

Next-Generation Trusted Process Enactment using Blockchain State Channels

Fabian Stiehle

Technical University of Munich, School of Computation, Information and Technology, Germany

Abstract

Blockchain-based enactment can guarantee the enforcement of process rules and the preservation of process data. This is achieved without introducing a centrally controlled component—eliminating certain trust concerns in interorganisational processes. However, many issues regarding scalability related metrics like throughput, latency, and cost, but also tradeoffs between transparency and confidentiality remain challenging. To address these issues, we aim to investigate state channel technologies for blockchain-based enactment. State channels have been proposed to scale blockchains. They allow to conduct most transactions off-chain, while mostly retaining the core security properties offered by blockchain. We aim to explore the design space of such channel constructions for process enactment and investigate their relation to other scaling solutions. In doing this, we aim to mitigate some of the current solutions' unfavourable qualities and provide a better understanding of available design options.

Keywords

Blockchain, Process Enactment, Interorganisational Processes, State Channels


1. Motivation


Blockchain-based enactment can guarantee the enforcement of process rules and the preservation of process traces in a tamper-proof manner—without the introduction of a centralised component. Through this, blockchain can reduce trust assumptions among untrusting parties [1]. Work, so far, has focused on enacting processes on the blockchain, so-called on-chain solutions. Often, by translating a process model into a smart contract [2]. Smart contracts are programs executed by the distributed nodes in the blockchain network; these nodes together reach a consensus on the execution and execution history of the contracts. On-chain solutions, however, come with downsides regarding cost, throughput, and latency. Previous work has, thus, focused on optimising the on-chain components (e.g., [3, 4, 5]).

In this work¹, we aim to take a different direction. We aim to investigate layer two *state channels* for blockchain-based process enactment. In a recent survey, we pointed to layer two technologies as a promising research direction to address scalability related challenges [2]. Layer two technologies attempt to scale the the on-chain environment (layer one) by reducing its load: they aim at performing computation off-chain. Security is guaranteed by collateral or delayed finality [7]. In state channels, transactions are exchanged off-chain and the blockchain

BPM 2023 Doctoral Consortium

 fabian.stiehle@tum.de (F. Stiehle)

 © 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

¹Part of this proposal describes work that has already been accepted for publication in [6]. Thus, some descriptions, specifically on research question one, are taken from there.

is only used as a settlement layer, reducing the on-chain footprint, while mostly retaining the core security properties offered by blockchain [7].

This idea initially developed from the concept of payment channels (see e.g., [8]). Say, you want to pay an online news site \$0.10 per article that you read. You create a channel with the news site, where you lock \$5.00 as initial funds (or *collateral*). Every time you read an article, you exchange an off-chain transaction with the news site, assigning an additional \$0.10 to their account. After 32 articles, you decide to close the channel, with the accrued \$3.20 assigned to the news site and the remaining \$1.80 refunded to your account. This concept can be generalised to state channels [9]. In state channels, participants wishing to transact first agree on a contract governing the rules of the channel and encode these on the blockchain (e.g., in a smart contract). They then conduct off-chain transactions. For each transaction, they agree on the outcome and cryptographically commit to their agreement. Finally, when they have concluded their contract, they submit the final state to the chain. If at any point a participant (supposedly) violates the rules, e.g., attempts to falsify the outcome of a transaction, or become unavailable, the last unanimously agreed transaction is posted to the blockchain. From this state, participants can then safely resume their interaction on the chain, where the blockchain protocol enforces the honest execution of the contract.

While state channels promise to improve scalability related metrics, they come with their own set of challenges. Applying the concept to concrete use cases is challenging. For example, state channel designs require participants to take turns in the protocol to prevent deadlocks. This can become complex in an n -party state channel design [10]. Additionally, state channels introduce new components and significantly increase complexity. Different state channel design have been proposed, each with their own set of advantages and drawbacks [7, 10]. Their implications to concrete use cases is not fully understood yet.

