Staking Assets Management on Blockchains: Vision and Roadmap

Stefan Driessen[1]

Jheronimus Academy of Data Science, Sint Janssingel 92, 5211 DA, 's-Hertogenbosch, The Netherlands University s.w.driessen@jads.nl
Supervisors: Willem-Jan v.d. Heuvel - w.j.a.m.v.d.heuvel@jads.nl
Damian Tamburri - d.a.tamburri@tue.nl

Abstract. This paper introduces and explores the vision wherefore stakeholders and the process of staking —that is, the idea of guaranteeing the quality of a process by risking valuable assets on their correct execution— may run both on and off a blockchain while in the context of cloud-enabled services and processes. The emerging trend behind blockchain-oriented computing and the reliance on stakeholders therein make distilling and evaluating this vision a priority to deliver high-quality, sustainable services of the future. We identify key defining concepts of stakeholders and the staking process, using three very different staking scenarios as a base. Subsequently, we analyze the key challenges that these stakeholders face and propose the development of a framework that can help overcome these challenges. Finally, we give a road-map to steer systematic research stemming from the proposed vision, leveraging design science along with short-cyclic experimentation.

Keywords: Blockchain · Staking · Service Monitoring · Cloud

1 Introduction

The continued rise in popularity of blockchain technology has sparked new solutions for a wide variety of service processes, such as smart contracts, that place a strong emphasis on trusted computing and transparency [1].

The trust or *trustlessness* in most of these solutions is realized by the provision of proofs, which are publicly shared on the blockchain. These proofs manifest themselves in many different forms and shapes, such as Proof-of-Work (PoW), Proof-of-Stake (PoS) or Proof-of-Authority (PoA) and it is a hot topic of debate to what extent these proofs actually realize a trusted or trustless environment, see, e.g. [5]. An emerging key actor in blockchain services is the *staker*; an actor who proves that they are invested in the quality/correctness of the service, its environment and its execution in an attempt to add trustworthiness to it. Usually, this is done by *locking* an asset (such as a cryptocurrency) on the blockchain, which will either lose value or fail to generate revenue if the service fails to live up to the promised/expected quality.

Stakers often come into play when services run partly on and partly off the blockchain; in a common scenario, a transaction may be comprised of *agreements* (including, for example, classically-defined business-level agreements and application-level agreements) that may be stored on the blockchain. The actual execution of the process, however, (e.g., delivery of a digital asset), will take place off-chain. The staker pertains to that actor in such scenarios that improves trustworthiness in the sense that it checks, verifies, or otherwise *witnesses* that, what has been promised initially, is (likely to be) actually delivered. For example, the staker can help to maintain a sufficient level of network security by replicating network assets. Roles that such a staker may play include, but are not restricted to, logging, monitoring, metering, provisioning, and compliance assurance.

Our vision in this sense is, therefore, to investigate those scenarios in which stakers attempt to add trustworthiness to a blockchain environment and to support the staker's endeavor to do so. This proposition has, so far, been neglected at best and deserves further attention.

This paper sets out to define in an abstract manner the concept of a *staker* in a staking scenario; we conclude the definition by defining the main challenges behind such a scenario. Subsequently, the paper proposes the development of a novel staking methodology and associated toolkit to support stakers in decision-making scenarios where actual staking can take place in a controllable and repeatable manner. The paper furthermore explores this approach against three prototypical blockchain-oriented orchestration service scenarios and plots a roadmap for future work.

2 Background

This section discusses 3 prototypical application staking scenarios and generalizes from them into several fundamental key characteristics to underpin a generic definition of a staker as well as a rudimentary staking framework.

1. Proof-of-Stake Consensus Protocol.
The most popular alternative to the Proof-of-Work (PoW) consensus protocol for blockchains is Proof-of-Stake (PoS) [4]. The PoS protocol (semi-)randomly assigns a staker, who has locked some cryptocurrency, the authority to create a new block, and update the blockchain and rewards them if the updated chain achieves consensus (i.e., is approved by other stakers) [6]. Actors sending transactions on the blockchain benefit from stakers who ensure that the blockchain is in a trustworthy state.

2. Staking in goods and services.
Distributed marketplaces for (digital) goods and services use blockchain and smart contracts for impartial enforcement of purchasing and are becoming more and more popular[1]. In order to guarantee the quality of an (off-chain) product

[1] See for example Ocean: https://oceanprotocol.com/, OpenBazaar: https://www.openbazaar.org/, CanYa: https://canya.io/, BitBay: https://bitbay.market/decentralized-marketplace/

being offered through such an (on-chain) decentralized marketplace, the marketplace can ask the stakers to stake some assets on high-quality products and reward stakers that stake in popular products. Both the seller and the buyer on the market can profit from the independent quality assurance provided by the staker in this scenario.

