Performance Analysis of a Hyperledger Fabric Blockchain Framework: Throughput, Latency and Scalability

Murat Kuzlu[1], Manisa Pipattanasomporn[2,3], Levent Gurses[4] and Saifur Rahman[2]
[1] Electrical Engineering Technology, Old Dominion University, Norfolk, VA, USA
[2] Bradley Department of Electrical and Computer Engineering, Virginia Tech – Advanced Research Institute, Arlington, VA, USA
[3]Smart Grid Research Unit, Chulalongkorn University, Bangkok, THAILAND
[4]Movel, Inc., Sterling, VA

***Abstract*— Focusing on one of the most popular open source blockchain frameworks—Hyperledger Fabric, this paper evaluates the impact of network workload on performance of a blockchain platform. In particular, the performance of the Hyperledger Fabric platform is evaluated in terms of: (a) throughput, i.e., successful transactions per second; (b) latency, i.e., response time per transaction in seconds; and (c) scalability, i.e., number of participants serviceable by the platform. The results indicate that the instance of Hyperledger Fabric platform being implemented can support up to 100,000 participants on the selected AWS EC2 instance. As long as the transaction rate is maintained within 200 transactions per seconds, the network latency is in the order of fraction of a second.**

***Index Terms*—Blockchain; Hyperledger fabric; throughput; latency.**

I. INTRODUCTION

Blockchain is the distributed ledger technology (DLT) having significant features, such as decentralized and trustable ledger of records. Since It was first introduced in 2008 [1], the blockchain technology today goes far beyond Bitcoin cryptocurrency applications. It has become one of the emerging and leading techologies that has revolutionized financial services [2], supply chains [3], healtcare [4], energy [5] and public services [6]. The main advantage of using the blockchain technology is its ability to support a publicly distributed ledger, where an identical copy of the ledger is replicated throughout a blockchain network. It also allows transactions to be anonymously secured among business partners, and automatically verifies and records data using cryptographic algorithms without the need for a central authority or intermediary.

So far, several blockchain frameworks have become available that provide adaptable and flexible platforms supporting various applications. These platforms include Hyperledger Fabric [7], Ethereum [8], Corda [9], Omni [10], Ripple [11], MultiChain [12], OpenChain [13] and Chain Core [14]. While there are already several blockchain projects being piloted, there exist some concerns about the technical challenges of a blockchain platform in terms of its throughput, latency and its ability to scale [15].

In the literature, several studies discussed the scalability and performance analysis of different blockchain platforms. Performance metrics of different blockchain platforms, mainly Hyperledger Fabric and Ethereum, were compared in [16, 17]. Authors in [16] introduced BLOCKBENCH – the evaluation framework for private blockchains, to analyze major blockchain platforms: Ethereum, Parity and Hyperledger Fabric. In [17], the performance analyses of both Hyperledger Fabric and Ethereum were presented. Findings indicate that the performance of Hyperdeger Fabric outperform that of Ethereum in terms of latency, throughput and execution time.

Performance evaluation of the Hyperledger Fabric platform was discussed in [18-22]. Authors in [18] analyzed the performance of two Hyperledger Fabric versions (v0.6 and v1.0). Results indicate that, for Fabric v1.0, number of nodes did not impact throughput, latency and scalability; while the performance of Fabric v0.6, on the other hand, decreased with the increasing number of nodes. Authors in [19] studied throughput and latency of Hyperledger Fabric v1.0 by varying the network workloads, including number of chaincodes, channels and peers. In [20], the authors investigated the performance of the Hyperledger Fabric blockchain platform v1.0 considering different block sizes, endorsement policies, number of channels, resource allocation and state database choices. In [21], focusing on Hyperledger Fabric v1.1, authors presented the impact of block size (i.e., size of transactions), peer CPU, and SSD vs RAM disk on blockchain latency and throughput. Scalability was also tested by increasing number of peers. Stochastic Reward Nets were used to analyze throughput, utlization and queue length for each peer of Hyperledger Fabric v1.0 [22]. Additionally, authors in [23] recommended performance metrics for performance evaluation of the Hyperledger Fabric blockchain platform.

