DOI: 10.15439/2020F210
ISSN 2300-5963 ACSIS, Vol. 22

Towards Trustworthy Horizontal Integration in Industry 4.0 Based on DLT Networks

Nikola Todorović, Marko Vještica, Vladimir Dimitrieski, Miroslav Zarić, Nenad Todorović, Ivan Luković
Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia
Email: {nikola.todorovic, marko.vjestica, dimitrieski, miroslavzaric, nenadtod, ivan}@uns.ac.rs

***Abstract*—In recent years, Industry 4.0 has promoted the enhanced horizontal integration of value chain participants, aiming to improve the efficiency and effectiveness of Cross-Organizational Business Processes. In this paper, we discuss transparency and data privacy challenges that occur with the introduction of a high level of horizontal integration. Private, permissioned Distributed Ledger Technology systems and smart contracts can be used to address these challenges and enhance the integration of business processes across the entire value chain. To make this possible, we propose a creation of a Model-Driven Software Development approach based on a Domain-Specific Modeling Language that would enable automatic generation of smart contracts. Generated smart contracts could then be used by collaborating parties to supervise the state of production and contract fulfillment in a trustworthy and secure way.**

I. Introduction

With the introduction of smart devices, warehouse systems, and production facilities capable of exchanging information autonomously, manufacturers started establishing global networks in the form of Cyber-Physical Systems (CPS). These systems are leading the way for the fourth industrial revolution [1]. Interconnected autonomous and cooperative entities enable closer collaboration between business parties within a value chain when executing Cross-Organizational Business Processes (CBPs) [2] [3]. Closer collaboration provides new possibilities and mutual benefits for involved parties and allows for improvements in their operational and business performance [4]. For instance, it increases the capability of value chains to manufacture small, customized batches of products cost-effectively [5] [6]. As an example, within the car manufacturing industry, this would mean that customers no longer have to choose from a set of manufacturer-defined option packages for a vehicle model, but would instead be able to mix and match individual components to meet their specific needs [7]. Although production of highly-customized goods is, to some extent, possible with current collaboration levels, it implies significantly higher prices and longer delivery times.

Facilitating CBPs implies integrating different IT systems to enable the interoperability of production systems of the involved parties. In the domain of Industry 4.0, this is referred to as Horizontal Integration – integration of various IT systems used in manufacturing and business planning processes that involve an exchange of materials, energy, and information [8]. Although beneficial, horizontal integration introduces two opposing challenges for the execution of CBPs – the protection of highly sensitive corporate data and a need for an appropriate level of transparency for the correct attribution of legal liability [1]. If not adequately managed, sensitive data exchanged to coordinate production and logistic activities between different companies could be misused by malicious members of the chain. Strict authorization rules must be imposed to protect sensitive data and to regulate whom and under what circumstances may obtain shared data. Contrarily, manufacturing facilities may be subject to a liability action for faults in their performance as part of the value chain. A lack of structural transparency could make it almost impossible to explicitly determine who performed a particular action, resulting in uncertainty regarding legal liability. Correct attribution of liability should be facilitated by the provision of precise documentary evidence concerning the different manufacturing steps and system statuses.

A method and a software solution for the secure, transparent, and trustworthy enactment and integration of CBPs must be utilized to address data protection and transparency challenges and support trust-building among CBP partners. One proposed approach is the use of Distributed Ledger Technology (DLT) platforms and blockchain technologies with smart contracts for the implementation of the horizontal integration, with end-to-end engineering spanning across the entire value chain [9]. DLT is a type of a distributed database, while blockchain represents a distributed data structure that implements DLT, and comprises cryptographically linked blocks that contain immutable records of network transactions [10]. Because data records stored in a block are immutable and contain an immutable hash of data stored in a previous block in a chain, data cannot be counterfeited or forged once recorded into a blockchain. Using a DLT platform would improve structural transparency within the value chain and increase trust between included members because it enables entities to have shared control over the access to and evolution of data. The transactions on the platform are generated and validated using smart contracts, computer programs whose execution is guaranteed by system rules, and for which the outcome of execution is verifiable and auditable by all network participants. Smart contracts have a potential to improve coordination within the value chain by automatically verifying that the production process actions are executed according to the contracted specification.

