Blockchain : Comparative Analysis of Public and Private networks

Omkar Ghag[1], Prof. Vidya Chitre[2]
[1]Mumbai University, India,
[2]Mumbai University, India.

Abstract—

Blockchain technologies have grown swiftly in recent years, primarily due to the advent of Bitcoin. Blockchain is a consensus of data structures or blocks programmed for cryptographically storing organized data, which is spread across nodes, so that various operations can be executed on it. Originally blockchain was designed for storing digital coins as system states and now has grown beyond crypto-currencies to support user-defined decentralized autonomous applications or smart contracts with Ethereum. Growing interest from the industry and wide range of applications has triggered development of new blockchain platforms. As the technology is progressing quickly, it is necessary and challenging to have a more profound perspective of what the core technology platforms have to provide in order to establish which blockchain implementation should be leveraged for a particular application. We therefore grasp the key aspects of blockchain and compare how the current blockchain implementations are different from each other, both qualitatively and quantitatively in terms of design architecture, codebase, consensus algorithms and performance. Drawing from the comparison we identify the current challenges as well as performance issues in blockchain adoption, thus suggesting possible solutions and future implementations to improve performance of blockchain and provide subsequent research directions.

Keywords— Blockchain, Consensus Protocol, Distributed ledger, Decentralized

I. Introduction

Blockchain technologies are increasingly becoming popular largely due to the success of Bitcoin and also because they offer a new tool for solving problems of which people were unaware before. Blockchain is a distributed database that is widely used for recording distinct transactions. Once a consensus is reached among different nodes, the transaction is added to a block that already holds records of all previous transactions. Each block contains the hash value of its last counterpart for connection and together they form a blockchain. Data is spread across and maintained by various nodes (the distributed data storage) and is thus decentralized. Consequently, a block becomes validated only once it has been verified by multiple parties (nodes) in the chain. [1] [2]

As a result of these unique features, in terms of database, blockchain can be seen as a solution to distributed transaction management in which nodes keep replicas of the data and execution of transactions only takes place on mutual agreement making it immutable, transparent and byzantine fault tolerant by design which is more secure than current database systems like MySQL and Oracle, thus reducing human error and the need for manual intervention in case of conflicting data. [1] [3]

From the original design of storing digital coins as system states blockchain has developed beyond crypto-currencies to support user-defined, decentralized and replicated applications known as smart contracts through Ethereum[10]. Interest from the industry has triggered the development of new blockchain platforms designed for private settings where participants are authenticated by the system called private (or permissioned) blockchain such as Hyperledger Fabric[11], as opposed to the early blockchain systems operating in public environments (or permissionless) where anyone can join and leave. [1] [3] J.P. Morgan has predicted that blockchains will soon replace currently redundant infrastructure by 2020. [4]

As a result, of growing commercial and academic interest, a large number of blockchain systems, each with some unique capabilities have developed. Some of them are still in development (research phase), while others are currently running. Both private and public sector are demanding to adopt blockchains, but they face variety of choices with different implementations available which vary in many ways such as in terms of purpose, consensus, method of participation (permissioned or permissionless), how governance is handled, security and much more. In order to determine which blockchain implementation should be leveraged for a given application it is important to be familiar with the differences between each implementation. [1] [3]

To achieve this, we start by understanding the origin and evolution of blockchain technology, structure and formation of a block along with the mining process involved. We distinguish major classes of blockchain systems, namely public, private and consortium blockchains and explain four key technical concepts by which current systems can be categorized: distributed ledger, cryptography, consensus protocol and smart contract. We then analyze, compare and conduct a comprehensive evaluation of four major blockchain implementations: Bitcoin, Ethereum, Multichain and Hyperledger both qualitatively and quantitatively in terms of design and special emphasis on performance based on transaction rate, size of block and transaction, block release time and scalability. [1] Drawing from the comparison we identify and discuss the current challenges as well as performance issues (bottlenecks) in blockchain adoption which can serve as potential research direction and suggest possible solutions and future implementations to overcome those challenges and improve performance of blockchains.