3. Staking in service monitoring.
Recent initiatives have proposed moving the management of Service Level Agreements (SLAs) to smart contracts on a blockchain [7, 3]. Checking that the (off-chain) service lives up to the Quality-of-Service (QoS) and non-functional requirements captured in the (on-chain) smart contract representation of the SLA depends on the correct monitoring of the service, which necessarily happens off-chain. This monitoring can be done by third parties who stake on their ability to provide independent, high-quality monitoring. These stakers take up the role of an oracle and are rewarded by the smart contract if they manage to achieve consensus on their measurements.

Based on the 3 cases introduced above, we generalize here four fundamental characteristics of the staker actor:

1. The staker provides supportive services with respect to trustworthiness to a process by *measuring/testing* or *guaranteeing* some of its quality aspects.
2. This process often runs, at least partially, off-chain and the staker is generally not the main actor in the process.
3. The staker demonstrates that they are invested in the quality of the process by backing up the accuracy of their measurements/tests/guarantee with some staked asset(s) on the blockchain (e.g., cryptocurrency).
4. The staker follows a protocol that allows it to check and be checked by other stakers.
5. The staker is rewarded by the process, depending on the value of their contribution and the size of their stake.

3 Problem Definition

As argued in section 2, there is increasing recognition of the usefulness of staking as a means to leverage trustworthiness to a decentralized process. Unfortunately, the perspective of the staker in these processes has thus far been rather rudimentary and overly simplistic in nature. Typically, existing methods and tools merely assume the staker's decisions to be taken without considering how exactly this decision process takes place.

However, staking is indeed an exceedingly challenging and critical endeavor since it concerns guaranteeing the quality of processes that may reside outside the staker's sphere of control. We claim that the only type of staking already happening in practice is staking on fully transparent processes such as the PoS in Scenario 1. The challenging nature of such an actor, who has little to no control over the process, has already been recognized in the past, for example, in the context of Service-Oriented-Architecture (SOA) testing by Canfora and

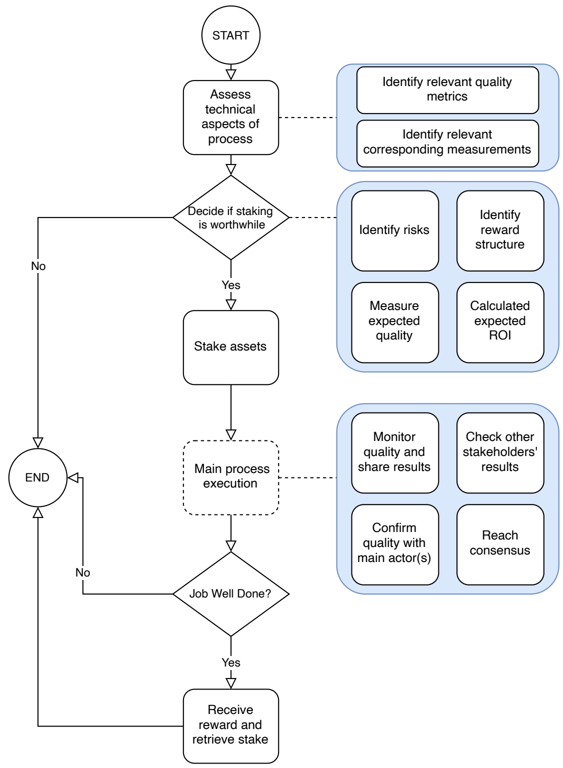

Fig. 1. The process of staking and the different steps involved in the necessary decisions of stakers. The main process execution is outside of the stakers control and is therefore not shown with a solid line around the box.

Di Penta [2]. We illustrate the stakers perspective using the generalized process of a staking process depicted in fig. 1.

When selecting a process to stake on, the staker should start by assessing the technical aspects of the main process that affect its (promised) quality and decide how and when these should be measured. These can be either the quality of a good, such as in Scenario 2, the relevant quality metrics of the process, such as in Scenario 1, or both, such as in Scenario 3. This decision is ultimately based on the staker's expected Return-on-Investment (ROI), which in turn depends on the perceived (expected) quality, the perceived risk, and the expected reward that results from the staking process. If the staker decides to stake, they lock some of their assets on the blockchain, at which point the main process can commence. The role of the staker during the main process depends entirely on the nature of said process; it can be that the staker is not required to do anything, as in Scenario 2, or that the staker has to monitor and guarantee the quality during the main process and, achieve consensus with other stakers, such as in Scenarios

1 and 3. Either way, once the process is over, the staker gets rewarded, usually in proportion to the usefulness of their contribution and the size of their stake.