DLT platforms usually provide low-level, general-purpose programming languages for implementing smart contracts. This is not always suitable in the context of Industry 4.0 because the manual specification of smart contracts would reduce the capability of value chains to synchronize and adapt their production in a timely manner. Moreover, it would mean that process and quality engineers, responsible for the production specification, need to be proficient in these languages. These problems could be mitigated by (i) raising the level of abstraction and providing them with a modeling language that is based on concepts and notations they are familiar with and already use in their domains, and then (ii) relying on automatic generation of smart contracts [11].

Our research aims to establish a methodological approach for horizontal integration that would create conditions for a trustworthy and traceable production. We propose a highly conceptualized architecture based on DLTs and smart contracts that would enable a formal description and execution of collaborative production processes. The architecture needs to be centered around a Domain-Specific Modeling Language (DSML) that would enable modeling interoperability requirements and implementation details [12]. The proposal needs to be facilitated by a software solution in which the Model-Driven Software Development (MDSD) principles and DSMLs are used to (i) specify contracted cross-organizational production processes formally and (ii) automatically generate smart contracts that observe the execution of production and store production records in an immutable distributed ledger. The described architecture would enable a trustworthy and secure analysis of records of events that occurred during the production and would allow parties to derive conclusions and determine if there are any discrepancies between negotiated and executed process steps.

The presented work is structured as follows. After the introduction, in Section II we discuss different challenges and requirements for a DSML regarding the notational aspects and execution significance of CBP models used to facilitate the automatic generation of smart contracts. Section III provides a context of collaboration in the domain of Industry 4.0 for which an MDSD approach will be used. In Section IV, we present an MDSD approach for the automatic generation of smart contracts. In conclusion, the proposal is summarized, and the authors give an outlook on predicted outcomes of the proposed investigation.

II. Research Challenges and Related Work

Collaboration within Industry 4.0 implies the execution of cross-organizational production processes between multiple independent parties that are part of a value chain. The execution includes entities from these parties involved in high-level interactions, directed at joint endeavors with the end-goal to deliver highly customized products to end-users cost-effectively and in a timely manner [13]. These entities produce multiple datasets, owned by the involved parties and maintained within their respective production systems. Production systems need to be integrated to share the common data and provide a mutual understanding of records generated during the execution of CBPs. Thus, the interoperability concerns for those systems, i.e., the capability of systems to exchange data and share information and knowledge, must be addressed [14].

One of the leading frameworks that facilitate interoperability in the domain of Industry 4.0 is the ATHENA Interoperability Framework (AIF) [8] [15]. The primary goal of AIF is to provide a generic solution that enables collaborative modeling and execution of CBPs and to be applicable to many different domains [16]. AIF takes a multidisciplinary approach for facilitating CBPs by merging three research areas that support the development of interoperability of enterprise solutions: (i) enterprise modeling, which is used to define interoperability requirements and supports solution implementation, (ii) architectures and platforms which provide implementation frameworks, and (iii) ontology to identify interoperability semantics in the enterprise. Because AIF uses a generic approach for enabling CBPs, the solution is based on languages and technologies suitable for the application in most scenarios and use cases.

The intended research should try to facilitate interoperability between different production systems by utilizing concepts identified in AIF and offering improvements in the way CBPs are modeled in the domain of Industry 4.0. Modeling of production processes in the domain of Industry 4.0 is essential in order to understand, control, and optimize process operations, and has been an important topic of our previous research [20]. Different notational aspects of CBP models, concerned with their expressiveness and visual representation, and the execution significance that concerns their computability by a machine, should be examined while considering different characteristics of the proposed DLT monitoring platform. This is discussed in Section II-A. Our research should also promote trust-building between parties involved in the value chain by utilizing DLT platforms based on blockchain and smart contracts. Different aspects of DLT platforms that have significance for the enactment and integration of CBPs are addressed in Section II-B.

