ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 12, December 2024

Performance and Scalability Analysis of Ethereum, Hyperledger Fabric, and IOTA

[1] Sukanya Kumari, [2] Jawahar Thakur

[1] Student, Department of Computer Science, Himachal Pradesh University, Shimla, Himachal Pradesh, India
[2] Professor, Department of Computer Science, Himachal Pradesh University, Shimla, Himachal Pradesh, India
Corresponding Author Email: [1] sukanyabhardwaj2123@gmail.com, [2] jawahar.hpu@gmail.com

Abstract*— Blockchain has become an increasingly prevalent subject in the last decade, with potential uses in supply chain, healthcare, and finance, among other areas. In academics and industry, its immutability, transparency, security, and decentralization attracted much attention. The Internet of Things is one technology that blockchain can assist (IoT). Yet, issues with blockchain's scalability, performance, and complexity continue to pose obstacles to its practical use. With specific focus on Internet of Things applications, this study adds to a comparative analysis that evaluates the performance and scalability of blockchain platforms. We focus on Ethereum, Hyperledger Fabric, and IOTA blockchain platforms. An IoT healthcare use case is created as part of the deployment. To determine the throughput and latency criteria, we tested the scalability and performance of private platform networks. We looked at how the platforms under study behaved when the number and rate of transactions increased to assess scalability. Hyperledger Caliper is used to collect these parameters. Experiment analysis shows that Hyperledger Fabric performs better than Ethereum and IOTA in terms of transaction throughput and latency, making it highly suitable for enterprise applications that require high performance and scalability within a permissioned network. Regarding performance and scalability, Fabric was more ideal than IOTA, followed by Ethereum for private networks such as IoT healthcare ecosystems.

***Index Terms*— Blockchain, Ethereum, Hyperledger fabric, IOTA, Internet of Things, performance, scalability.**

I. INTRODUCTION

Blockchain is a promising technology that has received a lot of attention in the past decade. Since Nakamoto first revealed it in 2008, its powers have expanded well beyond cryptocurrency. It transforms the ways that current technologies are applied and opens up previously unexplored application domains. Blockchain empowers people through recognized identification and asset ownership in applications including healthcare, supply networks, high-level organizations, and the financial sector. The decentralization, immutability, transparency, and security of the blockchain are its main advantages [1].

According to Swan et al. [2], the blockchain progression is as follows: Blockchain 1.0 comprises cryptocurrencies like Bitcoin, Blockchain 2.0 consists of smart contracts like Ethereum, and Blockchain 3.0 consists of Decentralized Applications (DApps) made possible by smart contracts. Blockchain technology has advanced in response to this development, and prominent businesses have launched blockchain initiatives across the globe [3]. As a result, integration with other technologies—like the Internet of Things (IoT)—has become viable [4]. Another cutting-edge technology that is far from maturity is IoT. IoT's drawbacks, including security, privacy, heterogeneity, interoperability, and maintenance, have emerged as its uses have grown [5]. Some of the problems with IoT applications may be resolved with the help of blockchain.

Blockchain has significant throughput and latency limitations in contrast with existing transaction processing systems like VISA, which can process thousands of transactions per second in a matter of seconds. For example, it takes roughly ten minutes for Bitcoin to complete seven transactions per second (TPS) [6]. More effective platforms, including Ethereum, Hyperledger Fabric and IOTA, were suggested after Bitcoin. Nevertheless, there are issues with their complexity, scalability, and performance that must be fixed. The purpose of this paper is to analysis and compare the scalability and performance of Ethereum, Hyperledger Fabric and IOTA for Internet of Things applications. Ethereum was selected because of its many features, which include DApps and smart contracts. It is also the second-most popular blockchain platform, therefore it has more resources and support than the majority of other platforms. Fabric was selected because it is the top private blockchain platform, offering capabilities like plug-and-play compatibility, modularity, and support for smart contracts. The inclusion of IOTA in the implementation for performance and scalability studies emerges from its distinct Tangle design, which permits indefinite scalability and concurrent transaction processing. It uses lightweight nodes that can run on minimum hardware, resulting in decreased resource consumption, and it includes fee-free transactions, making it perfect for microtransactions and Internet of Things applications. Incorporating IOTA offers a thorough analysis in contrast to established blockchain systems such as Ethereum and Hyperledger Fabric, emphasizing its possible benefits for scenarios requiring high throughput, low latency, and efficient resource usage.

The following is a summary of this article's primary contributions:

1) A comparison of the IOTA, Fabric and Ethereum networks conducted in the same controlled conditions using Caliper, a reliable benchmark tool.
2) An empirical approach for understanding these platforms' behavior under different workload scenarios.
3) Used performance metrics, including as throughput and latency, to assess platform scalability as transaction rates and numbers increased.
4) A use case and proof-of-concept implementation in the IoT healthcare field to highlight the challenges related to blockchain adoption in IoT applications.

The remainder of this article is structured as follows: An overview of the relevant work is given in Section II. The background information required to understand this study is presented in Section III. The design choices and decisions, as well as the IoT use case scenario in healthcare, are described in Section IV. In Section V, the experiments that were carried out are presented together with an implementation guide for the design decisions. Our experiment results are shown in Section VI along with an analysis of the outcomes. Section VII, which concludes, provides a summary of the work that was done and offers suggestions for further research.

II. RELATED WORK

A lot of research has been done on blockchain as an emerging technology over the past ten years. Numerous research publications, white papers, and community blogs are the outcome of these efforts. To fully understand this technology's usefulness, more research is still required on a few specific areas, most notably its scalability and performance. Yi Wang et al. [26] compared the performance and suitability of Ethereum and Hyperledger Fabric blockchain platforms for implementing Attribute-Based Access Control (ABAC) in smart home IoT environments. Chowdhury et al. [7] presented a comparative study of many popular DLT platforms, such as Ethereum, Fabric, and IOT. This research did not focus solely on blockchain platforms, in contrast to previous studies. Rather, it gathered DLT systems without consideration of the data structures on them. The analysis was divided into quantitative and qualitative categories by the authors. These requirements differ based on the kind of platform and include things like robustness and scalability. Even if there were enough platforms and criteria addressed in the comparison, the evaluation carried out in the study requires more in-depth analysis.

Numerous research examined the Ethereum or Fabric private networks' throughput and latency, or both. Kuzlu et al.'s analysis of Hyperledger Fabric's performance [8] took into account its scalability, latency, and throughput. The workloads of the Open and Query functions were taken into account when the authors measured these metrics using a Hyperledger Caliper. In contrast to Query, which only does one read per transaction, the Open function performs one write and one read. They also looked at the effects of transaction rates, transaction volume, and many transactions occurring at once. Their conclusions include the following: latency is particularly impacted by an increase in concurrent transactions, and transaction type has a significant impact on performance. But the experiment only took into account Open and Query workloads with low capacity.

In addition to the Hyperledger Fabric, the authors of [9] also took into consideration a private Ethereum implementation. They employed basic smart contracts for the analysis, which can issue, transfer, and establish accounts. The performance parameters were measured by the authors using a different methodology from that employed in [8]. Their results show that Fabric performs better than Ethereum throughout the board. The platforms' data collection methods vary, which could lead to an unfavorable assessment.

