

An overview of Hyperledger blockchain technologies and their uses

Vladimir Milićević*, Jovana Jović*, Nemanja Zdravković*

*Faculty of Information Technology, Belgrade Metropolitan University, Belgrade, Serbia
{vladmir.milicevic, jovana.jovic, nemanja.zdravkovic}@metropolitan.ac.rs

Abstract—Hyperledger is an open source community focused on developing a suite of stable frameworks, tools and libraries for enterprise-grade blockchain deployments. It presents an umbrella term comprising different blockchain technologies (BCTs), including distributed ledgers, libraries, and tools. Due to the variety of technologies, it is often not so easy to choose the correct BCT depending on the needs of a blockchain-based application.

In this paper, we present an overview of the most (and less) popular Hyperledger distributed ledgers, comparing their similarities and differences between themselves and with Bitcoin and Ethereum, with the ultimate goal of simplifying the choice of ledger for a given application.

Index Terms—Blockchain technology, Bistributed ledger, Hyperledger, Fabric, Sawtooth, Burrow, Iroha, Besu, Indy

I. INTRODUCTION

Blockchain Technology (BCT) has surpassed its initial use in cryptocurrencies such as Bitcoin [1] and Ethereum [2], and is widely considered a disruptive technology that may change the method of communications over the Internet [3], [4]. While the key aspects of all BCTs remain mostly the same, such as having one or multiple transactions form blocks that form a chain, different approaches of block formation, levels of access and most importantly, consensus mechanism, heavily differ in various technologies [5]. One of the most important BCT projects is Hyperledger [6], maintained by the Linux Foundation, which focuses on open source BCTs. Hyperledger today refers to an umbrella term that encompasses multiple distributed ledger technologies, as well as libraries and tools. For different BCT-based applications, not all distributed ledgers, termed projects throughout the paper, are equally suitable.

The aim of this paper is to examine the most popular Hyperledger projects: Fabric, Sawtooth, Iroha, Indy, Burrow, and Besu and do a comparative analysis on their approach to distributed ledger technology, with emphasis on their applicability to various BCT-based applications.

When researching different Hyperledger projects, the authors of this paper started with the question of "why are some BCTs better suited for certain applications?" For instance, Bitcoin and Ethereum are still the leaders in cryptocurrency, while Hyperledger-based BCTs are being considered in almost every healthcare application [7], [8]. One significant advantage of using BCT is that it can reform the interoperability of healthcare databases, providing authorized access to patient medical records, and to other hospital assets [9]. Supply chain applications also prefer Hyperledger [10]. In this overview paper, we wanted

to point out the key characteristics and key differences between provided Hyperledger projects (Fabric, Sawtooth, Iroha, Indy, Burrow, and Besu), with an additional comparison to Bitcoin and Ethereum. Furthermore, we consider the suitability of using said Hyperledger projects across different domains and thematic areas, alongside additional libraries and tools that are provided by the Hyperledger community which can affect procedures/processes.

This paper is organized as follows: Section II presents an overview of Blockchain technology and related frameworks. Section III contains a description of Hyperledger distributed ledgers. Section IV compares Hyperledger distributed ledgers with Ethereum and Bitcoin. Finally, Section V concludes the paper.

II. BLOCKCHAIN OVERVIEW

Blockchain imposes fundamental changes to the way personal data are currently being processed, and can improve current data security solutions. A Blockchain is a shared, append-only distributed ledger, in which all transactions, which can describe events (e.g. changes to bank accounts, updates to a electronic health record, each step in a supply chain, etc.) are stored in linked blocks [11]. Every transaction, apart from the data, contains a unique cryptographic signature, ensuring the ledger is resilient to modifications. In addition, this ledger is hence simultaneously shared across all members of the network, called nodes, resulting in real-time node update. A block can be viewed as a data structure consisting of a set of transactions, together with a header that connects the new block to the previous one. All blocks hence form a chain, and can trace back to the first block, called the genesis block. A blockchain relies on peer-to-peer networks, public-key cryptography, and distributed consensus. The combination of these three concepts is what secures blockchain transactions. Unlike a centralized system, no single entity should be able to control the process of adding a block to the chain, and all member nodes share equal rights, with every single block is at all time managed by all member nodes. This management system is accomplished with distributed consensus. This process establishes an agreement among the nodes in the blockchain network in the validation of each data block to be added to the chain. Depending on the consensus algorithm, nodes can either compete for correct transaction validation, be chosen randomly, or apply a different algorithm altogether.