A. Modeling Cross-Organizational Business Processes

Various research challenges should be taken into consideration during the proposed investigation on modeling CBPs. On one hand side, modeling of CBPs implies the ability of a DSML to describe production process specifications in a sufficiently detailed and understandable way to enable the execution of the process. On the other hand side, these specifications should be displayed to related parties through different process interfaces that facilitate understanding of collaboration within the value chain while preserving confidentiality of private, internal enterprise information. One of the most significant challenges for a DSML will be to devise a way to connect private production processes with openly exposed process interfaces and map different representations of intra-organizational processes at the cross-organizational business process level [3]. Additionally, the modeling language should provide users with an ability to model details needed on the execution level, e.g., showing invoked smart contracts and executed transactions, while separating CBP modeling from specific deployment architectures.

Preserving private data while integrating production processes with collaborating parties is a significant security concern for parties involved in a value chain. Some of the production data, e.g., an actor that performed a particular action or details about work instructions that were followed to perform a specific activity, should be revealed only in case of a legal dispute. Thus, raw production data must be aggregated, anonymized, and shared only with select parties involved in a specific production execution. Three different process types should be investigated and customized for use in the domain of Industry 4.0 to allow secure exposure of private process data to related parties: (i) private processes, that represent internal production processes executed by an organization, (ii) interface processes, used to coordinate internal actions with activities of external partners while concealing private data, and (iii) CBPs, used to describe how parties collaborate within a value chain. These process types and dependencies between them are shown in Fig. 1, created based on the Fig. 1 from [3]. In the figure, different process types are separated using dashed lines, while dependencies between them are depicted using red dotted arrows.

Private processes, displayed as the top layer in Fig. 1, should model production process specifications that are executed inside an organization. Modeling of private production processes is an important research topic within Industry 4.0 [17], but it is still not sufficiently covered with the existing studies [18]. Because of an increase in a production process complexity in Industry 4.0, production process models should help process designers think about production processes at a higher abstraction level and be more focused on modeling production process steps to minimize errors during processes specification. Formal production process models would enhance manufacturing flexibility and allow for more precise and domain-specific simulations and would provide means for better integration of humans in production processes in a way that is prescribed by the Lean manufacturing principles [19]. From a notational aspect of a DSML, private production process models need to be specified by using a language that includes concepts for representing materials, products, services, devices, human workers, communication between them, and all process steps and tasks needed to create a product. A notation should also enable describing how tasks described in a private process are aggregated and anonymized to allow mappings between private processes and interface processes. Aggregation and anonymization specifications would enable a formal transformation of data shared with collaborating business parties during production execution.

Interface processes, depicted as an additional layer beneath private processes in Fig. 1, can be used to provide an abstraction of private processes sufficient to coordinate internal actions with activities of external partners while concealing private data [21] [22]. An interface process is shared only with a contracting party and should specify tasks that should be performed by a manufacturer when executing the contracted production. Interface Process 1 shows that interface processes comprise anonymized and aggregated tasks, e.g., operator roles and machine types are used instead of specific actors. This layer should also specify constraints that should be followed and quality control inspections that need to be performed during the production execution. Interface processes will be used as a basis for the creation of suitable private processes

