Ziegler, Michael Herbert; Großmann, Marcel; Krieger, Udo R.

# Integration of Fog Computing and Blockchain Technology Using the Plasma Framework

## Primary publication

Ziegler, Michael Herbert; Großmann, Marcel; Krieger, Udo R. (2019): Integration of Fog Computing and Blockchain Technology Using the Plasma Framework, in: IEEE International Conference on Blockchain and Cryptocurrency, 14-17 May 2019, Seoul, South Korea, Piscataway, NJ: IEEE, pp. 120–123, doi: 10.1109/BLOC.2019.8751308.

# Integration of Fog Computing and Blockchain Technology Using the Plasma Framework

Michael Herbert Ziegler
*Fakultät WIAI, Otto-Friedrich-Universität*
D-96045 Bamberg, Germany
Email: michael-herbert.ziegler@stud.uni-bamberg.de

Marcel Großmann, Udo R. Krieger
*Fakultät WIAI, Otto-Friedrich-Universität*
D-96045 Bamberg, Germany
Email: {marcel.grossmann, udo.krieger}@uni-bamberg.de

***Abstract*—Considering the Internet-of-Things, the blockchain technology has recently become a major focus of research. One of its major drawbacks is given by the poor performance of blockchain systems subject to heavy load. In particular, systems which use Proof-of-Work as leader election strategy are not suitable for an active participation of IoT devices. We propose a new system architecture which uses the Plasma framework to integrate blockchain technology and fog computing and evaluate the performance of its prototype. The Plasma framework has the advantage to provide a scalable hierarchical design based on sidechains and an off-chain scaling strategy which is agnostic with regard to the architecture of the employed root chain.**

***Index Terms*—Blockchain, Fog Computing, Scaling, Sidechain, Plasma**

## I. INTRODUCTION

Nowadays, the *Internet-of-Things* (IoT) assists humans in everyday situations on a global scale. These IoT systems rely on a vast amount of data that are collected by their sensors to enable smart decision-making. It is not the primary focus of these devices to communicate directly with the end users. Rather, they convey messages among each other [1].

As the focus of IoT devices lies on a specific task, they are only equipped with the necessary amount of hardware to fulfil that task. In addition, IoT devices usually do not have access to a big amount of energy and, thus, they are not capable to handle complex computations beyond their intended purpose. Therefore, the IoT relies heavily on cloud services which provide reasoning on the transmitted data and relay results back to the next machine in the underlying process chain of an IoT application.

Not all IoT devices have the hardware that is necessary to communicate directly with cloud services. Intermediaries or brokers can help to bridge that gap providing not only communication channels, but also local services. Assembling these virtualized broker nodes, one can form a *fog computing layer*. This middleware is geographically distributed by nature since fog nodes need to be physically close to their connected client devices [2]. Combined with the paradigm of fog computing, the Internet-of-Things will provide a fast growing basis for new, rapidly evolving application scenarios of the blockchain technology.

A blockchain is a decentralized database which is distributed via replication among its contributing peers. Data are not directly updated, but a collection of change records represented by transactions are appended as aggregated entities called blocks. All participating peers can validate the state changes and must agree on the state of those data stored in the associated data plane. Thus, a blockchain represents the immutable history of state changes triggered by the transactions of the peers up to the most recent block.

The blockchain technology has been introduced by Satoshi Nakamoto [3] in 2008 to realize a truly decentralized digital currency called *Bitcoin* in a peer-to-peer interaction mode among its users without the need of a centralized authority. Its underlying blockchain technology is not limited to simple value transfers, but it has the potential to create a whole set of new decentralised applications. Deploying blockchain systems on a fog computing layer could therefore not only lessen the need for cloud services, but also completely remove the latter [4]. Participating in blockchain systems is resource intensive. Transactions need to be signed, uploaded, verified and a consensus protocol must be executed. Currently, only a small set of fog nodes is able to participate actively in these systems. This issue occurs in particular in those systems applying a Proof-of-Work scheme [3]. On the other hand, blockchain systems are currently not able to handle the projected load of the IoT. Bitcoin, for instance, is capable to send only at a rate of 3.3 – 7 transactions per second which is determined by the time between blocks, the transaction size and the maximum block size [5].