II. Literature survey

2.1 Origin of Blockchain

The first effort on chain of (records) blocks securely linked using cryptography was described in 1991 by Stuart Haber and W. Scott Stornetta. [5] They proposed a practical system for time-stamping of digital documents in order to maintain complete privacy of the documents and require no record keeping by the time stamping service which was achieved by using cryptographically secure hash functions and digital signatures for solving the issues of privacy, bandwidth, storage, incompetence and trust. [6] In 1992 Bayer, Haber and Stornetta incorporated Merkle Trees into the design, which greatly improved efficiency by allowing several documents to be added into one block. [5] Nakomoto et al. [7] conceptualized the idea of Bitcoin and developed a practical system for transferring money among entities connected in a peer to peer manner. Based on this idea, several distributed computing platforms for blockchains such as Ethereum [10] and Hyperledger [11] used to power bitcoins were developed. [8]

2.2 Evolution of Blockchain

Blockchain technology has been evolving rapidly and is classified into following generations:

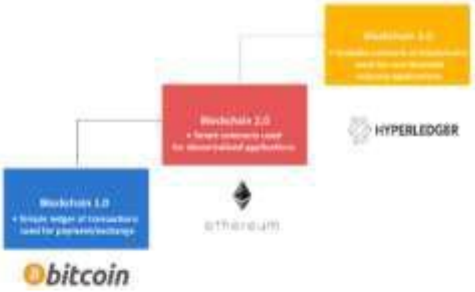
Blockchain 3.0
Blockchain 2.0
• Smart contracts used for decentralized applications
Blockchain 1.0
• Simple ledger of transactions used for payment/exchange
bitcoin
ethereum
HYPERLEDGER

Figure 1: Evolution of Blockchain

2.2.1 First Generation Blockchain

The first generation of blockchain evolution involves applications like Bitcoin started by Satoshi Nakamoto in 2009, which is a pure peer-to-peer version of electronic cash (digital asset) where online payments can be sent directly from one party to another without the participation of a centralized financial authority. [7] The network is secured by cryptographic algorithms and verified by a decentralized global network and recorded into an unchangeable public ledger. [9]

2.2.2 Second Generation Blockchain

Ethereum went live for the public on 30 July, 2015 as a second generation blockchain with a built-in Turing complete programming language, which allowed developers to write smart contracts and decentralized, shared and self executing applications (DAPP) where they can create their own customized rules for ownership, transaction formats and state transition functions. [9]

Through enabling ways to write distributed trust free application easily on top of inherently secured Ethereum further demonstrated and validated, the technology's ability to evolve beyond just a means of transferring value (digital assets). [9] [2]

2.2.3 Third Generation Blockchain

The first and second generation of blockchain have unresolved scaling issues due to limitations in terms of transaction throughput and speed. Thus, a blockchain generation enabled with hybrid networks where businesses would be able to benefit from the scalability, security and privacy of a private blockchain while still maintaining the ability to prove data integrity and achieve overall consensus through cross-chains is being developed. Networks that can be deployed on a multitier basis, while the thousands and millions of blockchain networks are able to communicate with each other swiftly. A network supported by improved virtual machine which is able to process transaction and information in a more trusted environment. Currently platforms such as Hyperledger, Iota - Tangle are being created as an infrastructure supporting these third-generation features that require blockchain technology to move in mainstream adoption. [9]

2.3 Blockchain Structure

A typical blockchain system consists of multiple nodes which do not fully trust each other. It is a special data structure which stores historical states and transactions. In blockchain each block is linked to its predecessor via a cryptographic hash of previous block, all the way back to the first (genesis) block. A transaction in a blockchain is a sequence of operations applied on some states. [1]

Each block is made of header containing metadata such as its:

1. **Previous block hash**
2. **Nonce**
3. **Merkle Tree that stores transactions**
4. **Followed by a list of transactions.** [7] [1]

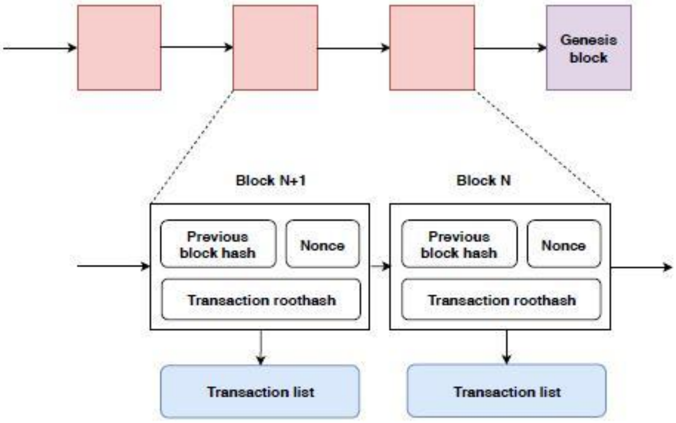

Figure 2: Typical blockchain data structure

2.4 Merkle Tree

Merkle Tree is a data structure of transaction hashes created by repeatedly hashing pairs of nodes until there is only one hash left called the **Root Hash or (Merkle Root)** [5]. After the latest transaction is buried under enough blocks in the chain, the spent transactions before it can be discarded to save disk space. In order to achieve this without breaking the block's hash, transactions are hashed in a Merkle Tree, with only the root included in the block's hash. Old blocks can then be compacted by stubbing off branches of the tree. The interior hashes do not need to be stored which saves disk space and simplifies verification of transactions. [7]

2.5 Block Formation

Each node broadcasts a set of transactions it wants to perform. Following announcement of new block candidate, nodes within the network called miners verify candidate and validate transaction details and create a record of the transaction by adding it to a block. Each node works on finding a difficult proof-of-work (consensus protocol) for its block. If node consensus is achieved, the new block is added to the chain and the miner broadcasts the block candidate to the blockchain network. Miners stop work on last block, return open transactions to pool, and begin work on next block. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash. [1] [5] [7]

Nodes always consider the longest chain to be the correct one and will keep working on extending it. If two nodes broadcast different versions of the next block simultaneously, some nodes may receive one or the other first. In that case, nodes work on the first one they received, but save the other branch in case it becomes longer. [7]

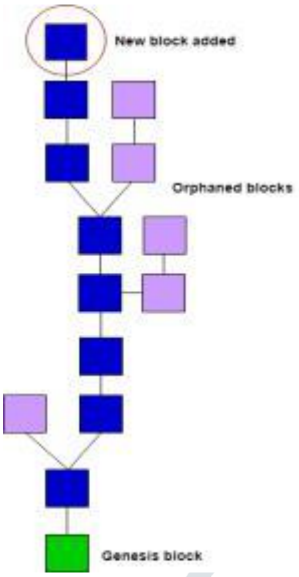

Figure 5: Adding a block to the chain

2.6 Mining Process

The idea of mining is based on block rewards for person or group who finds solution (nonce) to the cryptographic hashing algorithm. The solution is a mathematical calculation that uses results of the previous block hash. [5] [7] The mining process is as follows [5]:

1. Hash of previous block (pointer) is combined with current input data and hashed
2. Nonce added to that and hashed.
3. Result compared to required difficulty level.
4. If criteria met, new block candidate announced to nodes
5. If criteria not met, nonce updated and rehashed and the process is repeated.