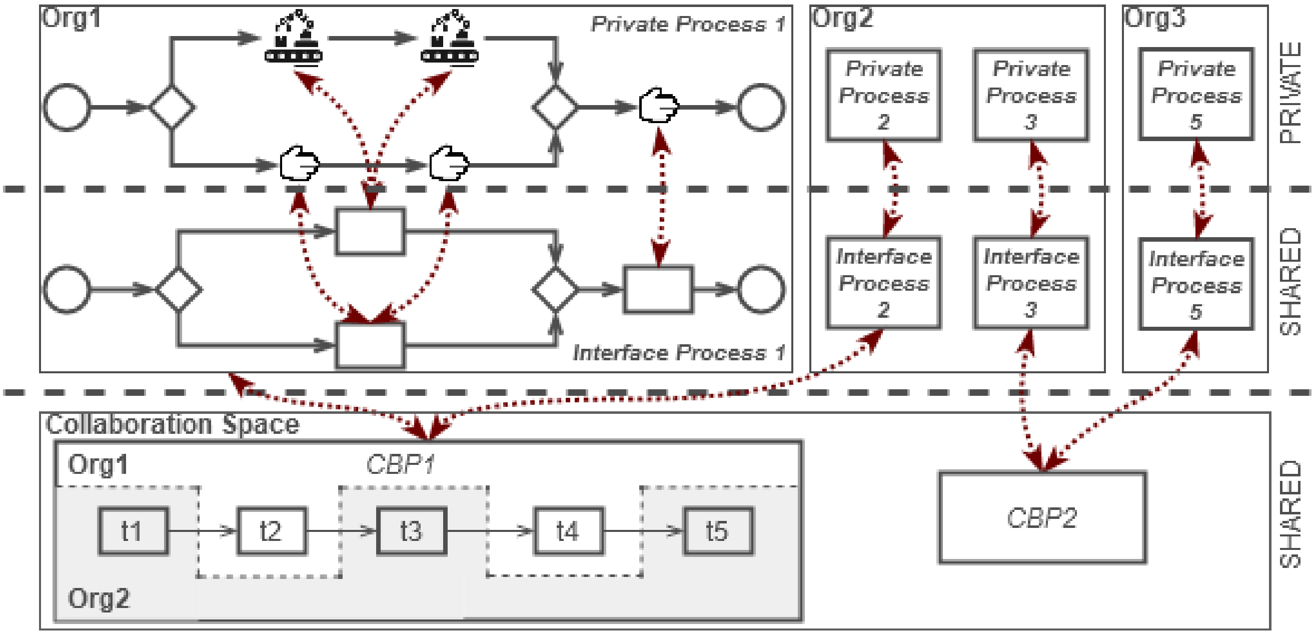

Fig. 1. Dependencies between different process types

and for the generation of smart contracts that will monitor the executed production.

The third layer in Fig. 1 depicts Collaboration Space, where cross-organizational business processes, created by integrating different process interfaces exposed by collaborating parties, can be used to describe how parties collaborate within a value chain. The collaboration should be based on a distributed process model where parties manage their own part of the overall production process [3]. From a high-level viewpoint, a CBP model should specify how the partner processes are interweaved and what tasks each of the parties must perform as agreed in their contract. This is shown with CBP2, created based on Interface Processes 1 and 4, where tasks t2 and t4 are performed by Org1, while Org3 performs t1, t3, and t5. A CBP model should also specify roles of the involved parties, milestones, i.e., critical points used to determine the state of a task, and messages that are transferred during process execution. The specifications of CBPs can be used to create smart contracts for trustworthy monitoring of the enactment of CBPs.

B. Execution Platform

Records of events occurred during the execution of manufacturing processes, provided by machines and operators that take part in the production, are stored and maintained within the production systems of parties involved in the value chain. An execution platform that facilitates horizontal integration should provide mechanisms that guarantee a secure and transparent distribution of records to related parties in order to achieve a common understanding of these events. The architecture recommended by AIF should be expanded to encourage the use of a DLT platform for information sharing and to support trust-building between parties.

The network for Interoperability Development of Enterprise Applications and Software (IDEAS) identified a list of Quality Attributes which highlight technical requirements that should be taken into consideration when developing software that promotes enterprise interoperability [23]. We selected three most important attributes for sharing data during the enactment of CBPs in Industry 4.0: (i) security, which describes the ability of a solution to protect enterprise resources and control access to them; (ii) scalability, that represents the ability of a solution to adjust to an increased number of production tasks; and (iii) performance, the ability of a solution to quickly execute a business task and to retrieve and return information in a timely fashion.

To satisfy these requirements, we propose the use of a private, permissioned, consortium-based DLT platform for storing CBP records. These platforms are administered by a set of identified participants operating under a governance model that enforces a certain degree of trust [10] [24]. Private DLT networks impose restrictions on 'read' access to the ledger, i.e., who can access the network and see transactions. Moreover, permissioned networks allow only a selected set of parties to make changes to the distributed ledger. When it comes to how and to whom the data propagates across the chain, a multi-channel data diffusion model should be used, where transactions and transaction-related data are broadcast to select parties involved in a specific production. Because of this, each node in the network would store only data about transactions with which it is involved. The role of smart contracts, generated based on production process specifications, is to monitor event records and validate that the production execution is conducted according to the contracted specifications. The consensus over the state of the ledger is achieved by relying on the specified endorsement policies designed in a manner that achieves consistent and reliable understanding between participants.