We argue that state channels are a fit for inter-organisational processes, where the focus is on a set of interconnected autonomous participants, initiating messages according to some schedule captured in a process model. Furthermore, using model-driven engineering techniques we aim to alleviate the additionally introduced complexity by generating well-tested artefacts from process models. We explore this notion in our first research question: *RQ1: How can state channel technology be utilised to achieve favorable tradeoffs for blockchain-based process enactment?*

The current state of the art in state channel designs focuses on the formalisation and security of protocols for general applications (e.g., [9, 11, 12]); it remains mostly unclear how different state channel designs compare in a quantifiable manner. Furthermore, how they compare to existing scaling solutions. It is unclear under what circumstances state channel designs will add benefits to a certain use case and when they will not. Thus, in research question two we ask: *RQ2: How do different state channel designs compare to other scaling approaches?*

In the seminal paper of Weber et al. [1], the use of blockchain in inter-organisational processes was motivated by enabling the resolution of contractual disputes between process participants. However, while the immutable process trace theoretically enables the identification of contractual breaches, no approach details a dispute resolution process, nor is its facilitation supported. It is unclear in which state the process remains once such a dispute is raised. This is also noted in our previous survey [2]. State channels, on the other hand, introduce a dispute phase, where disagreements and faults are resolved in order to continue the process. We expect that there is

Research Question	Methodology	Progress
RQ1 How can state channel technology be utilised to achieve favorable tradeoffs for blockchain-based process enactment?	Artefact Design & Evaluation	First Results
RQ2 How do different state channel designs compare to other scaling approaches?	Benchmark Studies	In Preparation
RQ3 How can blockchain facilitate the resolution of process enactment disputes?	Artefact Design & Evaluation; Case Study	Pending

Table 1

The project’s research questions, the corresponding scientific method, and the current progress.

potential to design dispute phases, specific to business cases, to incentivise honest participation. Thus, in research question three we ask: *RQ3: How can blockchain facilitate the resolution of process enactment disputes? what benefit can state channels add?*

Our early results show the value of applying state channel technology to process enactment [6]. In answering our research questions, we envision a holistic approach, improving on the essential tradeoffs of blockchain-based enactment.

2. Related Work

The current state of the art in blockchain-based business process enactment focuses on on-chain enactment [2]. Following the seminal work of Weber et al. [1], the research goal has been to improve the capabilities (eg., [13, 14, 15, 16]), support other modelling methodologies (eg., [17, 18]), or demonstrate improved cost of the on-chain components (eg., [5, 19, 4]). In a recent survey, we report that scalability in particular remains an open challenge [2]. To address this, related work has focused on improving the cost of the on-chain components. García-Bañuelos et al. [3] introduced an optimised generation of smart contracts through petri net reduction. López-Pintado et al. [5] and Loukil et al. [4] propose an interpreted approach, where a process model is not compiled but executed by an interpreter component on the blockchain. While the deployment becomes more costly, it leads to cost savings over multiple instance runs.

The state of the art in state channel² constructions focuses on the formalisation and security of protocols for general applications (e.g., [9, 11, 12]). More in line with our work, McCorry et al. [21] present a case study investigating the feasibility and applicability of their state channel construction. They present a template for migrating an existing smart contract to a state channel construction. While they present a state channel architecture for n parties, they chose a two-party game as their case. In two-party games, participants can take turns. However, when more participants are involved, a schedule becomes necessary [12]. In our work, we focus on a model-driven approach, where a process model is used to automatically generate the entire channel setup. We aim to use the process control-flow extracted from a

²Hyperledger Fabric uses the terminology of channels for their subnet functionality [20]. The similarity to state channels is weak; like subnets, fabric channels partition the on-chain ledger. State channels construct off-chain channels and use the security guarantees of the on-chain ledger as settlement and dispute resolution layer.

process model to naturally enforce a schedule upon the participants. We want to investigate the particularities of enforcing interorganisational processes in channels and evaluate our approach quantitatively and qualitatively in comparison to existing enactment and scaling approaches as well as investigating the design of a process-aware dispute resolution process.

3. Approach

In Table 1 we show the project's research questions, the corresponding scientific method to approach the question, and the current progress.

3.1. RQ1 How can state channel technology be utilised to achieve favorable tradeoffs for blockchain-based process enactment?

In question one, we follow a design science approach [22] to determine and evaluate different design options for constructing state channels capable of enacting processes. These artefacts will be thoroughly evaluated, both quantitatively (regarding latency, throughput, cost) and qualitatively (regarding blockchain specific security guarantees, see e.g., [23, Chapter 1.4]). Our first results show a significant reduction of the on-chain footprint for common settings when enacting processes in state channels [6]. Thus, reducing cost and increasing throughput. Additionally, we find that blockchain specific properties (immutability, non-repudiation, integrity, transparency, and equal rights) largely remain intact—as long as some additional assumptions are met, such as having at least one honest participant per channel.

