be used for classifying consistent and anomalous processes, which are also based on the constraints discussed in 3.1.

The following two rules are used to classify reworked processes, rule (2) when a single processes is reworked, whereas rule (3) when there is a cyclic dependency between more than two processes:

$$\text{Process}(?P), \text{StartEvent}(?ST), \text{beginsWith}(?P, ?ST), \text{followedBy}(?ST, ?S), \text{followedBy}(?S, ?S), \text{EndEvent}(?EE), \text{endsWith}(?P, ?EE) \rightarrow \text{Rework}(?P) \quad (2)$$

$$\text{Process}(?P), \text{StartEvent}(?ST), \text{beginsWith}(?P, ?ST), \text{followedBy}(?ST, ?S), \text{followedBy}(?S, ?S2), \text{followedBy}(?S2, ?S3), \text{followedBy}(?S3, ?S), \text{EndEvent}(?EE), \text{endsWith}(?P, ?EE) \rightarrow \text{Rework}(?P) \quad (3)$$

The following rule is used to classify a skipped process:

$$\begin{aligned} & \text{Process}(?P), \text{StartEvent}(?ST), \text{beginsWith}(?P, ?ST), \text{hasEventName}(?ST, ?Name), \\ & \text{notEqual}(?Name, "P4_PartsRequired"), \text{EndEvent}(?EE), \text{endsWith}(?P, ?EE) \rightarrow \\ & \text{Skip}(?P) \end{aligned} \quad (4)$$

Finally, the following rule is used to classify an anomalous process “Process 5” that has a wrong attribute value for the hasOwner data property, “Approver” instead of “Department”:

$$\begin{aligned} & \text{Rule: Process}(?P), \text{StartEvent}(?ST), \text{beginsWith}(?P, ?ST), \text{followedBy}(?ST, ?S), \text{has-} \\ & \text{Owner}(?S, ?Name), \text{notEqual}(?Name, "Department"^\wedge\text{xsd:string}), \text{EndEvent}(?EE), \\ & \text{endsWith}(?P, ?EE) \rightarrow \text{Attribute}(?P) \end{aligned} \quad (5)$$

4 Results and Analysis

After the construction of rules to detect the abovementioned anomalies, we populated the ontology with five sample processes, extracted from log of Table 3. Each process has its own sequence of events, denoted by Px_EventName. Fig. 5. Illustrates the asserted individuals of processes and their corresponding events.

The next step will be running the Pellet² reasoner to classify those processes into one of the classes presented in the procurement ontology, with the aid of SWRL rules defined previously for this purpose. The following three use cases demonstrate the ability of our framework to classify the asserted business onto the corresponding classes in our ontology.

4.1 Scenario 1, Classifying a consistent process.

Table 4. summaries the Business Process “Process1” individual, asserted with the following data and object properties among its events, based on its presented event in the log of Table 3. Based on those properties, ‘Process1’ is classified as a Consistent Process, according to SWRL rule (1). The result of classification after running the reasoner is illustrated in Fig. 6.

² <https://www.w3.org/2001/sw/wiki/Pellet>

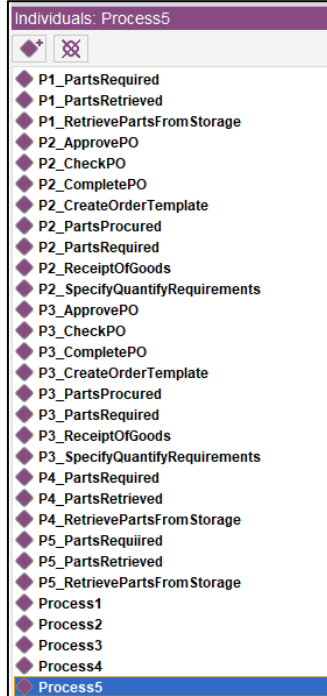


Fig. 5. Individuals of Processes 1 to 5, with their asserted events

Table 4. Properties of the ‘Process1’ Individual

Individual Name	Property	Value	Class
Process1	beginsWith	P1_PartsRequired	‘Start Event’
	endsWith	P1_PartsRetrieved	‘End Event’
P1_PartsRequired	followedBy	P1_RetrieveParts-FromStorage	‘Sequence’ Event
	followedBy	P1_PartsRetrieved	‘End Event’
P1_RetrievePartsFromStorage			

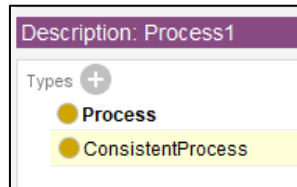


Fig. 6. Result of classifying ‘Process1’ Individual

4.2 Scenario 2, Classifying a Process with Reworked Event.

Table 5. summaries the Business Process “Process2” individual, asserted with the following data and object properties among its events, based on its presented event in the log of Table 3. Based on those properties, ‘Process2’ is classified as a Rework Process, according to SWRL rule (2), because the event ‘P2_Specify Quantity Requirements’ was executed twice. The result of classification after running the reasoner is illustrated in Fig. 7, in which ‘Process2’ was classified as a consistent business process (has a clear start and clear end events – constraints 1), and as a Rework process too, because of repeating the event ‘P2_Specify Quantity Requirements’.