Scalability and performance concerns must also be considered to enable a sufficiently reliable data transfer between integrated parties. Machines used in the production generate a large amount of data that needs to be processed by the blockchain network with low latency. By relying on the identities of participants, a permissioned blockchain can use more traditional Crash Fault-Tolerant (CFT) or Byzantine Fault-Tolerant (BFT) consensus protocols, that are more suitable for scaling the transaction throughput in the network [25].

Several existing solutions consider a use of DLT platforms for the enactment of CBPs [26] [27]. These solutions use smart contract generation to facilitate the collaborative processes integrated using the DLT network. In these solutions, authors present a tool that takes business process specification as an input and generates smart contracts that are then deployed on a public DLT network named Ethereum [28]. Described methods have several limitations regarding a use in Industry 4.0. A use of a public DLT network like Ethereum may not fit the high data security requirements of the Industry 4.0 domain. Instead, enterprise solutions that rely on a private, consortium federated DLT network could be used to protect highly sensitive corporate data. Scalability and performance may also become a concern with the use of a public DLT network, like Bitcoin or Ethereum, where each transaction needs to be processed by every single node in the network. For instance, Ethereum supports up to 15 transactions per second. This creates a severe bottleneck for the execution of production processes in Industry 4.0, where machines involved in the manufacturing generate transactions at a much higher pace. To the best of our knowledge, none of the existing solutions considers high security, performance, and scalability requirements in a unified way. Since these requirements are critical for the application of a solution in Industry 4.0, we propose an investigation of a solution for an architecture based on private, permissioned DLTs and smart contracts that would be suitable for the selected domain.

III. Collaboration Context

A system based on the proposed architecture would reside between collaborating parties with their factories. Such system would receive an order, create a smart contract on which all collaborators agree and forward the appropriate information to a factory for production. The scope of the proposed research is entirely in the cyber world of the Cyber-Physical Systems, and the factory's cyber and physical parts are considered a black box. From the project's standpoint, information systems of the "black box" factory, e.g., Enterprise Resource Planning (ERP) and Manufacturing Execution System (MES), need to be able to receive and store a production process specification together with other production order details such as quantity or desired time of production. In addition to these inputs, factory information systems need to provide a digital output on the work in progress, i.e., information on the currently executed steps of production processes, and data from the factory equipment.

Two steps need to be completed to generate a smart contract automatically. As a first step, a process designer must formally specify a production process specification based on a production order, which includes a product specification. After that, as a second step, a smart contract needs to be generated out of the formal production process specification, containing all necessary actions, i.e., contract clauses that should be performed by the manufacturer in order to produce the end product.

A formal specification of production processes is needed to model production processes with all the details required in every process step. Production process specification is often used as a basis for production coordination between different contractors within the value chain. For example, suppliers within the car manufacturing industry that follow the Advanced Product Quality Planning (APQP) framework are mandated to have a formally stated production process specification approved by the customer [29]. Our goal is to utilize these specifications and supplement them with additional data to gather enough information to generate smart contracts that monitor a described production process and facilitate value chain collaboration.

The formal specification of the production process must be empowered with a model-driven methodology to make it possible and easy to generate smart contracts from production process models, which will assure that all of the agreed production steps are executed.

IV. MDSD Approach to Horizontal Integration

To promote trustworthy collaboration between parties involved in a value chain, we propose a solution in which MDSD principles and DSMLs are used to model CBPs and automatically generate smart contracts. MDSD approach is a part of the Model-Driven paradigm, where models represent a primary artifact at all system development stages and are connected and organized at different abstraction levels. Some of the goals of MDSD are to: (i) increase software system developing speed through automatization and centralized representation of knowledge, (ii) improve software quality through formalization, and (iii) increase reusability of models [30]. The goal of DSMLs in MDSD is to bring modeling concepts closer to users familiar with an application domain so that they can specify their solution with less time compared to General Purpose Modeling Languages (GPMLs) [31]. We believe that an MDSD approach and DSMLs will have an essential role in increasing the capability of value chains to synchronize and adapt their production in a timely manner when executing CBPs.