IOTA's feeless transactions, potential for greater TPS values, and lower energy usage led the authors of [10] to conclude that it is the best blockchain platform for the IoT sector. Nonetheless, they identified IOTA's primary scalability and decentralization constraints. Scalability was reexamined for IOTA smart contracts, which introduced new features with the introduction of IOTA 2.0. In comparisons that highlight the execution of smart contracts, IOTA was shown to be more scalable than Ethereum, the more well-liked alternative. Smart contracts are carried out in simultaneously by IOTA and the entire Ethereum network, respectively. While comparing several blockchain platforms in terms of performance and scalability, this study is devoid of tests and data. However, the authors of [11] examined IOTA for offline scalability techniques. The results highlight the need for a more scalable offline blockchain solution by exposing gaps in IOTA's offline transaction capabilities.

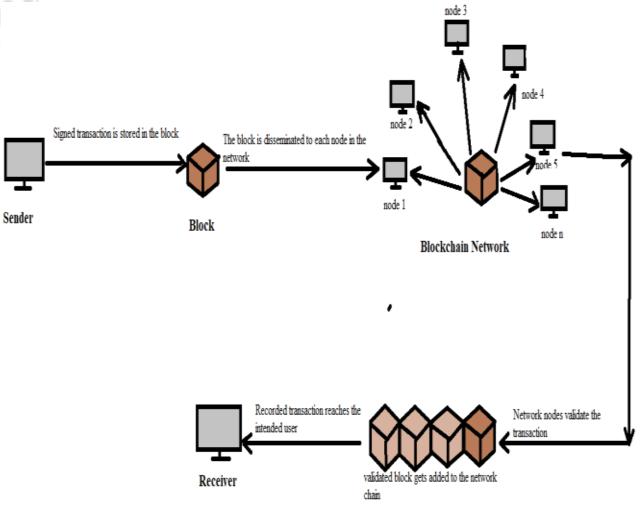


Figure 1. Blockchain working mechanism.

An analysis of previous research revealed that the majority of blockchain performance comparison tests failed to guarantee a consistent and stable environment on all platforms. The gap is addressed in this article as outlined in Section IV and Section V.

III. BLOCKCHAIN CONCEPT

The fundamental technology of Bitcoin, known as blockchain, was first suggested by an unidentified person named Nakamoto in 2008. Nakamoto created the concept of digital currency in decentralized peer-to-peer networks. The primary driving force was the desire to do rid of minimum transaction size restrictions and rising transaction costs for financial institutions.

Nakamoto's blockchain states that each transaction is recorded as a series of blocks. With cryptographic connections, the blocks are added to the chain in a linear, increasing fashion. Every block makes reference to the preceding block's hash. As a result, a hash chain is built that gives the ledger's data immutability. Distributed consensus algorithms also validate transactions as an extra layer of security.

Regarding its architectural benefits, blockchain generally offers several advantages. It is decentralized and reduces server expenses and bottlenecks in conventional central server architectures thanks to the distributed P2P network. The second benefit is that because transaction data is dispersed over the entire network and is validated, it is difficult to tamper with. Since each user's created address defines their identity, blockchain also offers anonymity. Finally, transactions may be tracked down and audited because of the validation procedure and timestamps utilized in keeping records [12].

However, because of its huge potential, the blockchain is being used for purposes other than cryptocurrency. Particularly when Buterin founded Ethereum in 2014 [13], blockchain technology became much more programmable and independent. For smart contracts to operate when a transaction fulfills a criterion specified in the contract, he suggested that they be incorporated into the system. As a result, Ethereum has developed into a platform for decentralized programs that employ smart contracts for logic.

Transactions are defined as the exchange of values between entities by Novo et al. [14]. A block is created by grouping the transactions collectively. Every transaction is broadcast to the network for validation after being signed by the owner. The signing and verification stages make use of digital signatures [15]. Specifically, during the signing stage, a transaction is encrypted using the owner's private key. Every node connected to the network can see the signed transaction. In the blockchain system, a certain node or group of nodes are referred to as miners. The miners verify the transaction by solving a cryptographic puzzle. A miner distributes the answer to other nodes after it has figured out the problem. The transaction is validated when it is confirmed by further nodes within the network. As a result, networks reach a consensus. By using the sender's public key to decrypt the data, the recipient can finally view the transaction's value and verify its integrity [15]. Fig. 1 provides a graphic representation of the blockchain's operational mechanism.

A. Ethereum

After Bitcoin, Ethereum is the second-largest blockchain in terms of cryptocurrency usage. Ethereum can do more than just work with currency, though. Ethereum was created with the goals of constructing decentralized apps, guaranteeing effective trade-offs between these applications, and offering security for small-scale applications, according to its white paper [13]. A Turing-complete programmable blockchain that allows anybody to create decentralized apps and smart contracts was presented as a way to put these concepts into practice. When together, Ethereum's three main characteristics make it stand out. These are Ethereum virtual machines (EVMs), smart contracts, and decentralized apps (DApps).

Smart contracts are blockchain-based programs that only run in response to predetermined criteria. The EVM, which is installed on each user's computer connected to the network, carries them out. Smart contracts are used by DApps as the application's logic. DApps offer a number of benefits over traditional application architectures, including as censorship resistance, transparency, and resilience [16].

B. Hyperledger Fabric

The Linux Foundation is the host organization for the open-source Hyperledger project. Its goal is to create blockchain technology suitable for enterprise use [17]. One of the Hyperledger subprojects is called Fabric. For a range of industrial applications, it offers modular distributed ledger technology [18]. Fabric stands on numerous important pillars that make it valuable on all blockchain systems. They are certificate authorities (CAs), modularity, chaincode, and permissioned blockchain.

Because Fabric is a permissioned blockchain, only approved organizations are allowed to join the network. Businesses may maintain their privacy, confidentiality, and robust scalability thanks to permissioned architecture. These characteristics might be preferred for enterprise applications as opposed to permissionless blockchains [18]. Chaincode is the term for Fabric's smart contracts. Standard programming languages can be used to write the distributed applications that Fabric runs [19]. Additionally, Fabric offers a plug-and-play consensus framework. Consensus techniques can be customized to meet specific use cases because to their versatility. Fabric needs some safe procedures for authorization and authentication because it is a permissioned blockchain. TLS certificates, enrollment certificates, and transaction certificates are the three types of certificates that increase Fabric's security. While transaction certificates are required for submission, enrollment certificates are utilized to establish a connection to the network [20].

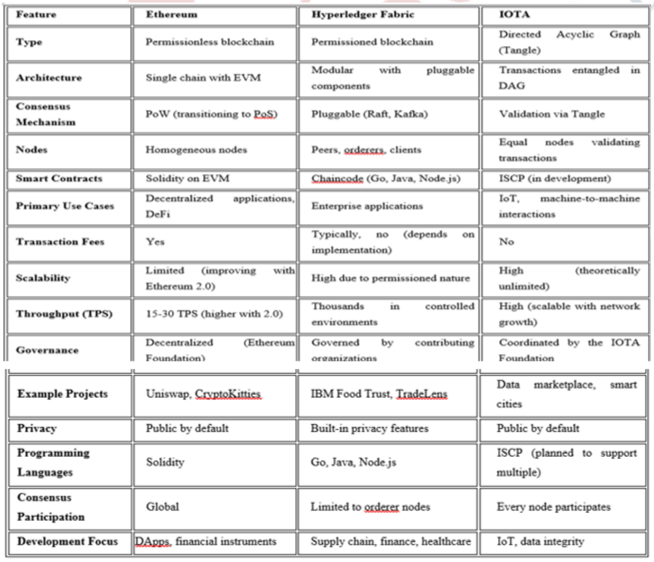
C. IOTA

IOTA is a blockchain technology that shows great promise and was created specifically for the Internet of Things (IoT). Its directed acyclic graph (DAG) distributed ledger, known as tangle, aims for high scalability and quick transaction confirmation. In comparison to a conventional single chain, the tangle greatly reduces blockchain speed issues; nevertheless, it also raises the possibility of double-spending attacks. By building illegitimate tangle branches to replace lawful ones, attackers within IOTA can perform double-spending attacks and jeopardize the security of the tangle [27].