Table 5. Table 6. Properties of the ‘Process2’ Individual

Individual Name	Property	Value	Class
Process2	beginsWith	P2_PartsRequired	‘Start Event’
	endsWith	P2_PartsProcured	‘End Event’
P2_PartsRequired	followedBy	P2_Specify Quantity Requirements	‘Sequence’ Event
	followedBy	Specify Quantity Requirements	‘End Event’
P2_Specify Quantity Requirements			

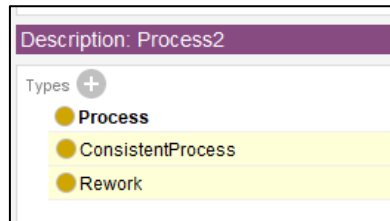


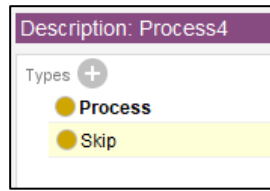
Fig. 7. Result of classifying ‘Process2’ Individual

4.3 Scenario 3, Classifying a Process with Skipped Event.

Table 6. summaries the Business Process “Process4” individual, asserted with the following data and object properties among its events, based on its presented event in the log of Table 3. Based on those properties, ‘Process4’ is classified as a Skip Process, according to SWRL rule (4), because it started with the sequence event “P4_Retrieve-PartsFromStorage” instead of the start event “P4_Parts_Required”. The result of classification after running the reasoner is illustrated in Fig. 8, in which ‘Process4’ was classified as Skip Process, because it skipped the start event “P4_Parts_Required”.

Table 7. Properties of the ‘Process4’ Individual

Individual Name	Property	Value	Class
Process4	beginsWith	P4_RetrieveParts- FromStorage	‘Sequence Event’
	endsWith	P4_PartsRetreived	‘End Event’
	hasEvent- Name	“ P4_Retrieve- PartsFromStor- age”	string
P4_RetrievePartsFromStorage			

**Fig. 8.** Result of classifying ‘Process4’ Individual

5 Conclusion

The major contribution of this research was modeling an example business process representing a procurement scenario, that was then used to classify business processes into consistent and/or anomalous (Skip/Rework or Attribute) processes accordingly. A generated event log from simulating the scenario was adopted to populate the ontology, with such processes and classify them successfully. In this work the constructed ontology serves as a knowledge base for discovered anomalies for future reporting, classification, and analysis. As a future work we aim at extending the ontology to be able to classify another type of anomalies such as:

- Early: An event was executed too early.
- Late: An event was executed too late.
- Insert: A random event was inserted to the workflow, and
- Switch: Two events swapped their order of execution.

We may also incorporate such framework as a layer in the architecture of the Blockchain-enabled Supply Chain Management System for anomaly classification. We may also incorporate deep learning techniques for anomaly detection in real world event logs, extracted from the Supply Chain Management Information System of the project stakeholders, which could also serve as another layer in the overall architecture.

6 Acknowledgment

This research was achieved as part of the National Priority Research Program (NPRP) Research Project: NPRP11S-1227-170135, funded by the Qatar National Research Fund (QNRF): “SupplyLedger project: Designing an Efficient Smart Contract based Blockchain System from Multi-modal Supply Chain Systems”.