A nonce is a concatenation of numbers used once in previous block hash. In case of Bitcoin this number is between the range of 0 to 4,284,967.296. [7]

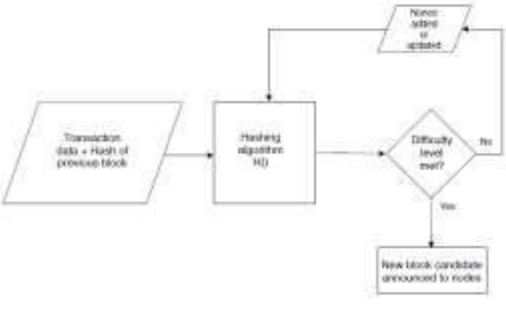
Nonce added or updated
Transaction data + Hash of previous block
Hashing algorithm H()
Difficulty level met?
No
Yes
New block candidate announced to nodes

Figure 6: Mining Process [5]

2.7 Classification of Blockchain

A blockchain system can be broadly categorized as either public, private and consortium.

2.7.1 Public (or Permissionless) Blockchain

In public blockchain any node can join and leave the blockchain is fully decentralized peer-to-peer system [1]. Example: Bitcoin [7]

2.7.1.1 Features:

1. **Authority:** No governing central authority, its decentralized system. [5]
2. **Transparency:** Data can be viewed publicly [7]
3. **Customizability and Control:** Decentralized blockchains are not customizable and offer less control over the network as any node can join and leave the system. [5]
4. **Performance:** Transaction throughput and speed is low which leads to performance issues. [1] [8]
5. **Consensus Protocol:** Most public blockchain systems employ variants of Proof of Work (PoW) for consensus as it is resistant to Sybil attack. [7] [1]
6. **Computationally:** Expensive [7] [1]

2.7.2 Private (or Permissioned) Blockchain

In Private, the blockchain enforces strict membership, there is access control mechanism to determine who can join the system. As a result, every node is authenticated, and its identity is known to the other nodes. [1] Example: IBM's Hyperledger Fabric [11]

2.7.2.1 Features:

1. **Authority**: Central authority, administration layer. [5]
2. **Transparency:** Data is private [5]
3. **Customizability and Control:** Centralized blockchains offer much more customizability and control over the network as every node is authenticated by the organization deploying the blockchain network. [1]
4. **Performance:** There are less transactions, resulting in faster throughput and speed which leads to better performance. [5]
5. **Consensus Protocol:** Since node identities are known in the private settings, most blockchains adopt one of the protocols from the vast literature on distributed consensus. [1]
6. **Computationally:** Cheaper

2.7.3 Consortium Blockchain

Consortium blockchain is partially private. It provides a combination between the low trust provided by public blockchains and the highly-trusted single entity model of private blockchains. Instead of allowing any person with an internet connection to participate in the verification of transactions process or allowing only one entity or authority to have full control, a few selected nodes are predetermined and allowed access. [12] A consortium platform provides many of the same benefits associated with private blockchains(efficiency and transaction privacy), without restricting power with only one company but operates under the leadership of a group instead of a single entity who decide access to the blockchain ledger. This platform is best suited for organizational collaboration. [12]

III. KEY ASPECTS OF BLOCKCHAIN SYSTEMS

Classifying blockchains as public, private or consortium is useful for identifying crucial characteristics of many blockchains. However, understanding their minute differences requires a system, thus the more, finer categorization. This section introduces four cornerstones, based on which a more detailed classification of the systems can be obtained.

3.1 Distributed Ledger

A ledger is a data structure that consists of an ordered list of transactions which for example may record monetary transactions between multiple banks, or goods exchanged among known parties. In blockchains, the ledger is replicated over all the nodes and transactions are grouped into blocks which are then chained together. Thus, distributed ledger is essentially a replicated append-only data structure which starts with some initial states and records entire history of updated operations made to the states. The application built on top of the ledger determines the data model of what being stored in the ledger. The data model captures key data abstractions, making it easy for the application to express its logic. A system supporting distributed ledgers is characterized by its target applications, by the number of ledgers, and by the ledger ownership. [1]