The solution should be based on three process types described in detail in Section II-A. A high-level overview of the approach is given in Fig. 2. Process designers are responsible for specifying Interface Process Models (IPMs) based on a production process specification contracted with collaborating parties. IPMs represent a high-level technical description of a production process that should include specification of (i) process steps, (ii) actor types or specific resources, e.g., machines, robots, and humans, which should execute process steps, (iii) input and output products, i.e., products like raw materials, components, or finished goods, and (iv) quality constraints, i.e., constraints that refer to quality assurance, and (v) execution constraints, e.g., constraints regarding operator safety in production. Based on an IPM, a smart contract generator (SC Generator) can be used to generate smart contracts. Generated smart contracts will be used to monitor if an organization has performed the production of goods according to the contracted specification.

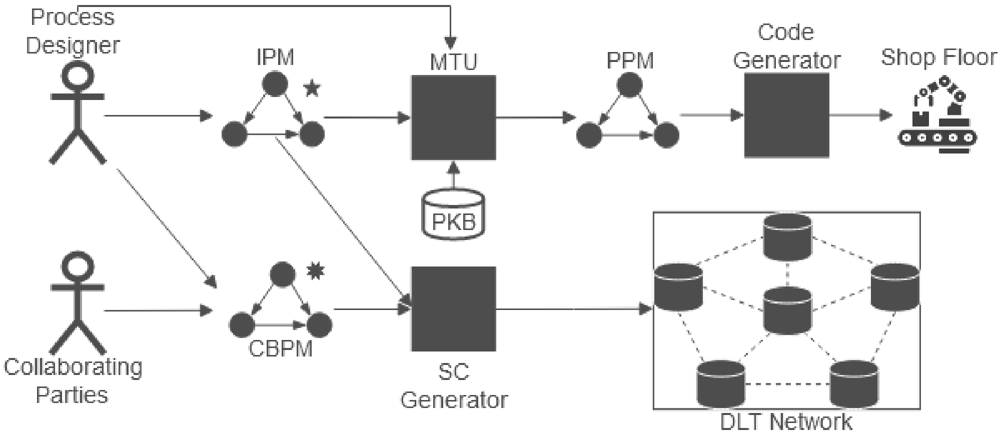

Fig. 2. The proposed MDSD approach

Private Process Models (PPMs) could be created by enriching IPMs and adding data about available resources and the necessary transport activities. A process designer should interact with a Private Knowledge Base (PKB) using a Model Transformation Utility (MTU) to create a PPM that includes the following information that enables process execution: (i) specific resources which should execute process steps, and (ii) logistic information for product and resource movement. Also, a process designer should add details that enable mapping between an IPM and a PPM. These details should define which steps from PPM are aggregated and how, and what represents private data that should not be revealed to related parties. Based on these details, the appropriate event records could be sent to smart contracts during the production execution.

A code generator can be used to generate instructions to resources in the production, and production could then be started. Generated code needs to be human-readable if instructions are sent to mobile devices of human workers or machine-readable if instructions are sent to robots.

A CBP Model (CBPM) is created to coordinate a production between different parties involved in a value chain. For this reason, process designers need to specify (i) which party executes a particular task in the CBP, (ii) what critical points used to determine the state of a task, and (iii) messages that are transferred during process execution. While an IPM can be used to generate smart contracts used to monitor a process executed by a single organization, a CBPM can be used by SC Generator to generate smart contracts that monitor the enactment of CBPs. Monitoring the enactment of CBPs implies observing communication between involved parties and tracking the state of each task.

Once smart contracts are generated, smart contracts should be stored in DLT to which all involved parties have access. The factory information system could then send appropriate signals and information about the fulfillment of specific production criteria, which would automatically trigger actions specified as a part of the stored smart contract. Collaborating parties could then oversee the state of production and contract fulfillment by looking at the immutable store.