One possible approach to overcome the shortcomings of conventional blockchain technology is IOTA's Tangle, a development of the IOTA Foundation [28]. The foundation of the Tangle is a Directed Acyclic Graph (DAG), a network of vertices connected by unidirectional edges that do not form loops, as opposed to a traditional blockchain, which is made up of blocks or chains. The genesis is represented by black in the picture, confirmed transactions are shown in green, unsure transactions (regarding their full acceptance) are shown in red, and tips (transactions still need validation) are shown in grey. IOTA's decentralized, distributed, immutable, and shareable digital ledger, which effectively stores transactions, depends on this concept. IOTA's main objective is to handle the enormous number of transactions that take place inside the vast network of linked IoT devices, which calls for a scalable ledger system [29]. By validating and approving two prior transactions before starting a new one, each peer in the Tangle helps to maintain the consistency of the network, doing away with the need for transaction fees and enabling quicker acceptance of new transactions.

D. Blockchain Parameters

Table 1. Comparision between Ethereum, Hyperledger Fabric, and IOTA



Feature	Ethereum	Hyperledger Fabric	IOTA
Type	Permissionless blockchain	Permissioned blockchain	Directed Acyclic Graph (Tangle)
Architecture	Single chain with EVM	Modular with pluggable components	Transactions entangled in DAG
Consensus Mechanism	PoW (transitioning to PoS)	Pluggable (Raft, Kafka)	Validation via Tangle
Nodes	Homogeneous nodes	Peers, orderers, clients	Equal nodes validating transactions
Smart Contracts	Solidity on EVM	Chaincode (Go, Java, Node.js)	ISCP (in development)
Primary Use Cases	Decentralized applications, DeFi	Enterprise applications	IoT, machine-to-machine interactions
Transaction Fees	Yes	Typically, no (depends on implementation)	No
Scalability	Limited (improving with Ethereum 2.0)	High due to permissioned nature	High (theoretically unlimited)
Throughput (TPS)	15-30 TPS (higher with 2.0)	Thousands in controlled environments	High (scalable with network growth)
Governance	Decentralized (Ethereum Foundation)	Governed by contributing organizations	Coordinated by the IOTA Foundation
Example Projects	Uniswap, CryptoKitties	IBM Food Trust, TradeLens	Data marketplace, smart cities
Privacy	Public by default	Built-in privacy features	Public by default
Programming Languages	Solidity	Go, Java, Node.js	ISCP (planned to support multiple)
Consensus Participation	Global	Limited to orderer nodes	Every node participates
Development Focus	DApps, financial instruments	Supply chain, finance, healthcare	IoT, data integrity

IV. DESIGN CHOICES

Section III-D listed all of the factors for analyzing the platforms in three categories, some of which were beyond the scope of this study. For a thorough examination and comparison of the selected platforms, specific parameters are the main focus. Thus, the parameters to be implemented are examined and presented in this subsection. The criteria have been selected with their application in mind. These parameters are used to test each platform. The platform's scalability and performance will be used to gauge its applicability. Performance and scalability are thought to be the two most important factors for a DLT platform. They have a direct bearing on whether a platform will be adopted in a given domain. A DLT platform's use is restricted by its scalability and low performance.

A. Technicality of the parameters

Based on the implementation parameters, the platform's performance is analyzed using two transaction metrics, throughput and latency. The number of transactions that the blockchain network can successfully complete in a second is known as throughput. When a transaction is committed into the ledger and included in a block, it is considered successfully processed. Conversely, latency refers to how long it takes a client to receive a response following the submission of a request. Since these two metrics are coupled to other parameters during the entire procedure, they form the central component of the experiment. The following are the parameters [21]:

$$L = t_c - t_s \qquad (1)$$

In Equation (1), L refers to transaction latency, where t_c is the confirmation time at the network threshold and t_s is the submit time.

$$T = n_c - t \qquad (2)$$

In Equation (2), Transaction throughput is denoted by T, where nc is the total number of committed transactions and t is the total duration in seconds. Transaction throughput and latency are monitored as objective performance metrics during the trials. An increase in the volume of transactions and transaction rates inside the network is a sign of scalability. Fig. 2 provides an instance of the relationship between the parameters:

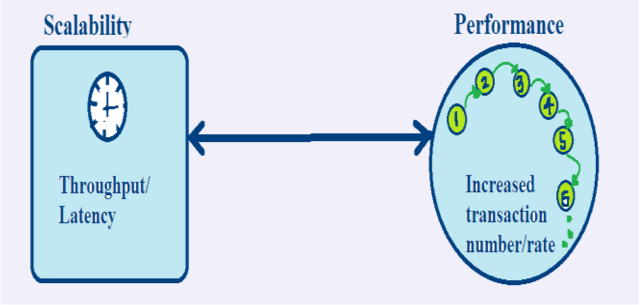


Figure 2. Relationship of the Parameters.

B. System Architecture Design

Common components could be used to characterize the overall architecture, even when each platform's system consists of multiple distinct components. The laptop utilized as the host in the present study has the following specifications:

Intel Core i5-1035G1 CPU @ 1.00GHz, 16GB RAM, 512GB SSD

The laptop serves as the host for the virtual machine (VM), as seen in Fig. 3. Because of the environment isolation it provides, it is possible to deploy the network in a virtual machine (VM), and resource utilization is essential for the success of the tests. To avoid any system problems, it is crucial to leave the host alone during the trials. You can also build instances and take screenshots of the computers when you use a virtual machine (VM). A rapid snapshot of the operational virtual machine could be reloaded in the event of a failure or the need for additional testing.

The software and tools required for the intended tests were installed in the virtual machine (VM). Every DLT platform is subjected to distinct experiments. Stated differently, Fabric, Ethereum, and IOTA, each employ a snapshot of the same virtual machine. The Ethereum, IOTA, and Fabric private networks are launched after the required software is installed and configurations are established. The performance and scalability experiments are carried out on these networks. A benchmarking tool is utilized to analyze and compare the platforms, as seen in Fig. 3. With different workloads, this program produces the throughput and latency results of the platforms under investigation.

Every platform is assessed in a nearby private setting as a summary of the methodology used. Since Ethereum is the most well-known public DLT platform, Fabric is already a private one. IOTA is primarily a public distributed ledger intended for open participation, but with considerable modification and extra privacy features, it can be made suitable for usage on private networks. To benchmark IOTA with Caliper, a specific adapter or plugin must be developed to transform Caliper's benchmarking tasks into operations compatible with IOTA's Tangle. This requires a lot of programming work to integrate with the distinct architecture of IOTA.

The precise needs and resources available for creating and maintaining the private network infrastructure would determine the viability and effectiveness of such an adaption.

Private networks, however, are more scalable and perform better than public ones. To assess them in this manner would be unfair. Since Ethereum also supports local private networks, private networks are the subject of the experiments. Their respective software needs, settings, and architectures vary. But for the sake of the study, a few criteria need to remain the same. Thus, it is necessary to apply the same workloads in a basic DLT platform use case. Quantitative data, which include direct experimental data and secondary data taken from the literature, are used to answer the research question. By changing specific factors in a controlled environment, the experimental data is produced.