3.2 Cryptography

Blockchain systems make heavy use of cryptographic techniques to detect tampering of the blockchain data stored in the ledger. This feature is crucial in public blockchain settings where there is no pre-established trust. For example, public confidence in crypto-currencies like Bitcoin, is predicated upon the integrity of the ledger; that is the ledger must be able to detect double spending. Even in private blockchains, integrity is equally essential because the authenticated nodes can still act maliciously. [1] [7]

There are at least two levels of integrity protection. First, the global states are protected by a hash (Merkle) tree whose root hash is stored in a block. Second, the block history is protected, that is the blocks are immutable once they are appended to the blockchain. The key technique is to link the blocks through a chain of cryptographic hash pointers: the content of block number n + 1 contains the hash of block number n. This way, any modification in block n immediately invalidates all the subsequent blocks. Blockchain implementations such as Bitcoin and Ethereum uses Elliptic curve Digital Signature Algorithm (ECDSA) to generate private and public key pairs due to reduced key size and hence speed. [1] [7] [8]

A user in a blockchain is uniquely identified by her public key certificate. In public settings, the user first generates a key pair (the default option being ECDSA based on the Secp256k1 elliptic curve), then derives the identity as the hash of the public key. [1]

In private settings, there is an additional access control layer. Hyperledger separates this layer from the blockchain, in the form of a membership provider service and a certificate authority service in which content confidentiality is achieved by encrypting the transactions (in part or in total) using common cryptographic algorithms such as AES (Advanced encryption standard) sending transactions to ordering service and appending blocks to the ledger. [13] [14]

3.3 Consensus

The data stored in the ledger reflects historical and current states maintained by the blockchain. Before being replicated, updates to the ledger must be agreed on by all parties. In other words, multiple parties must come to a consensus (agreement) which is in the best interest of the whole system which is achieved with the help of consensus algorithm or consensus protocol. [1] [7] The consensus algorithms are designed to achieve reliability or trust in network that involves multiple unreliable nodes as in blockchain may behave in Byzantine manners. The consensus protocol must therefore tolerate Byzantine failures. [1]

Various consensus protocols are being developed for blockchains which are widely classified as computation-based protocols that use proof of computation to randomly select a node which single-handedly decides the next operation. Bitcoin’s proof-of-work (PoW) is an example. [1] [8] Stake based protocol which uses the amount of stake (digital assets) a validator has and the respective age of the stake. Example, Ethereum’s upcoming PoS (Proof-of-stake) protocol is implemented as a smart contract. Referred to as Casper, it allows miners to become validators by depositing Ethers to the Casper account. The contract then picks a validator to propose the next block according to the deposit amount. [1]

The other is purely communication-based protocols in which nodes have equal votes and go through multiple rounds of communication to reach consensus. Example Practical Byzantine Fault Tolerance Algorithm PBFT which used in private settings (Hyperledger), as they assume authenticated nodes. [1] [3]

3.4 Smart Contract

Smart contract refers to the computation executed when a transaction is performed which can be regarded as a stored procedure invoked upon a transaction. The inputs, outputs and states affected by the smart contract execution are agreed on by every node. User-defined smart contracts can be created using the Ethereum blockchain. [1] [2] Developers are able, according to their needs, to specify any instruction in smart contracts; develop various types of decentralized applications, including those that interact with other contracts; store data; and transfer digital assets. Additionally, smart contracts that are deployed in blockchains are copied to each node to prevent contract tampering. With related operations executed by computers and services provided by Ethereum, human error can be reduced to avoid disputes regarding such contracts. Smart contracts are mostly used in voting system and cryptocurrency applications. [2] [16]