We believe future designs can build on these results and reduce the on-chain footprint further. For example, by exploring a channel network design, where multiple channels are supported by one contract. Furthermore, we aim to assess more complex process cases where latency is a limiting factor and multiple instances must be scheduled in parallel. Beyond the novel application of state channels to the BPM domain, we expect this RQ will contribute new knowledge regarding the design and application of state channel technology in general.

3.2. RQ2 How do different state channel designs compare to other scaling approaches?

To answer question two, we aim to perform comprehensive benchmark studies [24]. Our initial results show that state channel-based enactment can considerably reduce transaction cost and increase throughput [6]. But different state channel design have been proposed, extending on the original concept, each with their own set of advantages and drawbacks [7, 10].

The question remains, how these different designs compare in a qualitative and quantifiable manner and under what circumstances they add benefits to a use case. Furthermore, it remains unclear whether state channels can add favourable qualities to other scaling solutions as well. The enactment of processes, so far, has focused on on-chain enactment, and recommending the use of permissioned blockchains like Hyperledger Fabric to save cost [2]. We aim to assess to what extend state channel-based execution can improve scalability related metrics of existing scaling solutions, such as permissioned blockchains. We expect that state channel-based execution can improve throughput, reduce cost, but more importantly, considerably

reduce latency. Based on our in RQ1 designed methodology and artefacts, we aim to perform comprehensive benchmarks assessing different designs and cases. We aim to explore, whether adding state channels to existing scaling solutions can add real benefit given the assumptions and weakened guarantees of state channel-based execution.

3.3. RQ3 How can blockchain facilitate the resolution of process enactment disputes?

While traditional on-chain enactment enforces the control flow on the chain directly, state channels use a dispute protocol to ensure an honest participant can always enforce correct behaviour on the chain, should the channel deviate from the conforming flow. In question three we want to investigate the repercussions of this, as well as investigate different dispute resolution protocol designs.

To answer this question, we aim to design dispute phase protocols and evaluate them under certain use case conditions. A more advanced dispute design could, for example, penalise non-conforming participants and remove them from the process by starting a protocol which goal it is to re-bind the role of the participant. We believe there is potential to design advanced dispute processes, specific to process enactment, even improving upon current on-chain enactment approaches. Furthermore, there are potential disputes arising from real world issues, which are not depicted in control-flow. A dispute process could accommodate for such disputes by integrating different litigation phases, as, for example, proposed in [25].

4. Outlook

We propose to investigate state channels for the use in business process enactment. This promises to alleviate some of the inherent scalability issues of blockchain-based execution. Indeed, our initial results show that state channel-based enactment can considerably reduce transaction cost and increase throughput [6]. While our aim is to investigate the foundational aspects of this approach with a focus on scalability first, state channels also reduce the exposure of process data. While confidentiality is breached during a dispute phase, there is potential to design the dispute phase in a confidentiality-preserving manner; for example, by utilising zero-knowledge proofs. While these are usually costly operations, they would only be required in the case of a dispute, making their use more viable [26].

References

- [1] I. Weber, X. Xu, R. Riveret, G. Governatori, A. Ponomarev, J. Mendling, Untrusted Business Process Monitoring and Execution Using Blockchain, in: *BPM*, volume 9850 of *LNCS*, Springer, Cham, 2016, pp. 329–347.
- [2] F. Stiehle, I. Weber, Blockchain for business process enactment: a taxonomy and systematic literature review, in: *BPM: Forum*, volume 459 of *LNBIP*, 2022, pp. 5–20.
- [3] L. García-Bañuelos, A. Ponomarev, M. Dumas, I. Weber, Optimized Execution of Business Processes on Blockchain, in: *BPM*, Springer, Cham, 2017, pp. 130–146.