The experiments begin once the test environment is configured. However, a testing tool is required in order to measure the parameters that were determined in Section IV-A. Currently, Huawei's official benchmarking tool is called Hyperledger Caliper. With the help of the blockchain performance benchmark framework Caliper, users may test various blockchain solutions for pre-established use cases and receive a set of test results. It now supports Ethereum and Fabric, among other blockchain systems. Performance metrics like throughput, latency, success rate, and resource usage are provided by Caliper. The official documentation for Caliper provides an example of its architecture. Workload modules, benchmark artifacts, benchmark and network configuration files are required as inputs. It then generates a report on the system that is being tested [22].

C. Use Case Scenario: IOT Healthcare Application

The Internet of Things is a new technology that offers potential in many important fields. But because of its nature, it is unable to meet certain requirements for interoperability, security, and traceability [23]. IoT can currently benefit from DLT platforms. The architecture covered in the earlier parts was created with IoT use cases in mind. The healthcare industry is one area where collaboration between IoT and DLT platforms is crucial. The data gathered from IoT devices, including wearables and biosensors, especially in remote patient monitoring, necessitates a secure transfer to a healthcare facility and group movement of the devices [24]. Since a patient's medical record contains information directly related to their health, it is quite sensitive.

For the data to be properly analyzed, it needs to be sent to the doctor who oversees and treats the patient. It should also be impossible for someone else to take hold of it and use it maliciously.

In this use case, a wearable sensor that measures a patient's blood pressure is given to him. Periodically, it is measured and sent to his doctor. The doctor intervenes if it rises above the set limits. In Fig. 4, we simulate the VM as an Internet of Things device that operates as a node in either a private Ethereum or Fabric network in order to tailor our design architecture to this use case. Then, via the DLT platform network, the generated data from the IoT device is moved from this node to another node—a doctor's computer. During implementation, this process is modeled after the "Transfer" rounds. A healthcare facility's private network can be the DLT platform network. An insurance firm would also be able to access the network. It is possible to store the patient records in the chain, making them unchangeable. This could be a method of providing accurate billing. The network's throughput and latency are assessed in the given settings, taking into account the scenario mentioned above. To find out which of Ethereum, IOTA, and Fabric is better suited for such an IoT application scenario, tests are conducted on their performance and scalability.

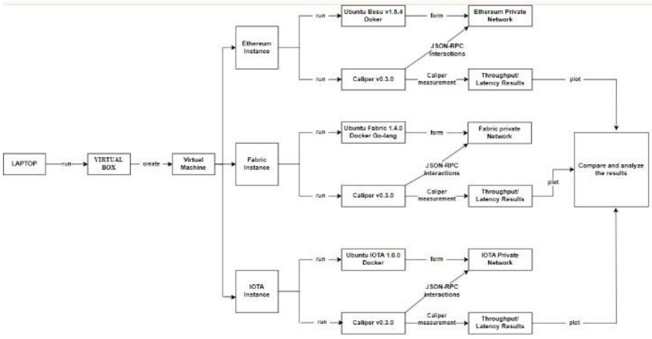


Figure 3. General Framework

V. IMPLEMENTATION AND EXPERIMENTS

This section describes how the previous section's design choices were implemented. The design explains how the work is carried out. Concise settings are required to set up the environment, tools, and software with the appropriate codes. The findings of the experiments are then presented for evaluation.

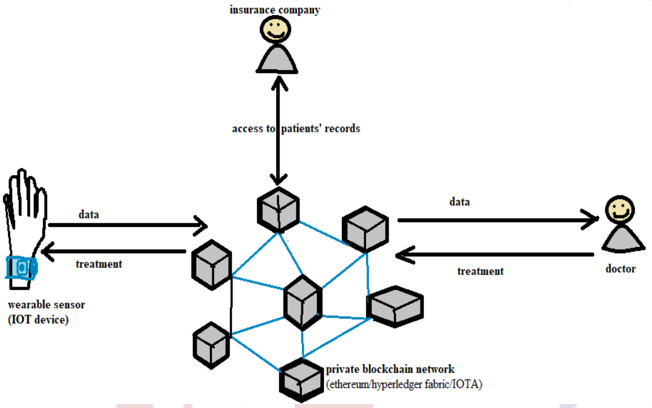


Figure 4. Healthcare IOT Use Case

To implement the design, follow these steps:

1) Set up environments for Ethereum and Hyperledger Fabric.
2) Set up Hyperledger Caliper, the benchmarking tool.
3) Integrate Hyperledger Caliper with the DLT platform network.
4) Write the code for the Hyperledger Caliper configuration files.
5) Transfer the files to Hyperledger Caliper and conduct the tests.
6) Present the test findings that were generated.

The experiments are conducted using Ubuntu 20.04 LTS in a virtual machine. This is a detailed description of a private DLT platform network setup and analysis because it is a challenging task. The platforms are handled individually since they have distinct requirements, binaries, and Docker images due to their nature. The primary software and tools required are Git, Curl, Docker Engine, Docker-compose, Golang, Node.js, npm, Python, and the Java Development Kit.

Binding Caliper to the platform's surroundings is the next step. Caliper requires the type and version of the platform's environment, defining it as a system under test (SUT). The SUTs utilized in this implementation are Hyperledger Fabric 1.4.0, Hyperledger Besu 1.5.4. and IOTA Chrysalis (IOTA 1.5). Consequently, preparing configuration files is the only step left before executing a benchmark. Two different file formats exist: benchconfig and networkconfig. They are necessary for Caliper to perform a benchmark and produce latency and throughput figures.

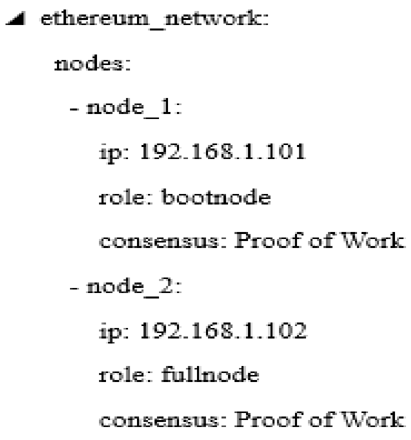
A. Benchmark Configuration

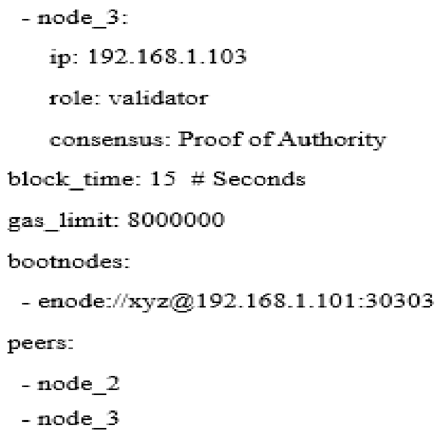
The benchmark configuration file, known as benchconfig, is in charge of carrying out the specified workload and gathering the data. There are three different kinds of settings in it: monitor, observer, and test. The test set allows for the definition of the transaction transmit rate, transaction count, and round type. Conversely, as their names suggest, monitor and observer settings deal with watching and observing. Regardless of the SUT type, the benchmark setup file remains the same [25]. Therefore, there is no risk in using the same file for Besu, Hyperledger Fabric and IOTA Chrysalis. Furthermore, having the same configuration for a healthy examination would be more advantageous.

A screenshot of the "config.yaml" section of the generated benchmark configuration file is written in network configuration section. The file contains three different types of rounds: open, inquiry, and transfer. For every type of round, transaction rates are set to 50 per second, while transaction numbers are set to 100 per second. The callback functions in this ".yaml" file refer to JavaScript files that contain the code for the round behaviors.