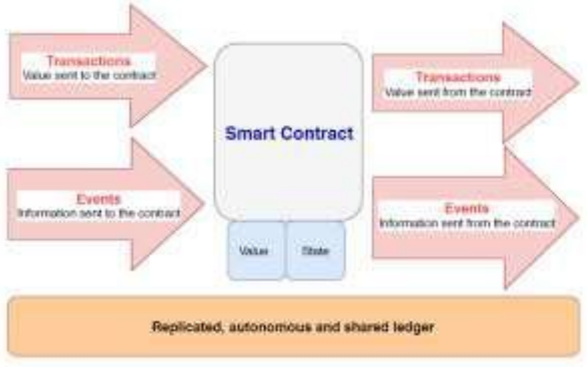
Transactions
Value sent to the contract
Events
Information sent to the contract
Smart Contract
Value
State
Transactions
Value sent from the contract
Events
Information sent from the contract
Replicated, autonomous and shared ledger

Figure 6: Smart Contract

IV. Comparison of blockchain application

Table 1: Comparison of Blockchains: Bitcoin, Ethereum, Multichain, Hyperledger Fabric, Hyperledger Sawtooth

Features	**Bitcoin [1] [3]**	**Ethereum [3] [10]**	**Multichain [1] [3] [16]**	**Hyperledger Fabric v0.6.0 and v1.0.0-rc1 [1] [3] [11]**	**Hyperledger Sawtooth [1] [3]**
Target Application	Cryptocurrency	Cryptocurrency	Digital assets (Provide a platform for creating your own private blockchain)	Enable the creation of private blockchains for industry use cases	Enable companies to deploy their own scalable and flexible blockchains.
What kind of data can be stored?	Cryptocurrency transactions	Cryptocurrency transactions	Any digital asset you want to store.	Chaincode (i.e. smart contracts)	Anything that can be defined by a transaction family
Data Model	Transaction based	Account based	Transaction based	Key - value	Key - value
Scripting or Smart Contract Languages	Golang, C++	Solidity, Serpent, LLL	C++	Golang, Java	Python, Javascript
Smart contract execution	Native	EVM	Native	Dockers	Native
Is the ecosystem open?	Yes	Yes	Configurable	No	Configurable (can be made public)
Native Currency	Bitcoin (BTC)	Ether (ETH or ETC)	N/A	Users can implement a cryptocurrency	Initially provides the MarketPlace transaction family,

				through chaincode	which can track assets
Who are the registration authorities?	N/A	N/A	Configurable	Defined before blockchain is initialized, and more can be assigned while running.	None, unless optional administration key is used.
Is decision making transparent?	Yes	Yes	Configurable	Configurable	Yes; transaction transparency is default
Does it use a managed (Public Key Infrastructure) PKI?	No	No	No	Yes	No
Who manages PKI?	N/A	N/A	N/A	One or more entities in membership services.	N/A
Block-release Timing	10 minutes	12 seconds	Configurable	unknown	Configurable
Transaction Size	200 bytes minimum to 250 bytes average.	Theoretically no max (actual max: 89 kB)	Maximum size configurable	Configurable	unknown

Transaction Rate	3 to 7 Transactions/second	15 to 20 Transansactions/second	configurable	> 10k transactions/second	unknown
Consensus Model	Nodes verify blocks and transactions, then select blockchain with the most blocks.	Similar to Bitcoin, but uses Ethereum Virtual Machine	Fixed ratio of admins approves privilege changes. Longest valid blockchain adopted as global consensus	Pluggable consensus framework; 2 plugins provided: PBFT, and “dummy” plugin. v1.0.0 uses Ordering service (Kafka)	Provides two: Proof of Elapsed Time (PoET), and Quorum Voting
Mining	Proof-of-work	Proof-of-work algorithm called Ethash	Round-robin system proof-of-work requirement is Configurable	N/A	N/A

IV. Discussion

5.1 Observations from the comparative study

After comparing different blockchain applications listed in the above table: 1, it is clear that every implementation of blockchain is unique in terms of purpose (application), data model used, number of ledgers offered, ownership, security mechanisms, consensus protocols, scalability, smart contract scripting language and the execution environment. These differences can be advantageous or disadvantageous depending on the requirements for the application utilizing a blockchain.