Performance evaluation of the Ethereum platform was discussed in [24, 25]. In [24], authors evaluated the performance of the Ethereum blockchain for Geth and Parity

978-1-7281-4693-5/19/$31.00 ©2019 IEEE
DOI 10.1109/Blockchain.2019.00003

(which are most popular Ethereum clients), and investigated the impact of different transaction types on the blockchain performance. Results indicate that the Parity client can support faster transactions than the Geth client. In [25], the test was conducted to analyze the performance of Ethereum blockchain related to its ability to read/write data on a relational database, i.e., MySQL.

A prototype blockchain network was implemented for storing personal health information as discussed in [26]. Performance analysis was conduted and findings indicated low response time (<500 ms) and high availability (98%).

This paper aims to present a method that can be used to investigate the impact of the blockchain network workloads on the performance of the most recent Hyperledger Fabric platform, v.1.4 – the first long term support release [27]. The network workload refers to varying number of transactions, transaction rate and transaction types. Performance of the blockchain network being evaluated include the throughput (in transactions per second, tps), latency (in seconds) and scalability (i.e., number of participants that the blockchain network can serve).

II. Hyperledger Fabric Platform Deployment

To evaluate the platform performance, an instance of a Hyperledger Fabric platform was deployed. The deployment model is depicted in Fig. 1, illustrating major blockchain components, together with the tools used.

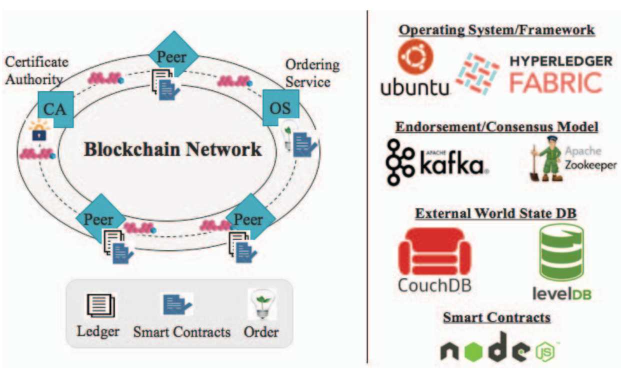

Fig. 1. The Hyperledger Fabric deployment model (top) with its development tools (bottom).

In this study, the entire Hyperledger Fabric deployment was installed and run on the Linux-Ubuntu operating system. Each component of the Hyperledger Fabric, including peers, Ordering Service (OS) and Certificate Authority (CA), was launched as a Docker container. Note that: a peer managed by each group of participants is responsible for executing transactions and maintaining its ledger; an OS provides services, such as broadcasting messages, guaranteeing message delivery, etc.; and a CA provides certificate services to blockchain participants.

Hyperledger Fabric allows developers to select either Solo or Kafka to provide an ordering service to the blockchain network. Solo and Kafka are consensus plugin interfaces available for the Hyperledger ordering service. In this development environment, Solo was used as the blockchain network has only one orderer. The Hyperledger Fabric supports either LevelDB or CouchDB as state database options; and CouchDB was used in deployment model.

The Hyperledger Fabric also supports Golang, Javascript and Java programming languages to execute smart contracts. The smart contracts in this study were written in Javascript and the Hyperledger Fabric Client SDK for Node.js was used to interact with the Hyperledger Fabric blockchain network.

III. The Benchmark Tool and Test Environment

This section discusses the Hyperledger benchmark tool selected for the experiment, its setup and the test environment.

A. *Hyperledger Benchmark Tool*

Hyperledger Caliper [28]–a blockchain benchmark tool– was selected for evaluating the blockchain implementation performance. The main advantage of using Hyperledger Caliper is that it represents many clients that can inject workloads in the blockchain network, i.e., multiple client threads. Fig. 2 depicts the deployment model of the Hyperledger Caliper platform for performance evaluation.

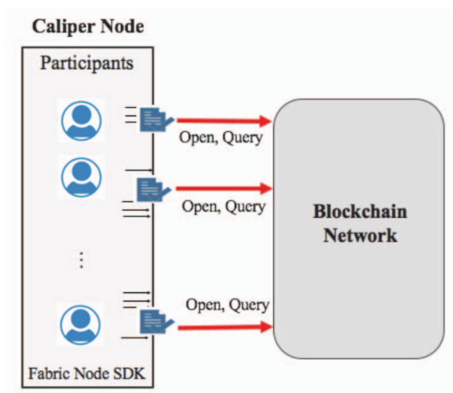

Fig. 2. The deployment model of Hyperledger Caliper for performance evaluation.