Regarding blockchains several scalability solutions are currently under development. They focus on accelerating the blockchain system itself by using alternative verification techniques and new consensus mechanisms. Another field of research is to offload work from blockchain systems in such a way that off-chain state changes can be verified on-chain.

Here we describe and evaluate a scaling technique for blockchains given by the *Plasma* framework [6]. It is blockchain system agnostic and only a few base requirements have to be satisfied. Furthermore, application systems can nest Plasma chains.

In this paper we first consider related work on the blockchain technology. Then we propose a new fog computing architecture which utilizes fog cells as Plasma application systems. The range of applications based on the fog computing layer can be expanded by adding blockchain technology to the portfolio of fog gateways. Then an implementation of our integrated architecture is evaluated. We check its performance in a test bed with a Raspberry Pi system adopting the roles of a Plasma chain operator and a client node. Finally, some conclusions are drawn from the sketched feasibility study that is based on our proof-of-concept.

## II. RELATED WORK

Blockchain technology and its foundations are a vivid area of current research, see [4]–[11]. In [4] a distributed fog computing layer has been proposed which cooperates with a distributed cloud. The distributed cloud and fog computing layer use blockchain technology with a hybrid form of Proof-of-Work (PoW) and Proof-of-Stake (PoS). The distributed cloud administrates the related fog cells. The authors of [7]–[9] also recognize a need to offload transactions from blockchain systems. Smart devices communicate with the overlayed blockchain by their local fog node. Consensus is reached by employing a reputation based system. The concept focuses in particular on transacting in the edge layer.

While all these concepts try to scale a blockchain system, they additionally make several assumptions regarding the users, software and hardware. These assumptions might work for some applications, but they are not flexible enough for all existing types. Furthermore, a new core blockchain is necessary for many of these approaches.

## III. THE BLOCKCHAIN FRAMEWORK PLASMA

The blockchain framework *Plasma* [6] uses sidechains as a means to scale. The latter are blockchains that validate data from other blockchain systems [12]. The framework extends the idea of payment channels by structuring off-chain work in its own blockchain. The child chain does not have to share any properties with the parent chain. It may implement a different block structure, block time, consensus algorithm, etc. The Plasma chain can even have a short lifetime.

The Plasma framework does not dictate strict specifications since each Plasma chain needs different properties depending on its application. A Plasma chain enables to execute off-chain transactions while retaining the security of the parent chain. Off-chain work is structured in blockchain systems itself, the Plasma chains. Multiple Plasma chains can coexist at the same time and, thus, they can have a short life time. This concept works by periodically sending Merkle roots of Plasma chain blocks to the parent chain. If a user wants to exit a Plasma chain, she can prove her state on the Plasma chain to the parent chain by means of a Merkle proof [6]. Each transmitted block proof of a Plasma chain represents all transactions made in the respective block. Thus, with one block submission from a Plasma chain to a parent chain, a high number of transactions is committed.

The parent blockchain system provides the baseline of security to which users of Plasma chains can fallback in case of fraud. In this way security constraints in Plasma systems can be reduced and the system itself is accelerated. The reliance on the security of the parent chain enables to use weaker consensus algorithms like Proof-of-Authority [13] (PoA) or Delegated PoS [14]. As an off-chaining concept, Plasma enables to strongly increase the throughput of a blockchain system. The number of Plasma chains which a parent chain can support is only limited by hardware constraints. As the blockchain framework Plasma is blockchain agnostic, only a few base requirements have to be satisfied. Creating a new Plasma chain is possible with a minimal setup phase, where a smart contract needs to be deployed on the parent blockchain. Not only basic value transfers, but also the execution of smart contracts are supported. In this way specialized child systems of a blockchain can be created for specific use cases.

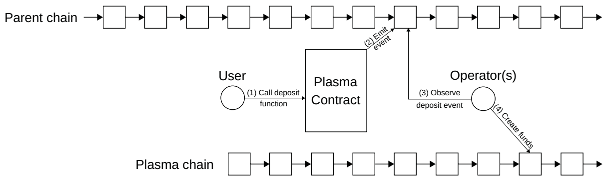

Fig. 1: Interaction between a Plasma chain and its parent chain.