References

1. T Mentzer J., Dewitt W., Keebler J. S, Min S., Nix N., Smith C., and Zacharia, Z. "Defining supply chain management," *Journal of Business Logistics*, vol. 22, 09 2001.
2. CoinDesk Homepage, <http://www.coindesk.com/state-of-blockchain-q1-2016>, last accessed 2021/10/15.
3. Bitcoin Homepage, <https://bitcoin.org/bitcoin.pdf>, "Bitcoin: A peer-to-peer electronic cash system," 2008, last accessed 2021/10/15.
4. Mann S., Potdar V., S.Gajavilli R., and Chandan A. 2018. Blockchain Technology for Supply Chain Traceability, Transparency and Data Provenance. The 2018 International Conference on Blockchain Technology and Application (ICBTA 2018). Association for Computing Machinery, New York, NY, USA, 22–26 (2018)
5. Xu X.et al., "The Blockchain as a Software Connector," 13th Working IEEE/IFIP Conference on Software Architecture (WICSA), Venice, 2016, pp. 182-191, (2016).
6. Omohundro S., Cryptocurrencies, smart contracts, and artificial intelligence. *AI Matters Journal* , 1(2):19–21, (2014).
7. Singh A., Parizi R. M., Zhang Q., Choo K. K. R., and Dehghantanha A., Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities. *Computers & Security Journal* , 88, 101654 (2020).
8. Sarno R., W. A. Wibowo, Kartini, Haryadita F., Effendi Y., and Sungkono K., "Determining Model Using Non-Linear Heuristics Miner and Control-Flow Pattern," *TELKOMNIKA Telecommunication Computing Electronics and Control Journal*, vol. 14 (1), 2016.
9. Sarno R., Pamungkas E. W., Sunaryono D., and Sarwosri, "Business process composition based on meta models," *International Seminar on Intelligent Technology and Its Applications (ISITIA)*, (2015).
10. Saylam R. and Sahingoz O. K., "Process mining in business process management: Concepts and challenges" 2013 International Conference on Electronics, Computer and Computation (ICECCO), Ankara, (2013)
11. Berners-Lee T., Hendler J., and Lassila O., "The semantic Web," *Scientific American Journal*, vol. 284, no. 5, pp. 28-37 (2001).
12. Studer R., Benjamins R. and Fensel D., "Knowledge engineering: Principles and methods". *Data & Knowledge Engineering Journal*, 25(1–2):161–198, (1998).
13. Guarino N., Oberle D., and Staab. S. "What is an ontology?" In *Handbook on ontologies*, pages 1–17. Springer, (2009).
14. Mostefai S. and Bouras A., "What ontologies for PLM: A critical analysis," *Proceedings of the IEEE International Technology Management Conference (ICE)*, 2006, pp. 1-8, doi: 10.1109/ICE.2006.7477092 (2006).
15. Muhammad Fahad M., Moalla N., Bouras A., "Towards ensuring Satisfiability of Merged Ontology", *Procedia Computer Science Journal*, 4, P 2216-2225. (2011).
16. Horrocks I., Patel-Schneider P. F., Boley H., Groszof B. and Dean M., "SWRL: A Semantic Web Rule Language Combining OWL and RuleML", *National Research Council of Canada, Network Inference, and Stanford University* (2004).

17. Natschlagler C., "Towards a BPMN 2.0 Ontology", International Workshop on Business Process Modeling Notation, Springer,(2011)
18. Abramowicz W., Filipowska A., Kaczmarek M., and Kaczmarek T., "Semantically enhanced Business Process Modelling Notation. Proceedings of the Workshop on Semantic Business Process and Product Lifecycle Management SBPM (2007).
19. Francescomarino C., Ghidini C., Rospocher M., Serafini L., and Tonella P., "Reasoning on Semantically Annotated Processes". 6th Int.Conf. on Service-Oriented Computing, (2008).
20. Ciccacese P. and Peroni S., "The Collections Ontology: creating and handling collections in OWL 2 DL frameworks". *Semantic Web*, 5(6): 515–529, (2016).
21. Gangemi A., Peroni S., Shotton D., and Vitali F., "A pattern-based ontology for describing publishing workflows", *Proceedings of the 5th International Workshop on Ontology and Semantic Web Patterns*, (2014).
22. Gangemi A., Peroni S., Shotton D., and Vitali F., "The Publishing Workflow Ontology (PWO)", *In Semantic Web*, 8 (5): 703-718. (2015).
23. Roy, Suman & Silvatichi, Gabriel & Holla, V. (2018). *Modeling Industrial Business Processes for Querying and Retrieving Using OWL+SWRL: Confederated International Conferences: CoopIS, C&TC, and ODBASE 2018*, Valletta, Malta, (2018).
24. Sarno R. and Sinaga F. P., "Business process anomaly detection using ontology-based process modelling and Multi-Level Class Association Rule Learning," *International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, 2015, pp. 12-17, (2015).
25. Sinaga, Fernandes, Sungkono, Kelly Rossa, Sarno, Riyanarto, Anomaly detection in business processes using process mining and fuzzy association rule learning - *Journal of Big Data* (2019/2020).
26. Qiushi C., Samet A., Zanni-Merk C., Beuvron F., Reich C. An Ontology-based Approach for Failure Classification in Predictive Maintenance Using Fuzzy C-means and SWRL Rules. *Procedia Computer Science*, Elsevier, 159, pp.630-639(2019).
27. H Gasmı, A Bouras. "Ontology-Based Education/Industry Collaboration System. *IEEE Access*, 1-16. (2017).