B. *Setup of the Hyperledger Caliper Benchmark Tool*

To set up the test benchmark tool, the benchmark configuration file was configured to vary transaction rates, transaction numbers, and transaction types.

- Transaction number (txNumber) defines the number of transactions generated in each round, i.e., txNumber = [1000, 5000] means that 1000 transactions are generated in the first round; and 5000 are generated in the second.
- Transaction rate (rateControl) defines the transaction rate during the benchmarking test sub-rounds. For the conducted experiments, the fixed-rate controller was used, in which input transactions were sent at fixed intervals specified as transactions per second, tps.
- Transaction type has two options as specified by Caliper, i.e., open and query. An "open" transaction performs one

read and one write operations in one transaction, while a "query" transaction simply reads the state from CouchDB.

The blockchain network configuration file was also configured, which defines the network parameters, such as the organizations, peers, nodes, the channel name, etc.

In this study, the impact of varying transaction numbers, transaction rates and transaction types on blockchain performance is of interest. Hence, these are parameters to be varied in the experiment, while network configuration remains fixed throughout the experiment.

C. *Test Environment*

AWS EC2 instance having 16 vCPUs, 3.0 GHz Intel Xeon Platinum processors and 32GB RAM was used to run the test benchmark platform. The AWS EC2 instance ran Ubuntu 16.04 LTS and peers, CA, OS, and Caliper with Fabric release v1.4. This test environment was used to understand the impact of the hardware selection, i.e., CPU and RAM, on the throughput, latency, and scalability, of the blockchain deployment.

IV. Case Study Description

To conduct the performance test on Hyperledger Caliper, several use cases were defined by varying specifications of key parameters. The resulting performance of the blockchain platform was then recorded and analyzed, against three performance metrics:

- Throughput – successful transactions per second
- Latency – response time per transaction
- Scalability – the ability to support increasing number of participants.

Specific parameters used for performance evaluation are summarized in Table I.

TABLE I. Parameters for performance evaluation

Key parameters	Transaction per second (tps)	Number of transactions	Simul-taneous transactions
Case I: Impact of test environment and transaction rates	100, 150, 200, 250, 300tps	1,000	N/A
Case II: Impact of number of transactions	200tps	1,000 10,000 100,000	N/A
Case III: Impact of simultaneous transactions	Equivalent to number of simultaneous transactions	N/A	100 200 300

In Case I, the objective is to study the impact of varying transaction rates on the performance of the blockchain network. Hence, several transaction rates were used, which are 100, 150, 200, 250 and 300 tps. Total number of transactions was fixed at 1000. Simultaneous transactions were not defined.

In Case II, the experiment was conducted to understand the impact of large-scale transactions on cloud-based blockchain throughput and latency. The total number of transactions was varied from 1000, 10000 to 100000. The transaction rate was fixed at 200 tps. Number of simultaneous transactions was not defined.

In Case III, the impact of simultaneous transactions on blockchain performance was evaluated by varying the number of simultaneous transactions (100, 200 and 300).

In all scenarios, both types of transactions (open and query) were tested, and the number of organizations was fixed at two, as this is our expected deployment scenario.

V. Experimental Results

Results obtained from the Hyperledger Caliper test benchmark were interpreted to evaluate the performance of a specific blockchain implementation. Experimental results are discussed below.

A. *Case I: Impact of Transaction Rates*

Fig. 3 shows the average throughput (in tps) and latency (in seconds) at different transaction rates, varying from 100, 150, 200, 250 and 300tps.

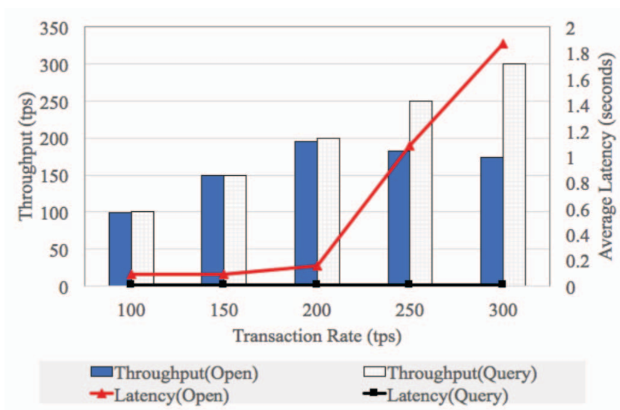

Fig. 3. Impact of transaction rates and transaction types on the blockchain throughput and latency.