The interactions between a Plasma chain and an initially generated root chain, e.g. Ethereum [11], as well as the involved functional agents are illustrated in Figure 1. To gain entry into a Plasma chain, a user needs to deposit those funds that she wants to use on the Plasma chain to the Plasma contract. She does that by calling a designated function of the Plasma contract (1) on the parent chain which emits a deposit event (2) on the latter. The operator of the Plasma chain monitors the parent chain continuously for state changes of the Plasma contract and, thus, observes the event (3). When the operator notices the deposit of the user, it creates a deposit transaction in the Plasma chain (4) and distributes it with the next block to other nodes of the Plasma network. As soon as the user gets the block with her deposit, she can start to transact on the Plasma chain. With the received block header, she can also immediately exit the chain. If she does not get a deposit on the Plasma chain as a result of fraudulent behaviour of the operator, she can withdraw her funds from the contract on the parent chain after a contest period [6]. This basic behavior and related interaction patterns should be emulated efficiently by a prototype implementation of the Plasma framework in a fog computing environment.

## IV. A FOG-COMPUTING ARCHITECTURE INTEGRATING THE BLOCKCHAIN FRAMEWORK PLASMA

We have pointed out that blockchain systems are suitable candidates to be integrated into a fog computing architecture. Using the sketched Plasma framework, not only fog nodes can be assimilated into a blockchain system, but also edge devices. Low powered IoT nodes can adopt the role of clients in a Plasma system, while fog nodes act as operator nodes. Fog cells manage their connected edge devices and, thus, form powerful application systems. The application-centric view can

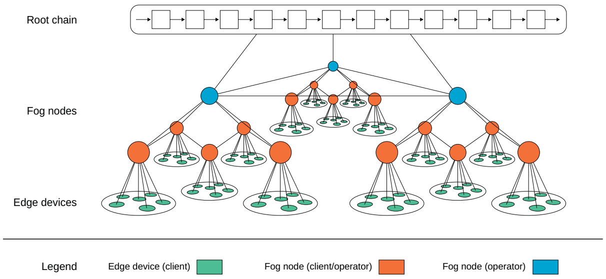

Fig. 2: Proposed architecture integrating fog computing and the blockchain concept of Plasma. Edge devices (green) communicate with fog gateways (red), which in turn communicate with higher level fog gateways (blue) and their Plasma instances that are associated with the root chain on top.

also be applied to Plasma systems, i.e. the latter manages the data of one or several similar application systems.

Following these design principles, we propose a new, hierarchically structured, integrated fog-blockchain architecture shown in Figure 2. It comprises two layers of Plasma systems built on top of each other. Systems at the lower level directly manage edge nodes as their clients that communicate with fog gateways. The latter in turn communicate with higher level fog gateways which also host Plasma instances. These higher level Plasma systems view these lower level systems as their clients and communicate directly with the root blockchain. In our prototype architecture the latter component is realized by means of Ethereum [11].

## V. EVALUATION OF THE INTEGRATED ARCHITECTURE

Regarding the case study of our proof-of-concept, the underlying integrated fog-blockchain architecture is depicted in Figure 3. Using Ethereum as parent chain, we have realized one Plasma chain based on the *Minimal Viable Plasma*[1] (MVP) implementation of OmiseGo in this architecture. The Plasma chain is completely controlled by one Plasma operator. PoA is used as consensus protocol and, therefore, mining of blocks is not necessary. Transactions can be included in blocks and transaction finalization on the Plasma chain is very fast. The operator shares data with its users and handles the commitment of blocks as well as deposit recognition. Users do not interact with the Plasma chain directly, but must send their transactions to the operator. The operator then executes the tasks on the user's behalf. Users and operators can interact with the Ethereum root chain via a Web3-client. This client offers a JSON-RPC API which provides read and write access to the parent chain. In practice users and operators do not have to use the same Web3-node, but can either run their own one or, depending on the use case, use a public node. Interactions between an operator and the users are realized via a HTTP-REST API. Users can send transactions to the operator and kick off the submission of Plasma blocks. If users want to enter or exit the system, they interact with the Web3-client directly. Deposit events are in turn observed by the operator via queries to the Web3-client. Note that deposit events are actual smart contract events emitted by the Plasma contract, rather than local data saved on the Web3-node.