A crucial step in the flowchart in fig. 1 is the decision of whether staking is worthwhile. The staker wants to maximize their expected ROI. This requires that the staker balances the promised reward against the risk of unintended behavior in the process, which can violate the quality that the staker is guaranteeing. In order to assess this risk, the staker has to have a good understanding of the process it is staking in from the perspective of the main actor(s), who control the process, *as well* as the staking and/or consensus protocol. By staking, the staker effectively guarantees the quality of both the main process, which they deem likely enough to be used, *and* the (un)likelihood that the process will contain unintended behavior. We have found a surprising lack of literature, theory, and tools that address this challenge and propose our own ideas on this in the next section.

The systems architect should take into account that the staker's objective (maximizing ROI) does not necessarily align well with the objectives of the other actors in the system. This, in turn, means that the service that the supportive services mentioned in section 2 staker is another problem that arises from the lack of consideration for the staker's perspective. We have found that existing literature on staking in blockchain-based scenarios consistently fails to address this potential misalignment between the staker and the other actors in the system.

4 Proposed Solution

As a way to address the challenges introduced in section 3, we propose the introduction of a staking framework, which can assist both aspiring stakers as well as systems architects in the process of staking and the decisions associated with this process. This framework is bound to contain at least the following supportive elements:

1. A "cookbook" which will contain standard patterns, best practices in staking scenarios and, how to deal with them;
2. A number of techniques that can be leveraged for assessing the process, the staking protocol, the alignment between the staker and, the other actors;
3. A tool suite which can be used to streamline the staking process outlined in fig. 1 by automating (part of the) the required risk assessment and the connected orchestration machinery;

We illustrate the usefulness and expected impact of our proposed framework by describing a hypothetical staker who participates in scenario 3 following the flowchart in fig. 1.

When designing a blockchain-based solution that involves staking, a systems architect considers the framework's standard patterns and best practices to help them select the best option for their system. Additionally, they consider whether the objectives of the staker and the other actors line up well. Before selecting a service, the staker uses our proposed framework to identify relevant patterns and best practices regarding quality metrics (e.g., QoS & non-functionals). They leverage our assessment techniques to make sure they understand which measurements allow them to monitor these quality metrics effectively. Using this knowledge and our proposed tool suite, the staker identifies scenarios where they could fail to achieve consensus (and thus not get rewarded for staking). Finally, after having confirmed their understanding of both the process they are staking in **and** the process of staking itself can they select processes that have acceptable expected ROIs.

Once the decision is made, the staker stakes their assets and participates in the monitoring process and, if consensus is achieved, it gets rewarded by the smart contract that enforces the SLA. For a more in-depth explanation of this scenario, we invite the reader to check Uriarte et al. [7].

5 Roadmap and Contributions

This paper has outlined the contours of a methodological framework for effectively supporting stakers, -and the process of staking- in a blockchain environment. The results of this paper are core research results in nature. More research is required in several main directions.

Firstly we intend to conduct one or more case studies, possibly followed by a survey. This will help us map more clearly how the proposed challenges are experienced first-hand by the actors in blockchain-based staking scenarios. It is likely this will result into a call for research based on the challenges we identify.

Secondly, we intend to apply action research to solve the problems arising from the challenges identified in this paper. This will provide a proof-of-concept for future endeavors, both in academics and in business, that aim to solve these challenges.

Finally, we hope to generalize a framework, supported by theory, that can be applied to the set of problems described in this paper and addressed in the previous step. This will both open up the door for further theory development regarding staking in blockchain scenarios, as well as directly benefit both stakers and systems architects.

References

1. Butijn, B.J., Tamburri, D.A., Heuvel, W.J.V.D.: Blockchains: a systematic multivocal literature review. arXiv preprint arXiv:1911.11770 (2019)
2. Canfora, G., Di Penta, M.: Service-oriented architectures testing: A survey. In: Software Engineering, pp. 78–105. Springer (2007)
3. Foundation, O.P., GmbH, B., Circus, N.: Ocean protocol: A decentralized substrate for ai data & services technical whitepaper (2019), https://oceanprotocol.com/tech-whitepaper.pdf
4. Irresberger, F., John, K., Saleh, F.: The public blockchain ecosystem: An empirical analysis. Available at SSRN (2020)

5. Mik, E.: Smart contracts: terminology, technical limitations and real world complexity. Law, Innovation and Technology **9**(2), 269–300 (2017)
6. Saleh, F.: Blockchain without waste: Proof-of-stake. Available at SSRN 3183935 (2020)
7. Uriarte, R.B., Zhou, H., Kritikos, K., Shi, Z., Zhao, Z., De Nicola, R.: Distributed service-level agreement management with smart contracts and blockchain. Concurrency Computation (March), 1–17 (2020). https://doi.org/10.1002/cpe.5800