It can be seen that, with the "open" type of transaction (where one read and one write were performed per transaction), the blockchain network could only handle up to 200 tps without any significant network latency. When the transaction rate increased above 200 tps, the blockchain throughput decreased as the transaction rate increased and the latency significantly increased. This implies that this specific test environment could handle up to 200 tps of the "open" transaction type without any significant network delay.

For the "query" type of transaction (where only one read was performed in a transaction), the blockchain network obviously could handle 300 tps without any delay. The average latency was still close to zero. This implies that the test environment did not reach its maximum limit, and it could support higher transaction rates.

B. Case II: Impact of the Number of Transactions

To understand the impact of the total number of transactions on blockchain throughput and latency, the total number of transactions was varied from 1,000, 10,000 to 100,000 and these transactions were sent to blockchain at the rate of 200 tps (which was the upper transaction rate limit identified in Case I). Fig. 4 shows the average throughput and latency for both "open" and "query" transaction types at different numbers of transactions.

The experimental results indicate that the total number of transactions has no significant impact on the throughput and latency if the selected transaction rate (tps) can be handled by the hardware. As shown in Fig. 4, the throughput was flat in all cases at 200 tps, which was the transaction rate used. The latency was also flat for "query" transaction at 0.01 seconds, while it was slightly higher for "open" transaction at 0.13-0.16 seconds. For the "open" transaction case, the latency slightly increased when a very high number of transactions was used. The increase in latency was however still considered very small, i.e., 0.03 seconds.

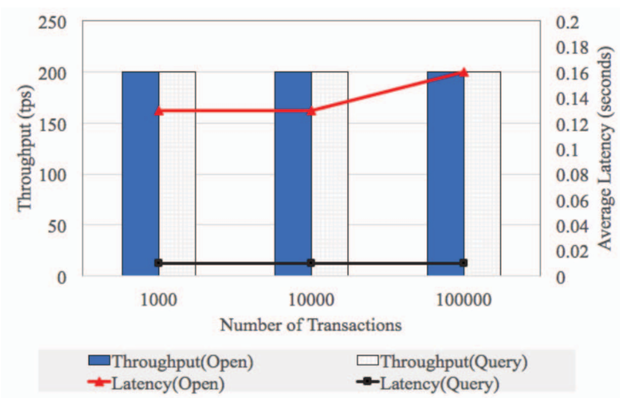

Fig. 4. Impact of the number of transactions on the throughput and latency for "open" and "query" transactions.

These results can be generalized as follows:

- A very high number of transactions can be supported, i.e., 100,000, at a very low latency when the transactions have less operations (only read, not both read and write).
- The type of transactions affects the network latency due to complexity and the number of operations involved even though the blockchain platform is hosted by a high capability hardware system.

In a blockchain network, the scalability can be measured in different ways, i.e., based on the number of channels, the number of peers, the number of organizations, the number of nodes, the number of transactions, the consensus protocol, etc. In this `study, the total number of transactions was selected as the measurement parameter of the scalability. It is assumed that each transaction represents a participant. Based on the results, the implemented Hyperledger platform can support 100,000 participants with the selected hardware configuration. If the hardware configuration has higher spec, higher number of participants can be supported. Hence, there is in fact no limit on the number of participants that can take part in the Hyperledger Blockchain network. It depends on the selected hardware and the blockchain network configurations.

C. Impact of Multiple Simultaneous Transactions

To evaluate the impact of transactions simultaneously submitted by many participants on the blockchain throughput and latency, additional use cases were run under the same test environment. It is assumed that there are 100, 200 and 300 participants, each of which sent a request to blockchain simultaneously. Blockchain throughputs and latency were recorded for both "open" and "query" transactions. Results are shown in Fig. 5. It can be seen that both throughput and latency increase with the increasing number of simultaneous transactions.

