# Towards a Process-oriented Analysis of Blockchain Data

Claudio Di Ciccio[1]

**Abstract:** Blockchains sequentially store the history of transactional information, in a virtually immutable and distributed way. Moreover, second-generation blockchains such as Ethereum are programmable environments, and every operation invocation towards the smart contracts corresponds to a transaction sequentially collated in the ledgers. They thus allow for the controlled enactment of multi-party processes as well as the immutable recording of their distributed execution. Despite the verification, tracking, and monitoring of such blockchain-enabled processes appears paramount, a formal and implemented framework encompassing those aspects is still a mostly unexplored research avenue. The talk revolves around the current state of the art, as well as the opportunities and challenges that arise when it comes to conducting a process-oriented analysis on data stemming from blockchains, from a representation and modelling perspective.

**Keywords:** Blockchain; Distributed ledger; Process mining; Logging logic

Blockchain-based collaborations lay the backbone of processes involving multiple participants that interact between them [Me18, Hä18]. Recently, techniques have been devised that allow for the direct translation of business process models into Smart Contracts [Di19]. Blockchains trace the sequence of tasks carried out in the course of process executions by the totally ordered recording (upon consensus) of transactions between involved parties. The payload of transactions can provide further information on the tasks carried out. Second-generation blockchain technologies such as Ethereum allow Smart Contracts to emit events that can be captured by Distributed Applications (ĐApps). Event logs and data parameters of the transactions can reveal notifications and execution context. They can, thus, enable process analytics on the blockchain [vdA16, Me18]. The persistence and immutability of those data cater for auditing endeavours on the enacted processes [JH19].

Nevertheless, understanding the behaviour and performance of blockchain-enabled processes still requires noticeable manual labour. The way in which logs and exchanged data are engineered is tightly bound to how the the Smart Contracts are encoded. As shown by [Di18] the interpretation of the information stored in the blockchain is far from trivial. We can, for instance, observe that at block `1196772` of the Ethereum public blockchain, transaction `0x656252f3`... reports on a call of function `0xefe73dcb` on contract `0x0e6e0313`... from account `0x1387e749`.... By reverse-engineering the Application Binary Interface (ABI) of the invoked Smart Contract, one can extract the function signature (specifically, `Customer_Has_a_Problem()`) and speculate that the function name is the activity label [Mü19]. However, information pertaining to process semantics such as the

[1] Sapienza University of Rome, Department of Computer Science, Rome, Italy. diciccio@di.uniroma1.it

running process instance to which that transaction belongs, the conditions that led to that invocation, or the role of the sender, remain obscure. This hampers the ex-post interpretation of the sources of information, let alone their automated analysis. The promised verification and traceability of executed processes ends up being ad-hoc and demanding manual effort, not so differently from what used to happen when striving to understand the behaviour of legacy systems through their logs [OGX12].

This issue calls for the introduction of a specification language that decouples the business logic (encoded, e.g., in Smart Contracts) from the *logging logic*. Preliminary ideas, exposed in [Kl19, Mü19], show interesting results to generate XES[2] event logs for process mining from transactions stored on the blockchain through metadata descriptors. We argue, though, that a semantically rich language for logging logic is needed, so that actions carried out via blockchain operations are connected to the stored data in a semantically expressive way. A promising basis to this end is given by the recent Object-Centric Behavioural Constraints (OCBC) specification language for processes [Ar19]. However, the logging language should not dictate how the process behaves, but define the conditions under which logging information is stored, and how.

New opportunities and unaddressed challenges open up in this context, including the following: from a formal perspective, the problems of satisfiability of logging specifications and of their consistency with the original process; from a design perspective, the adoption of aspect-oriented programming approaches to decouple business logic code from the logging logic descriptors, and the mechanisms to grant access to (parts of) the stored information; from an implementation perspective, the trade-off between richness, abstraction and retrievability of data on one side, and the execution and storage costs on the other side.

**Acknowledgements.** The author thanks Luciano García-Bañuelos, Jan Mendling, Marco Montali, Wil van der Aalst, and Ingo Weber for the fruitful discussions and helpful insight. The author is also grateful to Stefan Bachhofner, Dominik Felix, Dominik Haas, and Roman Mühlberger for their investigations and active collaboration.

## References

[Ar19] Artale, A.; Calvanese, D.; Montali, M.; van der Aalst, W.M.P.: Enriching Data Models with Behavioral Constraints. In: Ontology Makes Sense. volume 316, pp. 257–277, 2019.

[Di18] Di Ciccio, C.; Cecconi, A.; Mendling, J. et al.: Blockchain-Based Traceability of Inter-organisational Business Processes. In: BMSD. volume 319, pp. 56–68, 2018.

[Di19] Di Ciccio, C.; Cecconi, A.; Dumas, M. et al.: Blockchain Support for Collaborative Business Processes. Informatik Spektrum, 42:182–190, 2019.

[Hä18] Härer, Felix: Decentralized Business Process Modeling and Instance Tracking Secured by a Blockchain. In: ECIS. p. 55, 2018.

[2] `http://xes-standard.org/`. Accessed 20/01/2020

[JH19] Jans, M.; Hosseinpour, M.: How active learning and process mining can act as Continuous Auditing catalyst. Int. J. Accounting Inf. Systems, 32:44–58, 2019.

[Kl19] Klinkmüller, C.; Ponomarev, A.; Tran, A.B. et al.: Mining Blockchain Processes: Extracting Process Mining Data from Blockchain Applications. In: Business Process Management: Blockchain and Central and Eastern Europe Forum. pp. 71–86, 2019.

[Me18] Mendling, J.; Weber, I.; van der Aalst, W.M.P. et al.: Blockchains for Business Process Management - Challenges and Opportunities. ACM TMIS, 9(1):4:1–4:16, 2018.

[Mü19] Mühlberger, R.; Bachhofner, S.; Di Ciccio, C. et al.: Extracting Event Logs for Process Mining from Data Stored on the Blockchain. In: BPM Workshops. pp. 690–703, 2019.

[OGX12] Oliner, A.J.; Ganapathi, A.; Xu, W.: Advances and challenges in log analysis. Commun. ACM, 55(2):55–61, 2012.

[vdA16] van der Aalst, W.M.P.: Process Mining - Data Science in Action. Springer, 2016.