While most implementations like Bitcoin target transaction of cryptocurrency, platforms like Multichain aim to improve on their limitations by offering ledgers which can store multiple digital assets and providing restricted visibility to permitted users. It also provides an alternative to proof of work through mining diversity.[3] While on the other hand, implementations like Ethereum and its derivatives, namely Hydrachain, Quorum, Monax, Parity and Dfinity go beyond crypto-currency and asset management by providing ledgers that support running general, user-defined computations called smart contracts. [1]

As can be seen from the table:1 Bitcoin and Ethereum have scalability restrictions due to limited transaction throughput and speed with Ethereum currently only managing a maximum of 20 tps, while Bitcoin only reaches a capacity of 7 transactions per second as its block release time is 10 minute which is very slow. Hyperledger offers solution for this by providing pluggable consensus framework which allows greater control over consensus and restricted access to transactions which results in improved performance scalability and privacy with Hyperledger fabric managing greater than 10k tps. [3]

In Hyperledger, the smart contract is compiled and runs directly on the native machine within Docker environment, thus it does not have the overheads associated with executing high-level EVM byte code as in Ethereum. As a result, Hyperledger is much more efficient in terms of speed and memory usage [1]. However, Hyperledger frameworks are still in development phase and cannot be used for security sensitive applications. Thus, there are no other established applications besides crypto-currency. [3]

Thus, the blockchains which focus on cryptocurrency transactions work well for monetary applications but are not suitable for other kinds of data storage. Most of the blockchains are not yet fully suitable for mainstream usage. Their designs, architecture, codebase and consensus algorithms are still being researched and optimized constantly. There are still many challenges in adopting blockchain technology for mainstream usage which are identified in the following sections along with possible solutions and future implementations to overcome those challenges and improve upon the performance of blockchain.[1] [3]

V. Research Challenges

Scalability issues due to limited transactional throughput, high cost per transaction and slow speed as the network size increases. High energy consumption and computational power required to achieve consensus. Inefficiencies in performance due to arhitechture design of the blockchain network. Managing private keys in public blockchains systems. Difficult and costly to integrate with legacy systems. Problem in porting smart contracts between different blockchain systems. [1] [5]

VII. Performance Improvement Solutions

Segregating the storage, execution engine and consensus layers in order to optimize and scale them independently by outsourcing them to separate entities. Adopting new hardware primitives such as trusted hardware, multi-core CPUs and large memory to improve scaling, contract execution and I/O performance. Sharding the block verification process (consensus mechanism) into parallel sub-entities and then combining the completed data can solve scaling issue as throughput increases with network size. Example: Zilliqa which can handle 2,400tps aims to achieve 8000 tps through sharding. Certain high-level operations can be pre-defined making it easier to write complex smart contracts. [1] [17]

VIII. Conclusion

We first understood the origin and evolution of blockchain technology, structure and formation of a block along with the mining process involved. We distinguish major classes of blockchain systems, namely public, private and consortium blockchains and explain four key features by which current systems can be elaborately categorized. We then classified and compared current implementations such as Bitcoin, Ethereum, Multichain and Hyperledger Fabric and Sawtooth based on various key aspects and metrics of the technology. We then deduce from comparative study that every implementation of blockchain is unique in terms of purpose (application), data model used, number of ledgers offered, ownership, security mechanisms, consensus protocols, scalability, throughput, smart contract scripting language and execution environment. These differences can be advantageous or disadvantageous depending on the requirements of the application utilizing a blockchain network as most of the blockchain systems are not yet fully suitable for mainstream usage due to various challenges and performance issues based on throughput and scalability. Also, their designs, architecture, codebase and consensus algorithms are still being researched and optimized constantly. Most of the current applications focus on monetary transactions which may not be optimal for other type of data storage. The identified solutions and future implementations can help to overcome current challenges as well as provide directions for future research in order to improve the performance and flexibility of blockchains for various applications.