B. Network Configuration

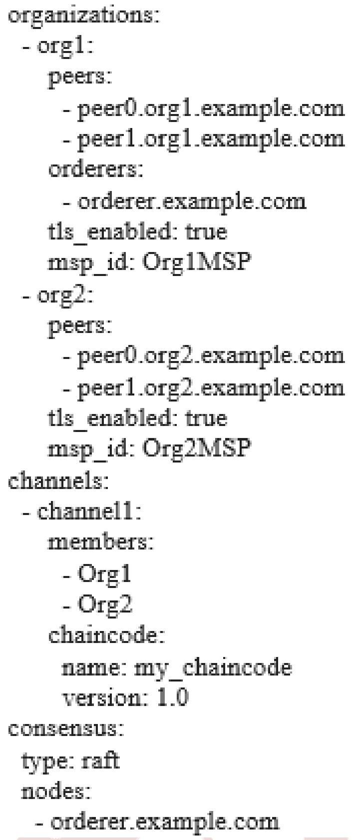
Unlike the benchmark configuration file, the network configuration file must be distinct for different DLT platforms. Therefore, particular files are created for Hyperledger Fabric, IOTA Chrysalis and Besu to be run by Caliper.


```
◢ ethereum_network:
    nodes:
     - node_1:
        ip: 192.168.1.101
        role: bootnode
        consensus: Proof of Work
     - node_2:
        ip: 192.168.1.102
        role: fullnode
        consensus: Proof of Work
```



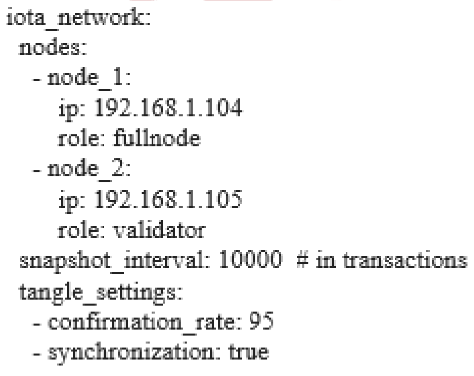
```
  - node_3:
    ip: 192.168.1.103
    role: validator
    consensus: Proof of Authority
block_time: 15  # Seconds
gas_limit: 8000000
bootnodes:
  - enode://xyz@192.168.1.101:30303
peers:
  - node_2
  - node_3
```

Figure 5. Ethereum (Besu) Network Configuration



```
organizations:
 - org1:
   peers:
    - peer0.org1.example.com
    - peer1.org1.example.com
   orderers:
    - orderer.example.com
   tls_enabled: true
   msp_id: Org1MSP
 - org2:
   peers:
    - peer0.org2.example.com
    - peer1.org2.example.com
   tls_enabled: true
   msp_id: Org2MSP
channels:
 - channel1:
   members:
    - Org1
    - Org2
   chaincode:
    name: my_chaincode
    version: 1.0
consensus:
 type: raft
 nodes:
  - orderer.example.com
```

Figure 6. Hyperledger Fabric Network Configuration



```
iota_network:
 nodes:
  - node_1:
    ip: 192.168.1.104
    role: fullnode
  - node_2:
    ip: 192.168.1.105
    role: validator
 snapshot_interval: 10000  # in transactions
 tangle_settings:
  - confirmation_rate: 95
  - synchronization: true
```

Figure 7. IOTA Network Configuration

VI. DESIGN CHOICES

Section III-D listed all of the factors for analyzing the platforms in three categories, some of which were beyond the scope of this study. For a thorough examination and comparison of the selected platforms, specific parameters are the main focus. Thus, the parameters to be implemented are examined and presented in this subsection. The criteria have been selected with their application in mind. These parameters are used to test each platform. The platform's scalability and performance will be used to gauge its applicability. Performance and scalability are thought to be the two most important factors for a DLT platform. They have a direct bearing on whether a platform will be adopted in a given domain. A DLT platform's use is restricted by its scalability and low performance.

A. Experiments

Caliper generates a report with throughput (tps) and latency(s) figures after the configuration files are passes to it and it is launched. These are the primary performance metrices but also show how scalable DLT platforms are. When the number of transactions in the network increases, the response of these indicators determines how well the platforms scale. Thus, during the scalability test, the benchmark configuration file is modified.

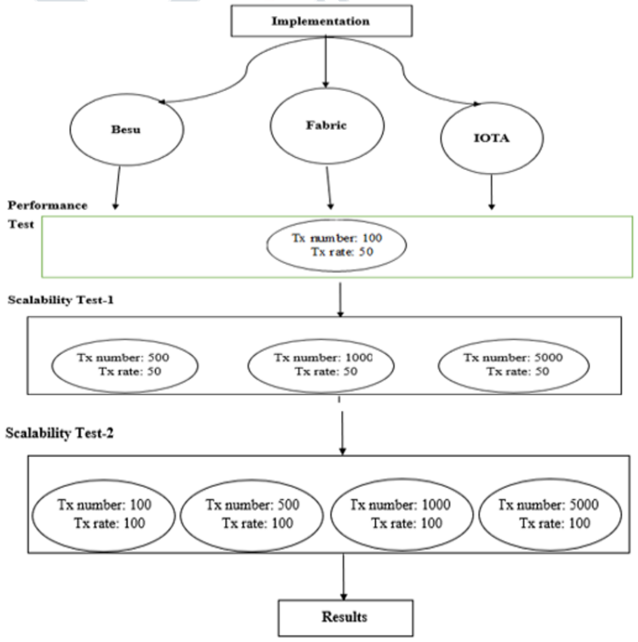


Figure 8. Implementation Diagram.

B. Performance Tests

The performance test is the first to be run. It is determined by considering the throughput and average latency values that Caliper produced for Fabric, Besu, and IOTA. Two rounds are used with the benchmark configuration file, i.e., open, and transfer. This indicates that if there were 100 transactions, the transaction rate would be 50. To get average numbers and eliminate instantly misleading values, these three sets of rounds are repeated ten times. A diagram of the implementation including the performance portion is shown in fig. 9.

C. Scalability Tests

For scalability measurements, the number of transactions and transaction rates in these rounds have increased. In turn, the transaction counts increase to 500, 1000, and 5000. To understand how the platforms scale, the aforementioned procedure is carried out once more using these transaction counts and 50 and 100 transaction rates. Figure 10 and 11 display the results of the scalability testing. The results are shown in the section that follows.

To understand how the platforms scale, the aforementioned procedure is carried out once more using these transaction counts and 50 and 100 transaction rates. Figure 7 and 8 display the results of the scalability testing. The results are shown in the section that follow.

VII. RESULTS AND DISCUSSION

The implementation of the recommended design choices was covered in the preceding section. The results are shown in this section based on the implementation. They are supported by related graphs. The results are examined and understood. This is how the analysis and comparison between Besu, Hyperledger Fabric, and IOTA are implemented. Secondary data from similar works are also used to evaluate the findings.

A. Performance Test Results

The graph contrasts the average latency and throughput for 100 transactions at an average rate of 50 transactions per second for Besu, Fabric, and IOTA. IOTA is the fastest in both open and transfer rounds since it continuously exhibits the lowest latency.