V. Conclusion

In this paper, we have proposed an approach for a secure and transparent enactment of CBPs based on DLTs and smart contracts, used for tamper-proof monitoring of the production execution. We have identified and outlined challenges and requirements regarding notational aspects and execution significance for modeling CBPs in Industry 4.0 to facilitate the automatic generation of smart contracts using an MDSD approach. The approach should be based on a DSML that enables the specification of (i) private processes, that represent internal production processes executed by an organization, (ii) view processes, used to coordinate internal actions with activities of external partners while concealing private data, and (iii) CBPs, used to describe how parties collaborate within a value chain. Smart contracts, generated using MDSD principles, should be stored in an immutable distributed ledger and used for monitoring the production performance. Collaborating parties could then supervise the state of production and contract fulfillment by looking at the records of events that occurred during the production. This would promote the understanding of collaboration within the value chain while preserving the confidentiality of private, internal enterprise information.

Predicted outcomes of the proposed research are a system prototype and a new innovative method for trustworthy and automatic monitoring of the enactment of collaboration between parties involved in a value chain. The proposed approach will be tested on a collaboration example from a car manufacturing industry, designed with industry experts and based on openly accessible data. The anticipated value for parties involved in a value chain is a new approach to provide increased safety and transparency during the enactment of collaboration as contracts are automated and tamper-proof. The expected scientific implication is a new methodological approach for horizontal integration that would create conditions for a trustworthy and traceable production.

Acknowledgment

This research paper has been supported by the Ministry of Education, Science and Technological Development through the project no. 451-03-68/2020-14/200156: “Innovative scientific and artistic research from the FTS domain."

References

[1] Kagermann, H., Helbig, J., Hellinger, A., & Wahlster, W. (2013). Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group. Forschungsunion.

[2] Monostori, L. (2014). Cyber-physical production systems: Roots, expectations and R&D challenges. Procedia Cirp, 17, 9-13, https://doi.org/10.1016/j.procir.2014.03.115

[3] Lippe, S., Greiner, U., & Barros, A. (2005). A survey on state of the art to facilitate modelling of cross-organisational business processes. XML4BPM, 1, 7-22.

[4] Flynn, B. B., Huo, B., & Zhao, X. (2010). The impact of supply chain integration on performance: A contingency and configuration approach. Journal of operations management, 28(1), 58-71, https://doi.org/10.1016/j.jom.2009.06.001

[5] Shafiq, S. I., Sanin, C., Szczerbicki, E., & Toro, C. (2016). Virtual engineering factory: Creating experience base for industry 4.0. Cybernetics and Systems, 47(1-2), 32-47, https://doi.org/10.1080/01969722.2016.1128762

[6] Critical Manufacturing, and IYNO. (2017) “The New MES: Backbone of Industry 4.0.”

[7] Schmitt, G. R. A., Bettinger, C., & Rock, G. (2018). Glencoe–A Tool for Specification, Visualization and Formal Analysis of Product Lines. Transdisciplinary Engineering Methods for Social Innovation of Industry, 4, 665-673.

[8] Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. Journal of industrial information integration, 6, 1-10, https://doi.org/10.1016/j.jii.2017.04.005

[9] Fernández-Caramés, T. M., & Fraga-Lamas, P. (2019). A review on the application of blockchain to the next generation of cybersecure industry 4.0 smart factories. IEEE Access, 7, 45201-45218, https://doi.org/10.1109/ACCESS.2019.2908780

[10] Hileman, G., & Rauchs, M. (2017). 2017 global blockchain benchmarking study. Available at SSRN 3040224.

[11] France, R., & Rumpe, B. (2007, May). Model-driven development of complex software: A research roadmap. In Future of Software Engineering (FOSE’07) (pp. 37-54). IEEE, https://doi.org/10.1109/FOSE.2007.14

[12] Vještica, M., Dimitrieski, V., Pisarić, M., Kordić, S., Ristić, S., & Luković, I. (2019, Nov) Towards a formal description and automatic execution of production processes. In Proceedings of 2019 IEEE 15th International Scientific Conference on Informatics, Poprad, Slovakia (pp. 463–468), https://doi.org/10.1109/Informatics47936.2019.9119314

[13] Ferreira, F., Faria, J., Azevedo, A., & Marques, A. L. (2016, October). Industry 4.0 as enabler for effective manufacturing virtual enterprises. In Working Conference on Virtual Enterprises (pp. 274-285). Springer, Cham, https://doi.org/10.1007/978-3-319-45390-3_24