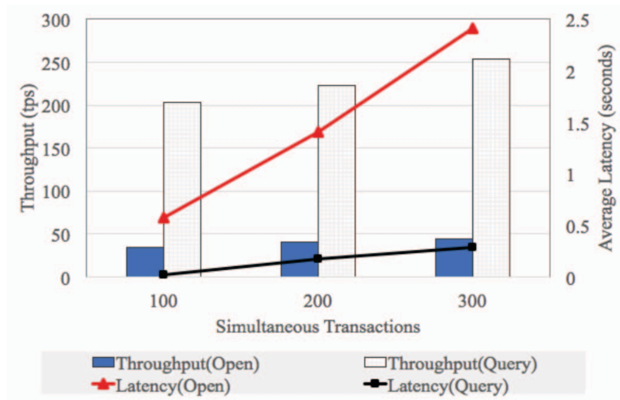

Fig. 5. Impact of the number of participants on throughput and latency.

With "open" transactions, throughputs were 34.6, 40.3 and 44.8 tps for 100, 200 and 300 simultaneous transactions, respectively. Latency also increased with the increasing number of simultaneous transactions. This implies that the system has already reached its maximum capability to handle such number of simultaneous transactions as the blockchain throughput could not catch up with many simultaneous transactions given as the inputs. The increasing latency implies that the blockchain network had to queue the remaining transactions for later processing.

On the other hand, with "query" transactions, the throughput was higher than the "open" transaction, and the latency was significantly lower. This is as expected as "query" transactions perform only one read operation per transaction. Under this case, the system could handle 200 simultaneous transactions without any queuing requirement as the throughput appeared to be higher than 200tps. While at 300 simultaneous transactions, queuing was needed and hence the latency started to increase.

These results can be generalized as follows:

- Under this specific test environment, an increase in simultaneous transactions significantly affects the performance of the blockchain network, in particular the latency.

- Transaction type has a significant influence on the performance of the blockchain network.
- Throughput is relatively flat as the number of simultaneous transactions increases if the system has already reached at its maximum limit.
- Latency increases with the increase in simultaneous transactions if the system has already reached at its maximum limit.

VI. Conclusion

This paper discusses the impacts of the blockchain network workload on performance of the Hyperledger Fabric blockchain framework in terms of throughput, latency and scalability. Several use cases were performed by varying transaction rates (tps), total number of transactions, number of simultaneous transactions and type of transactions in the simulated environment.

Under the test environment with the AWS EC2 instance having 16 vCPUs, 3.0 GHz Intel Xeon Platinum processors and 32GB RAM, the blockchain network could support up to 200tps. This was the transaction rate that the specific blockchain network could support without any significant latency. At 200tps, the blockchain network of interest could easily support 100,000 transactions, i.e., 100,000 participants. The average response time was quite less than anticipated, i.e., 0.01 and 0.16 seconds for 100,000 "query" and "open" transaction requests, respectively. There existed however the impact of simultaneous transactions on network latency and throughput.

In summary, the throughput, latency and scalability of a blockchain network depend on hardware configuration, blockchain network design and smart contact complexity/ operations. These results may be different under different test environments. It is expected that the findings reported in this paper can serve as a guideline that can help select a suitable hardware configuration, as well as a blockchain network and its parameters that can support a specific blockchain implementation and requirements.

VII. References

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," [Online]. Available: http://www.bitcoin.org. Retrieved: April 2019.

[2] 5 blockchain technology use cases in financial services, https://www2.deloitte.com/nl/nl/pages/financial-services/articles/ blockchain-technology-use-cases-in-financial-services.html. Retrieved: April 2019.

[3] How blockchain is revolutionizing supply chain management https://www.ey.com/Publication/vwLUAssets/ey-blockchain-and-the-supply-chain-three/$FILE/ey-blockchan-and-the-supply-chain-three.pdf. Retrieved: April 2019.

[4] Blockchain: Opportunities for health care, https://www2.deloitte.comn/us/en/pages/public-sector/articles/blockchain-opportunities-for-health-care.html. Retrieved: April 2019.

[5] Blockchain Technology Will Transform the Power Industry, https://www.powermag.com/blockchain-technology-will-transform-the-power-industry. Retrieved: April 2019.