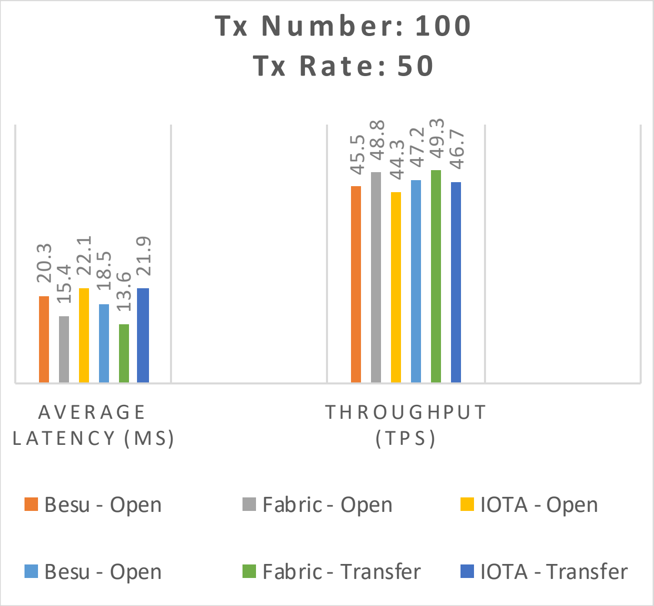


Figure 9. Performance Test Results

B. Scalability Test Results

The graph below shows the results of the scalability and performance tests conducted on Ethereum (Besu), Hyperledger Fabric, and IOTA based on different transaction numbers (100, 500, 1000, and 5000) with a fixed transaction rate of 50 in an open round. The results are displayed as average latency and throughput for each blockchain platform. The average latency increases as the number of transactions increases, with Besu typically maintaining the lowest latency, followed by IOTA and Fabric. On the other hand, throughput stays relatively stable across different transaction numbers for each platform, with Besu and IOTA showing slightly higher throughput than Fabric. The data shows that while latency increases with the transaction load, throughput stays constant, indicating the platforms' capacity to handle higher transaction volumes without experiencing a significant decline in performance.

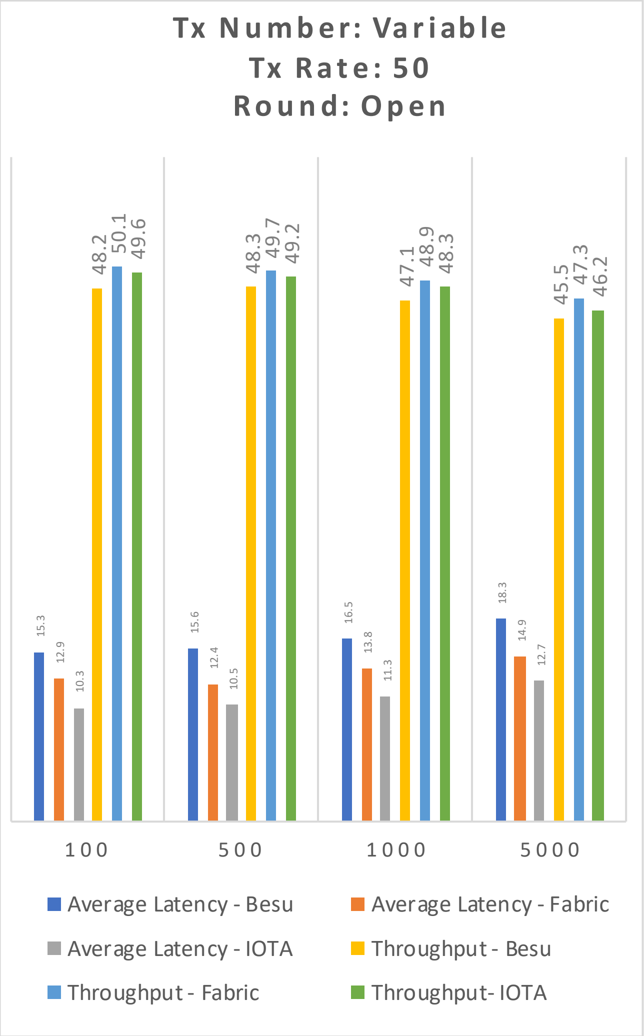


Figure 10. Scalability Test - Open - Tx rate: 50.

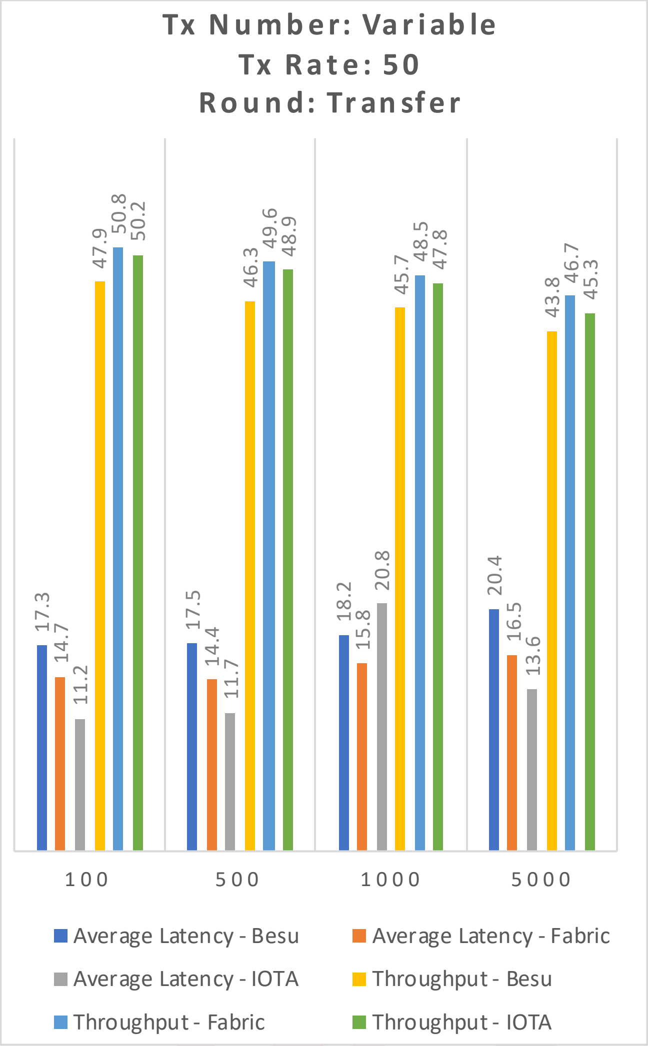


Figure 11. Scalability Test - Transfer - Tx rate: 50

The above graph shows the scalability and performance outcomes for Ethereum (Besu), Hyperledger Fabric, and IOTA over various transaction counts (100, 500, 1000, and 5000) for a transfer round with a fixed transaction rate of 50. The information displays each platform's average throughput and latency.

All platforms exhibit an increase in average latency as the volume of transactions increases, with Besu typically retaining lower latency than Fabric and IOTA. IOTA, on the other hand, has the largest latency, especially when the volume of transactions rises. Throughput remains relatively stable for each platform as transaction numbers increase with IOTA and Besu slightly outperforming Fabric in terms of throughput. The findings show that throughput stays constant while latency rises with transaction load, suggesting that all three platforms can manage higher transaction volumes without observing a noticeable reduction in performance.