[14] Chen, D., Doumeingts, G., & Vernadat, F. (2008). Architectures for enterprise integration and interoperability: Past, present and future. Computers in industry, 59(7), 647-659, https://doi.org/10.1016/j.compind.2007.12.016

[15] Berre, A. J., Elvesæter, B., Figay, N., Guglielmina, C., Johnsen, S. G., Karlsen, D., ... & Lippe, S. (2007). The ATHENA interoperability framework. In Enterprise interoperability II (pp. 569-580). Springer, London, , doi: 10.1007/978-1-84628-858-6_62

[16] Ruggaber, R. (2006). Athena-Advanced technologies for Interoperability of heterogeneous enterprise networks and their applications. Interoperability of enterprise software and applications, 1, 459-60, https://doi.org/10.1007/1-84628-152-0_45

[17] Da Xu, L. (2011). Enterprise systems: state-of-the-art and future trends. IEEE Transactions on Industrial Informatics, 7(4), 630-640, doi: 10.1109/TII.2011.2167156

[18] Petrasch, R., & Hentschke, R. (2016, July). Process modeling for Industry 4.0 applications: Towards an Industry 4.0 process modeling language and method. In 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE) (pp. 1-5). IEEE, https://doi.org/10.1109/JCSSE.2016.7748885

[19] Shah, R., & Ward, P. T. (2003). Lean manufacturing: context, practice bundles, and performance. Journal of operations management, 21(2), 129-149, https://doi.org/10.1016/S0272-6963(02)00108-0

[20] Vještica, M., Dimitrieski, V., Pisarić, M., Kordić, S., Ristić, S., & Luković, I. (2020) An Application of a DSML in Industry 4.0 Production Processes. In IFIP Advances in Information and Communication Technology, Novi Sad, Serbia, pp. 441-448, https://doi.org/10.1007/978-3-030-57993-7_50

[21] Schulz, K. A., & Orlowska, M. E. (2004). Facilitating cross-organisational workflows with a workflow view approach. Data & Knowledge Engineering, 51(1), 109-147, https://doi.org/10.1016/j.datak.2004.03.008

[22] Liu, D. R., & Shen, M. (2001, April). Modeling workflows with a process-view approach. In Proceedings Seventh International Conference on Database Systems for Advanced Applications. DASFAA 2001 (pp. 260-267). IEEE.

[23] Chen, D., & Doumeingts, G. (2003). European initiatives to develop interoperability of enterprise applications—basic concepts, framework and roadmap. Annual reviews in control, 27(2), 153-162, https://doi.org/10.1016/j.arcontrol.2003.09.001

[24] Wüst, K., & Gervais, A. (2018, June). Do you need a blockchain?. In 2018 Crypto Valley Conference on Blockchain Technology (CVCBT) (pp. 45-54). IEEE.

[25] IBM. (2020) Hyperledger Fabric Documentation. https://hyperledgerfabric.readthedocs.io/en/release-1.4/ Accessed April 14, 2020.

[26] Weber, I., Xu, X., Riveret, R., Governatori, G., Ponomarev, A., & Mendling, J. (2016, September). Untrusted business process monitoring and execution using blockchain. In International Conference on Business Process Management (pp. 329-347). Springer, Cham.

[27] Klinger, P., & Bodendorf, F. (2020). Blockchain-based Cross-Organizational Execution Framework for Dynamic Integration of Process Collaborations. In 15th International Business Informatics Congress, https://doi.org/10.30844/wi_2020_i2-klinger

[28] GitHub. Ethereum White Paper: A Next-Generation Smart Contract and Decentralized Application Platform. https://github.com/ethereum/wiki. Accessed January 25, 2020.

[29] Stamatis, D. H. (2018) Advanced Product Quality Planning: The Road to Success.

[30] Dimitrieski, V. (2017) Model-Driven Technical Space Integration Based on a Mapping Approach. Ph.D. Thesis, University of Novi Sad, Faculty of Technical Sciences, Serbia.

[31] Mernik, M., Heering, J., & Sloane, A. M. (2005) When and how to develop domain-specific languages. ACM Comput. Surv., vol. 37, no. 4, pp. 316–344, doi: 10.1145/1118890.1118892.