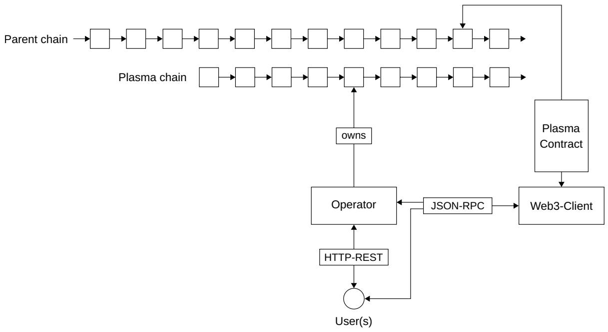

Fig. 3: Users and an operator interact with the Ethereum parent chain via a Web3-client.

To execute this Plasma implementation on a client and an operator node in a virtualized fog computing environment and to evaluate its performance, the developed modules were first separated and containerized into Docker images. Then the realized Plasma system has been evaluated in terms of two test cases. Namely, a single-board computer system (SBC) of the type Raspberry Pi 3 B Plus (RPi) with HypriotOS 1.9 and an ARMv7@1.4 GHz CPU was employed as *client* node of a user and alternatively as *active operator* node, see Figure 3. In both test scenarios the total number of executed transactions was set to 2000 items. The resulting CPU usage of the Raspberry Pi system and the realized throughput of transactions have been evaluated as basic performance metrics.

### A. Test Scenario of a Client

In the first test scenario the Raspberry Pi system acts as a client of a blockchain user and constantly sends transactions to the operator node. Transactions are simple value transfers as MVP does not support a smart contract execution on the Plasma chain. The 2000 transactions are completed in 2 minutes and 49 seconds yielding an average throughput of *11.83 transactions per second*. Figure 4 depicts the CPU frequency related to all transactions. The periodical dips arise from the commitment of Plasma blocks at the operator side. The CPU reaches its maximum usage during the active emission of transactions as expected.

### B. Test Scenario of an Operator Node

The RPi SBC has also been tested as an active operator node of our integrated fog-Plasma architecture in Figure 3. Sending 2000 transactions, the RPi operator node has been able to receive, store and submit all transactions in 16 minutes and 1 second yielding an average throughput of *2.08 transactions per second*. Figure 5 shows that one core of the RPi SBC has always been at a maximum utilization level when transactions were submitted to the parent chain. In contrast to that, receiving transactions resulted in a slightly

---

[1] URL: https://github.com/omisego/plasma-mvp, Accessed 19. Dec 2018

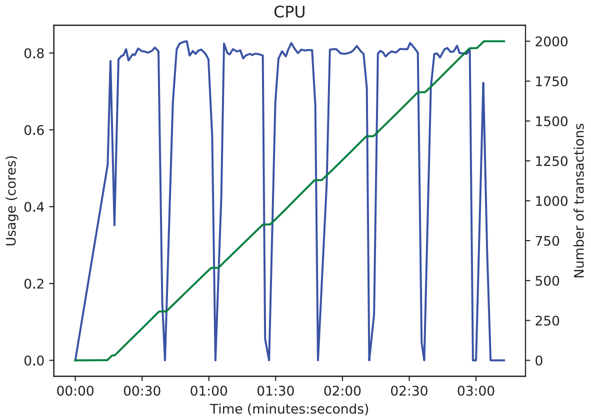

Fig. 4: CPU usage (blue) and the number of transactions (green) over time with a RPi SBC as client.

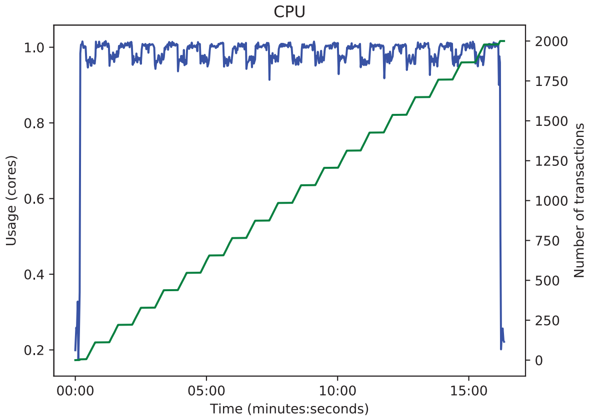

Fig. 5: CPU usage (blue) and the number of transactions (green) over time with a RPi SBC as operator.

lower utilization. This outcome has the following reason. To submit the current Plasma block to the Ethereum parent chain, the Merkle root has to be computed. The related calculation of the root executes computationally expensive cryptographic routines. Thus, a bulk of the time is consumed by submitting the Plasma blocks. In contrast to that, this operation is less stressful on the CPU of the RPi SBC when the latter acts as a client.