[6] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," the Int'l journal of Business & Information Systems Engineering, vol. 59, no. 3, pp. 183–187, 2017.

[7] Hyperledger Fabric [Online]. Available: https://www.hyperledger.org/projects/fabric. Retrieved: April 2019.

[8] Ethereum [Online]. Available: https://www.ethereum.org/. Retrieved: April 2019.

[9] Corda [Online]. Available: https://www.corda.net/. Retrieved: April 2019.

[10] Omni Layer [Online]. Available: https://www.omnilayer.org/. Retrieved: April 2019.

[11] Ripple [Online]. Available: https://ripple.com/. Retrieved: April 2019.

[12] MutiChain [Online]. Available: https://www.multichain.com. Retrieved: April 2019.

[13] Openchain [Online]. Available: https://www.openchain.org. Retrieved: April 2019.

[14] Chain [Online]. Available: https://chain.com/. Retrieved: April 2019.

[15] M. Pilkington, "Blockchain technology: principles and applications," *Research Handbook on Digital Transformations*, pp. 1–39, 2015.

[16] T. T. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K. Tan, "Blockbench: A framework for analyzing private blockchains," *in Proc. the 2017 ACM International Conference*, pp. 1085–1100, Chicago, IL, USA, May 14-19, 2017.

[17] S. Pongnumkul, C. Siripanpornchana and S. Thajchayapong, "Performance Analysis of Private Blockchain Platforms in Varying Workloads," *In Proc. 2017 the 26th International Conference on Computer Communication and Networks (ICCCN)*, Vancouver, BC, Canada, 2017, pp. 1-6.

[18] Q. Nasir, I. A. Qasse, M. A. Talib, and A. B. Nassif, "Performance Analysis of Hyperledger Fabric Platforms," *Security and Communication Networks,* vol. 2018, Article ID 3976093, 14 pages, 2018. https://doi.org/10.1155/2018/3976093.

[19] A. Baliga, N. Solanki, S. Verekar, A. Pednekar, P. Kamat and S. Chatterjee, "Performance Characterization of Hyperledger Fabric," *In Proc. 2018 the Crypto Valley Conference on Blockchain Technology (CVCBT)*, Zug, Switzerland, 2018, pp. 65-74.

[20] P. Thakkar, S. Nathan and B. Viswanathan, "Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform," *2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Milwaukee, WI, 2018, pp. 264-276.

[21] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. D. Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolic, S. W. Cacco and J. Yellick, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," *In Proc. 2018 the Thirteenth EuroSys Conference (EuroSys'18)*, No. 30, Porto, Portugal, April 23-26, 2018.

[22] H. Sukhwani, N. Wang, K. S. Trivedi and A. Rindos, "Performance Modeling of Hyperledger Fabric (Permissioned Blockchain Network)," *In Proc. 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*, Cambridge, MA, 2018, pp. 1-8.

[23] Hyperledger Performance and Scale Working Group, Hyperledger Blockchain Performance Metrics, 2018 [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2018/10/HL_Whitepaper_Metrics_PDF_V1.01.pdf. Retrieved: April 2019.

[24] S. Rouhani and R. Deters, "Performance analysis of ethereum transactions in private blockchain," *In Proc. 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China, 2017, pp. 70-74.

[25] S. Chen, J. Zhang, R. Shi, J. Yan and Q. Ke, "A Comparative Testing on Performance of Blockchain and Relational Database: Foundation for Applying Smart Technology into Current Business Systems," *In Proc. 2018 Int'l Conference on Distributed, Ambient and Pervasive Interactions*, pp. 21-34, Las Vegas, July 15-20, 2018.

[26] A. Roehrs, C. A. Costa, R. R. Righi, V. F. Silva, J. R. Goldim and D. C. Schmidt, "Analyzing the performance of a blockchain-based personal health record implementation," *Journal of Biomedical Informatics*, Vol. 92, April 2019.

[27] Hyperledger Fabric, "What's new in v1.4: Hyperledger Fabric's First long term support release," 2019 [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-1.4/whatsnew.html. Retrieved: April 2019.

[28] Hyperledger Caliper [Online]. Available: https://www.hyperledger.org/projects/caliper. Retrieved: April 2019.