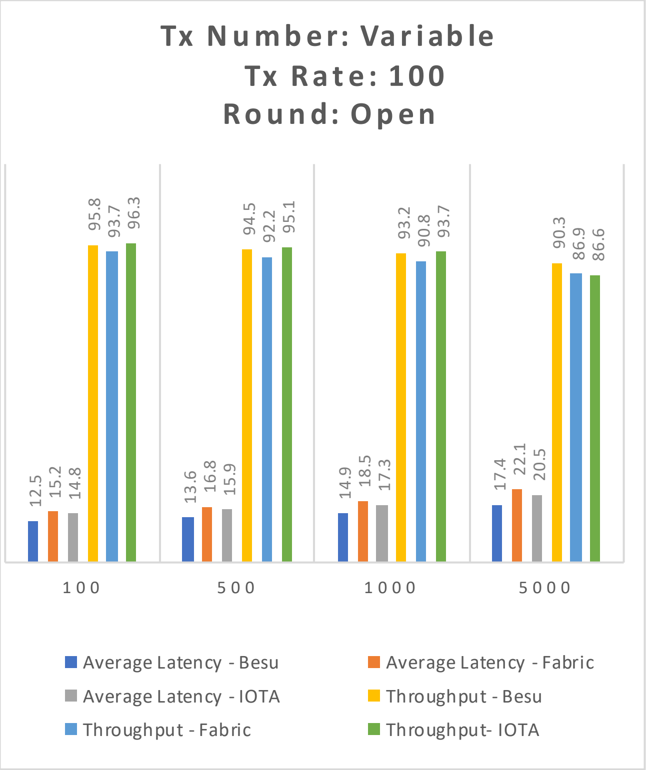


Figure 12. Scalability Test - Open - Tx rate: 50.

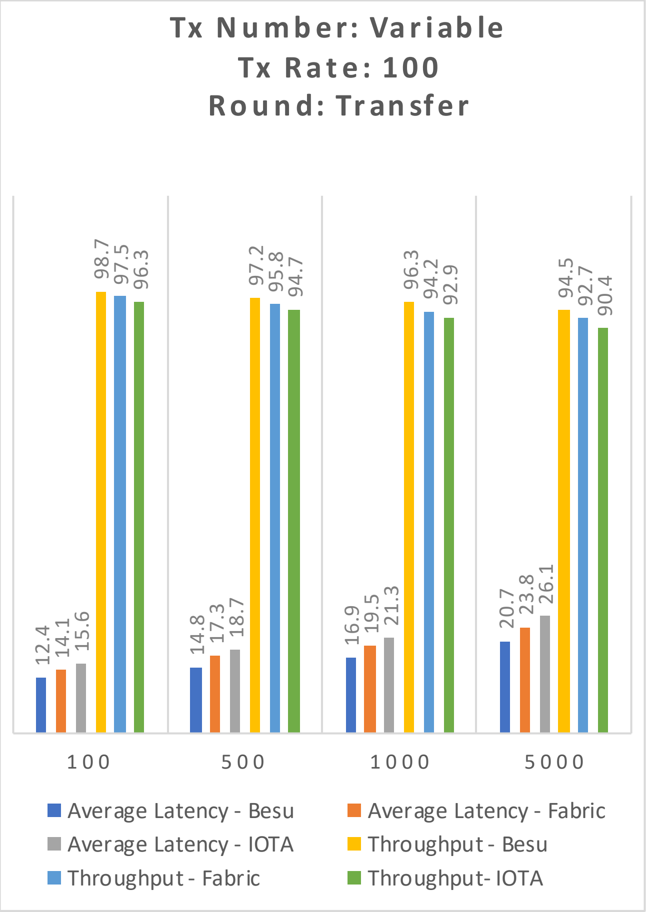


Figure 13. Scalability Test – Transfer – Tx rate: 100

The findings of the scalability test demonstrate that average latency increases noticeably on all three platforms- Ethereum (Besu), Hyperledger Fabric, and IOTA- as the number of transactions rises from 100 to 5000. The system with the highest latency is Hyperledger Fabric, whose latency increases from 15.2 ms at 100 transactions to 22.1 ms at 5000 transactions. Conversely, Ethereum (Besu) and IOTA continue to have comparatively reduced latencies, but they also see an increase as the volume of transactions increases. With throughput dropping from 95.8 tps and 96.3 tps at 100 transactions, respectively, Ethereum (Besu) and IOTA show better throughput stability. The throughput of Hyperledger Fabric, on the other hand, decreases more noticeably- from 93.7 tps at 100 transactions to 86.9 tps at 5000 transactions. In general, under higher transaction loads, Ethereum (Besu) and IOTA outperforms each other in terms of latency and throughput, whereas Hyperledger Fabric exhibits a more noticeable decline in performance.

Ethereum (Besu) consistently showed the lowest average latency, starting at 12.4 ms for transactions and gradually increasing to 20.7 ms from 5000 transactions in the scalability test results for Ethereum, Hyperledger Fabric, and IOTA with a fixed transaction rate of 100 and varying transaction numbers (100, 500, 1000, 5000) in a transfer round. As the volume of transactions increased, Besu also kept a high throughput, beginning at 98.7 TPS and gradually dropping to 94.5 TPS. Hyperledger Fabric demonstrated somewhat increased latency, starting at 14.1 ms for 100 transactions and increasing to 23.8 ms for 5000 transactions. In tandem, the throughput experienced a decrease from 97.5 TPS to 92.7 TPS. IOTA had the highest latency, ranging from 15.6 ms for 100 transactions to 26.1 ms for 5000 transactions. It also showed the lowest throughput, starting at 96.3 TPS and decreasing to 90.4 TPS as the volume of transactions rose. Based on the statistics, it appears that Besu, Fabric, and IOTA are the three platforms that can handle higher transaction quantities the most quickly. As transaction load increases, IOTA has a significant impact on latency and throughput. According to the experimental results, the key findings can be summarised as follows:

1) IOTA is best for applications requiring high throughput and low latency.
2) Hyperledger Fabric excels in permissioned environments where performance and scalability are critical.
3) Ethereum, while powerful and widely used, has scalability challenges but is working towards improvements with Ethereum 2.0.
4) Ethereum faces high latency and limited throughput (15-30 TPS) due to its proof-of-work mechanism but is working on scalability improvements with Ethereum 2.0.
5) Hyperledger fabric offers lower latency (1-2 seconds) and higher throughput (up to 3,000 TPS) in permissioned environments, with good scalability through modular architecture.
6) IOTA provides the lowest latency and high throughput with its Tangle architecture, enabling scalable performance without traditional mining or block constraints.
7) Hyperledger Fabric provides best throughput leads in transaction processing speed within the permissioned networks.

VIII. CONCLUSION AND FUTURE WORK

This study evaluated and compared the scalability and performance features of Ethereum and Hyperledger Fabric, two well-known blockchain platforms. It primarily analyzed the throughput and latency parameters of private Ethereum, Fabric, and IOTA networks.

The tool used to generate these characteristics was Hyperledger Caliper, which was integrated with the private Ethereum, Fabric, and IOTA networks. Scalability tests were carried out by examining the transaction number and rate increase, as these metrics were direct indicators of performance. In particular, fixed 100-transaction and 50-transaction rates are used for all the three networks for the performance tests. However, two different scalability test kinds were conducted. The first one maintained the transaction rate at 50 while increasing the transaction number in the order of 500, 1000, and 5000. Each test was run ten times to acquire average values; this prevented instant peaks in the results. In the second scalability test, the transaction rate was set to 100 and the transaction numbers were raised in the order of 100, 500, 1000, and 5000.

Based on the empirical results, IOTA's Tangle architecture typically results in the lowest latency. Hyperledger Fabric leads in throughput in permissioned environments, but IOTA also demonstrates strong performance in terms of scalability and throughput. IOTA is best suited for high scalability scenarios because of its DAG-based architecture, which makes it highly efficient as the network grows. Hyperledger Fabric is scalable in permissioned environments but has limitations in fully decentralized scenarios. Ethereum is currently dealing with significant scalability issues, but is actively working on solutions through Ethereum 2.0 to improve its performance and scalability. The organisations should determine their needs and priorities while choosing the DLT platform. Because average latency and throughput may be critical to the patient's health in the IoT healthcare scenario in Section IV-C, Fabric would be a better DLT platform.