## VI. CONCLUSION

We have pointed out several problems which limit the application of current blockchain systems in a fog computing environment for advanced IoT applications. Possible solutions have been discussed that focus on moving work away from the parent blockchain. Consequently, we have described the Plasma framework that has been developed for this purpose. Then we have proposed a potential architecture which combines the blockchain framework Plasma and fog computing. After that Plasma has been evaluated in a fog computing environment based on a test bed of Raspberry Pi nodes. The MVP implementation used in our proof-of-concept demonstrates that Plasma can scale blockchain systems and has revealed some promising performance results.

Plasma has already exhibited its high potential and differentiates itself from other off-chaining solutions, in particular by its minimal requirements with regard to the parent blockchain and the easy setup by the implementor. Current research on Plasma focuses on minimizing the size of proofs [10], [15], [16], on fastening the finality with state channels and on exploring the usage of zero-knowledge proofs. This development will enable improved efficiency which is necessary to run real-world IoT applications and constitutes a subject of our future research.

## ACKNOWLEDGMENT

The authors would like to thank the implementors of OmiseGo who provided the used MVP implementation.

## REFERENCES

[1] K. Ashton, “That ’Internet of Things’ Thing,” 2009. [Online]. Available: https://www.rfidjournal.com/articles/pdf?4986

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, ser. ACM Digital Library, M. Gerla, Ed. New York, NY: ACM, 2012, p. 13.

[3] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[4] P. K. Sharma, M.-Y. Chen, and J. H. Park, “A Software Defined Fog Node Based Distributed Blockchain Cloud Architecture for IoT,” *IEEE Access*, vol. 6, pp. 115–124, 2018.

[5] K. Croman, et al., “On Scaling Decentralized Blockchains,” in *Financial Cryptography and Data Security*, ser. LNCS sublibrary. SL 4, Security and cryptology, J. Clark, S. Meiklejohn, P. Ryan, D. Wallach, M. Brenner, and K. Rohloff, Eds. Berlin: Springer, 2016, vol. 9604, pp. 106–125.

[6] J. Poon and V. Buterin, “Plasma: Scalable autonomous smart contracts,” 2017. [Online]. Available: https://plasma.io/plasma.pdf

[7] A. Dorri, S. S. Kanhere, and R. Jurdak, “Blockchain in Internet of Things: Challenges and Solutions,” *CoRR*, 2016. [Online]. Available: http://arxiv.org/pdf/1608.05187v1

[8] A. Dorri, S. S. Kanhere, and R. Jurdak, “Towards an Optimized Blockchain for IoT,” in *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation - IoTDI ’17*, Unknown, Ed. New York, New York, USA: ACM Press, 2017, pp. 173–178.

[9] A. Dorri, et al., “Blockchain for IoT security and privacy: The case study of a smart home,” in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. Piscataway, NJ: IEEE, 2017, pp. 618–623.

[10] G. Konstantopoulos, “Plasma Cash: Towards more efficient Plasma constructions,” 2018. [Online]. Available: https://github.com/loomnetwork/plasma-paper/blob/master/plasma_cash.pdf

[11] G. J. Wood, “Ethereum: A Secure Decentralised Generalised Transaction Ledger,” 2014. [Online]. Available: https://ethereum.github.io/yellowpaper/paper.pdf

[12] A. Back, et al., “Enabling Blockchain Innovations with Pegged Sidechains,” 2014. [Online]. Available: https://blockstream.com/sidechains.pdf

[13] G. Wood, “PoA Private Chains,” 2015. [Online]. Available: https://github.com/ethereum/guide/blob/master/poa.md

[14] Bitshares Foundation, “Delegated Proof-of-Stake Consensus,” 2018. [Online]. Available: https://bitshares.org/technology/delegated-proof-of-stake-consensus

[15] V. Buterin, “RSA Accumulators for Plasma Cash history reduction,” 2018. [Online]. Available: https://ethresear.ch/t/rsa-accumulators-for-plasma-cash-history-reduction/3739

[16] I. Gulamov, “Plasma Prime design proposal,” 2018. [Online]. Available: https://ethresear.ch/t/plasma-prime-design-proposal/4222