References

[1] Tien Tuan Anh Dinh, Rui Liu, Meihui Zhang, Gang Chen, Beng Chin Ooi and Ji Wang, “Untangling Blockchain: A Data Processing View of Blockchain Systems,” IEEE Transactions on Knowledge and Data Engineering (Volume: 30 , Issue: 7 , July 1, 2018).

[2] Jiin-Chiou Cheng, Narn-Yih Lee, Chien Chi, and Yi-Hua Chen, “Blockchain and Smart Contract for Digital Certificate” from Department of Computer Science and Information Engineering, Southern Taiwan University of Science and Technology, Taiwan in 2018 IEEE International Conference on Applied System Invention (ICASI) April 2018.

[3] Zane Hintzman, “Comparing Blockchain Implementations,” A Technical Paper which was prepared for SCTE/ISBE in 2017 Fall Technical Forum.

[4] J. Morgan and O. Wyman, “Unlocking economic advantage with blockchain. a guide for asset managers.” 2016.

[5] Parke Blake and James Frazer, “Quardev: Blockchain Technology Overview,” published by the Quardev Laboratories. Link:http://www.quardev.com/wpcontent/uploads/sites/13/2018/05/Quardev-Blockchain-Final2.pdf

[6] Stuart Haber and W. Scott Stornetta, “How to Time-Stamp a Digital Document,” 1991. Link: https://www.anf.es/pdf/Haber_Stornetta.pdf.

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

[8] Sriram Sankaran, Sonam Sanju and Krishnashree Achuthan from Center for Cybersecurity Systems and Networks Amrita Vishwa Vidyapeetham, “Towards Realistic Energy Profiling of Blockchains for securing Internet of Thing” in 2018 IEEE 38th International Conference on Distributed Computing Systems held from 2 to July 6th, 2018.

[9] Akash Gaurav, Auxesis Engineering Team, “Auxledger: Enterprise Blockchain Infrastructure for Decentralized Internet,” Technical Induction paper, published on date June 10th, 2018. Link:https://auxledger.org/whitepapers/auxledger-introductory-paper-en.pdf

[10] “Ethereum Project,” Link: https://www.ethereum.org/.

[11] “Hyperledger,” Link: https://www.hyperledger.org/

[12] Collin Thompson “The difference between a private, Public & Consortium Blockchain” published in Blockchain Daily News. Link:https://www.blockchaindailynews.com/The-difference-between-a-Private- Public-Consortium-Blockchain_a24681.html

[13] Shourya Shirsha Nandi, “Technical difference between Ethereum, Hyperledger fabric and R3 Corda,” published in Medium Corporation [US] on March 2016. Link:https://medium.com/@micobo/technical-difference-between ethereum- hyperledger-fabric-and-r3-corda-5a58d0a6e347

[14] Hyperledger, “Hyperledger Fabric Model,” Link:https://hyperledger-fabric.readthedocs.io/en/release-1.3/fabric_model.html

[15] Yong Shi, “Secure storage service of electronic ballot system based on block chain algorithm”, Department of Computer Science, Tsing Hua University, Taiwan, R.O.C., 2017 Sofocle Technologies, “Blockchain in the Healthcare Sector,” Link: https://www.sofocle.com/industry/blockchain-in-healthcare/

[16] Antony Lewis “In a nutshell: MultiChain (Epicenter Bitcoin interview - Nov 2015),” published in Bits on Blocks on March 7th, 2016 Link:https://bitsonblocks.net/2016/03/07/in-a-nutshell-multichain-epicenter-bitcoin-interview-nov-2015/

[17] Bitrewards, “Blockchain Scalability: The Issues, and Proposed Solutions,” published in A Medium Corporation [US] on April 25, 2017. Link:https://medium.com/@bitrewards/blockchain-scalability-the-issues-and- proposed-solutions-2ec2c7ac98f0 J. Breckling, Ed., The Analysis of Directional Time Series