The key limitations of the work are the number of tests conducted and the execution of experiments within the same local network. The tests may be run many more time sin a more reliable distributed environment to improve this study and broaden its scope and ensure better results. In addition to Ethereum, Fabric, and IOTA, we intend to incorporate other blockchain platforms in future studies because of their promising scalability feature. We also intend to evaluate them in terms of scalability, performance, and security. A more thorough understanding of the advantages and disadvantages of certain blockchain technologies will result from this deeper analysis.

REFERENCES

[1] S. Underwood, ‘‘Blockchain beyond Bitcoin,’’ *Commun. ACM*, vol. 59, no. 11, pp. 15–17, Oct. 2016.

[2] M. Swan, *Blockchain: Blueprint for a New Economy*

[3] Sebastopol, CA, USA: O’reilly, 2015.

[4] S. M. Alshurafa, D. Eleyan, and A. Eleyan, ‘‘A survey paper on blockchain as a service platforms,’’ *Int. J. High Perform. Comput. Netw.*, vol. 17, no. 1, p. 8, 2021.

[5] D. Unal, M. Hammoudeh, M. A. Khan, A. Abuarqoub, G. Epiphaniou, and R. Hamila, ‘‘Integration of federated machine learning and blockchain for the provision of secure big data analytics for Internet of Things,’’ *Comput. Secur.*, vol. 109, Oct. 2021, Art. no. 102393, doi: 10.1016/j.cose.2021.102393.

[6] D. Unal, M. Hammoudeh, M. A. Khan, A. Abuarqoub, G. Epiphaniou, and R. Hamila, ‘‘Integration of federated machine learning and blockchain for the provision of secure big data analytics for Internet of Things,’’ *Comput. Secur.*, vol. 109, Oct. 2021, Art. no. 102393, doi: 10.1016/j.cose.2021.102393.

[7] J. Yli-Huumo, D. Ko, S. Choi, S. Park, and K. Smolander, ‘‘Where is current research on blockchain technology? —A systematic review,’’ *PLoS ONE*, vol. 11, no. 10, Oct. 2016, Art. no. e0163477.

[8] M. J. M. Chowdhury, MD. S. Ferdous, K. Biswas, N. Chowdhury, A. S. M. Kayes, M. Alazab, and P. Watters, ‘‘A comparative analysis of distributed ledger technology platforms,’’ *IEEE Access*, vol. 7, pp. 167930–167943, 2019.

[9] M. Kuzlu, M. Pipattanasomporn, L. Gurses, and S. Rahman, ‘‘Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability,’’ in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 536–540.

[10] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, ‘‘Performance analysis of private blockchain platforms in varying workloads,’’ in *Proc. 26tt Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2017, pp. 1–6.

[11] N. Sealey, A. Aijaz, and B. Holden, ‘‘IOTA tangle 2.0: Toward a scalable, decentralized, smart, and autonomous IoT ecosystem,’’ in *Proc. Int. Conf. Smart Appl., Commun. Netw. (SmartNets)*, Nov. 2022, pp. 01–08.

[12] Rawat, V. Daza, and M. Signorini, ‘‘Offline scaling of IoT devices in IOTA blockchain,’’ *Sensors*, vol. 22, no. 4, p. 1411, Feb. 2022.

[13] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, H. Wang, A. S. M. Kayes, M. Alazab, and P. Watters, ‘‘Blockchain challenges and opportunities: A survey,’’ *Int. J. Web Grid Services*, vol. 14, p. 352, Jan. 2018.

[14] V. Buterin. (2014). *A Next-Generation Smart Contract and Decentralized Application Platform*. [Online]. Available: https://ethereum.org/en/whitepaper/

[15] O. Novo, ‘‘Blockchain meets IoT: An architecture for scalable access management in IoT,’’ *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1184–1195, Apr. 2018.

[16] S. Xie, H. Dai, X. Chen, and H. Wang, ‘‘An overview of blockchain technology: Architecture, consensus, and future trends,’’ in *Proc. IEEE Int. Congr. Big Data*, Oct. 2017, pp. 557–564.

[17] Antonopoulos and G. Wood, *Mastering Ethereum: Building Smart Contracts and DApps*. Sebastopol, CA, USA: O’Reilly Media, 2019.

[18] (2020). *Hyperledger—Open Source Blockchain Technologies*. Accessed: May. 13, 2024. [Online]. Available: https://www.hyperledger.org/

[19] Hyperledger Fabric. *Hyperledger—Open Source Blockchain Technologies*. Accessed: May. 13, 2024. [Online]. Available: https://www.hyperledger.org/wpcontent/uploads/2020/03/hyperledger_fabric_whitepaper.pdf

[20] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, and D. Enyeart, ‘‘Hyperledger fabric: A distributed operating system for permissioned blockchains,’’ in *Proc. 13th EuroSys Conf.*, 2018, pp. 1–15.

[21] Cachin. (2016). *Architecture of the Hyperledger Blockchain Fabric*. Accessed: May. 13, 2024. [Online]. Available: https://pdfs.semanticscholar.org/f852/c5f3fe649f8a17ded391df0796677a59927f.pdf?_ga=2.153008701.585498121.1597310266-259713143.1596530036

[22] (2018). *Hyperledger Blockchain Performance Metrics*. Accessed: May. 17, 2024. [Online]. Available: https://www.hyperledger.org/wpcontent/uploads/2018/10/HL_Whitepaper_Metrics_PDF_V1.01.pdf

[23] Hellani, H.; Sliman, L.; Samhat, A.E.; Exposito, E. Computing Resource Allocation Scheme for DAG-Based IOTA Nodes. Sensors **2021**, 21, 4703.

[24] H. Caliper. *Architecture*. Accessed: May. 27, 2024. [Online]. Available: https://hyperledger.github.io/caliper/v0.4.0/architecture/

[25] M. Hammoudeh and R. Newman, ‘‘Information extraction from sensor networks using the watershed transform algorithm,’’ *Inf. Fusion*, vol. 22, pp. 39–49, Mar. 2015.

[26] M. Hammoudeh, G. Epiphaniou, S. Belguith, D. Unal, B. Adebisi, T. Baker, A. S. M. Kayes, and P. Watters, ‘‘A service-oriented approach for sensing in the Internet of Things: Intelligent transportation systems and privacy use cases,’’ *IEEE Sensors J.*, vol. 21, no. 14, pp. 15753–15761, Jul. 2021.

[27] H. caliper. Installing and Running Caliper. Accessed: June 7, 2024. [Online]. Available: https://hyperledger.github.io/caliper/vNext/installing-caliper/

[28] J. Liu, X. Li, and C. Zhang, "A Survey on the Evolution of Wireless Sensor Networks," *Sensors*, vol. 23, no. 16, pp. 7046, Aug. 2023, doi: 10.3390/s23167046.

[29] Y. Chen, Y. Guo, M. Wang, E. Xu, H. Xie and R. Bie, "Securing IOTA Blockchain Against Tangle Vulnerability by Using Large Deviation Theory," in *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 1952-1965, 15 Jan.15, 2024, doi: 10.1109/JIOT.2023.3283788.

[30] Zhang, Y.; Nakanishi, R.; Sasabe, M.; Kasahara, S. Combining IOTA and Attribute-Based Encryption for Access Control in the Internet of Things. Sensors 2021, 21, 5053.

[31] Hellani, H.; Sliman, L.; Samhat, A.E.; Exposito, E. Computing Resource Allocation Scheme for DAG-Based IOTA Nodes. Sensors 2021, 21, 4703.