- [4] F. Loukil, K. Boukadi, M. Abed, C. Ghedira-Guegan, Decentralized collaborative business process execution using blockchain, *WWW* 24 (2021) 1645–1663.
- [5] O. López-Pintado, M. Dumas, L. García-Bañuelos, I. Weber, Interpreted execution of business process models on blockchain, in: *EDOC*, IEEE, 2019, pp. 206–215.
- [6] F. Stiehle, I. Weber, Process channels: A new layer for process enactment based on blockchain state channels, in: *BPM'23: International Conference on Business Process Management*, Utrecht, Netherlands, 2023. In press, accepted.
- [7] L. Gudgeon, P. Moreno-Sanchez, S. Roos, P. McCorry, A. Gervais, Sok: Layer-two blockchain protocols, in: *Financial Cryptography and Data Security*, 2020, pp. 201–226.
- [8] J. Poon, T. Dryja, The bitcoin lightning network: Scalable off-chain instant payments, <https://lightning.network/lightning-network-paper.pdf>, 2016. Online, Accessed 2023-03-29.
- [9] A. Miller, I. Bentov, S. Bakshi, R. Kumaresan, P. McCorry, Sprites and state channels: Payment networks that go faster than lightning, in: *Financial Cryptography and Data Security*, 2019, pp. 508–526.
- [10] L. D. Negka, G. P. Spathoulas, Blockchain state channels: A state of the art, *IEEE Access* 9 (2021) 160277–160298.
- [11] S. Dziembowski, S. Faust, K. Hostáková, General state channel networks, in: *ACM SIGSAC CCS*, 2018, pp. 949–966.
- [12] S. Dziembowski, L. Eckey, S. Faust, J. Hesse, K. Hostáková, Multi-party virtual state channels, in: *EUROCRYPT*, Springer, Cham, 2019, pp. 625–656.
- [13] O. López-Pintado, L. García-Bañuelos, M. Dumas, I. Weber, A. Ponomarev, Caterpillar: A business process execution engine on the Ethereum blockchain, *Software: Practice and Experience* (2019) spe.2702.
- [14] Q. Lu, A. Binh Tran, I. Weber, H. O'Connor, P. Rimba, X. Xu, M. Staples, L. Zhu, R. Jeffery, Integrated model-driven engineering of blockchain applications for business processes and asset management, *Software: Practice and Experience* 51 (2021) 1059–1079.
- [15] O. López-Pintado, M. Dumas, L. García-Bañuelos, I. Weber, Controlled flexibility in blockchain-based collaborative business processes, *Information Systems* 104 (2022) 101622.
- [16] A. Abid, S. Cheikhrouhou, M. Jmaiel, Modelling and Executing Time-Aware Processes in Trustless Blockchain Environment, in: *Risks and Security of Internet and Systems*, volume 12026 of *LNCS*, 2020, pp. 325–341.
- [17] G. Meroni, P. Plebani, F. Vona, Trusted Artifact-Driven Process Monitoring of Multi-party Business Processes with Blockchain, in: *BPM: Forum*, volume 361, 2019, pp. 55–70.
- [18] M. F. Madsen, M. Gaub, T. Høgnason, M. E. Kirkbro, T. Slaats, S. Debois, Collaboration among adversaries: Distributed workflow execution on a blockchain, in: *Symposium on Foundations and Applications of Blockchain*, 2018.
- [19] C. Sturm, J. Szalanczi, S. Schönig, S. Jablonski, A lean architecture for blockchain based decentralized process execution, in: *BPM Workshops*, 2018.
- [20] E. Androulaki, A. Barger, Bortnikov, et al., Hyperledger fabric: a distributed operating system for permissioned blockchains, in: *EuroSys*, 2018, pp. 1–15.
- [21] P. McCorry, C. Buckland, S. Bakshi, K. Wüst, A. Miller, You sank my battleship! a case study to evaluate state channels as a scaling solution for cryptocurrencies, in: *Financial Cryptography and Data Security*, Springer, 2019, pp. 35–49.

- [22] A. Hevner, S. Chatterjee, A. Hevner, S. Chatterjee, Design science research in information systems, *Design research in information systems: theory and practice* (2010) 9–22.
- [23] X. Xu, I. Weber, M. Staples, *Architecture for Blockchain Applications*, Springer, 2019.
- [24] D. Bermbach, E. Wittern, S. Tai, *Cloud service benchmarking*, Springer, 2017.
- [25] S. Migliorini, M. Gambini, C. Combi, M. La Rosa, The rise of enforceable business processes from the hashes of blockchain-based smart contracts, in: *BPMDS*, 2019, pp. 130–138.
- [26] Y. Zhang, Y. Long, Z. Liu, Z. Liu, D. Gu, Z-channel: Scalable and efficient scheme in zerocash, *Computers & Security* 86 (2019) 112–131.