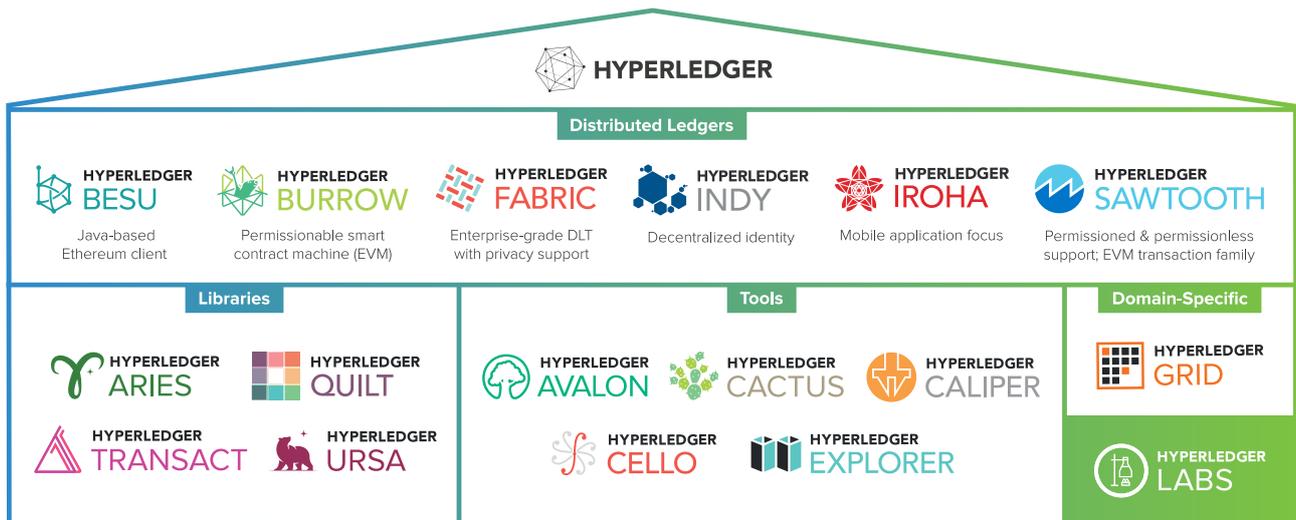


Fig. 1. The Hyperledger Project umbrella.

It is important to note that Blockchains are a class of technology; the term refers to different forms of distributed databases with variations in their technical and governance arrangements and complexity.

III. THE HYPERLEDGER PROJECT

Hyperledger is an open source community focused on developing a suite of stable frameworks, tools and libraries for enterprise-grade blockchain deployments [6]. It aims to advance blockchain technology by identifying and realizing a cross-industry open standard platform for distributed ledgers, which can transform the way business transactions are conducted globally [6]. Similar to The Linux Foundation, Hyperledger has a modular approach to hosting projects, as shown in Fig. 1. All Hyperledger projects are open source, they are easy to obtain. All Hyperledger projects, except Hyperledger Indy, share a common similarity, and that is their general purpose application, differently from Hyperledger Indy, which focuses on decentralized identity.

One of the key differences between the projects was their consensus mechanism. Due to the variety of blockchain usage requirements, Hyperledger is working on several different consensus mechanisms [12]. Fabric uses the Apache Kafka platform [13] as the main Crash Fault Tolerance (CFT) protocol on the network which is permissioned voting-based. Consensus in Hyperledger Indy is based on Redundant Byzantine Fault Tolerance (RBFT), which is a protocol inspired by Plenum Byzantine Fault Tolerance (Plenum). Hyperledger Iroha introduces a BFT consensus algorithm called Sumeragi, which tolerates numbers of Byzantine faulty nodes in a network. Hyperledger Sawtooth facilitates pluggable consensus for both lottery and voting algorithms. By default, Hyperledger Sawtooth uses a lottery-based, Nakamoto consensus algorithm called Proof of elapsed time (PoET). Hyperledger Burrow comes with Byzantine Fault-Tolerant Tendermint protocol with a greater transaction rate, whereas Buru implements various consensus algorithms that are involved

in transaction validation, block validation, and block production, i.e. mining in Proof of Work (PoW). Hyperledger Sawtooth has the most support for smart contract languages.

The core Hyperledger-based use cases are banking, healthcare, supply chain management, financial services, information technology, government, and media and entertainment. For example, Hyperledger Indy offers suitability for the banking use case where applicants can share only the information the banks need to make a decision in a way that guarantees truth. In financial services, Sawtooth provides a reliable and powerful way to support post-trade activities, but also, Fabric can be deployed as fully disjoint networks with separate endorser sets and ordering nodes to provide privacy and confidentiality. Credentialing in the hospitals as a way to performing "due diligence" on the medical staff represents only one of many subcases in healthcare. Credentialing provides a good use case for Blockchain where Indy provides off-the-shelf solutions that would otherwise require architecting and developing new software. Sawtooth can help in supply chain management accommodates both domain-specific data and the transaction families that operate on it, including data constraints such as verifying the calibration of a sensor.

Hyperledger incubates and promotes a range of business Blockchain technologies, including many libraries and tools that provide support for the creation, maintenance, deployment, providing cryptographic work, etc. This paper aims to find how additional libraries and tools provided by the Hyperledger community affect procedures/processes in found use cases.

Table I shows a summary of the key similarities and differences between Hyperledger projects.

IV. COMPARISON WITH BITCOIN AND ETHEREUM

Hyperledger is often compared to Bitcoin and Ethereum [12]. Presented below are the differences in participation, consensus, and the limitations of each technology, summarized in Table II [14].

TABLE I
KEY SIMILARITIES AND DIFFERENCES BETWEEN VARIOUS HYPERLEDGER PROJECTS.

Project	Sawtooth	Fabric	Indy	Burrow	Iroha	Besu
Advantages	Distributed state agreement, Adapters for transaction logic, Versatility, Scalability, Transaction families	Enterprise backing, Relative maturity, Private channels, Modular architecture, Smart contracts	Identity management	Lower barrier to entry, Use of the Ethereum Virtual Machine (EVM)	Mobile libraries	Network client variety, Plugins, Monitoring on Besu
Consensus mechanism	Proof of Elapsed Time, Practical Byzantine Fault Tolerance, Raft, Devmode	Kafka, Raft, Solo	Redundant Byzantine Fault Tolerance (RBFT)	Tendermint	Sumeragi	PoW, and PoA (IBFT, IBFT 2.0, Etherhash, and Clique)
Smart contract technology	Transaction Families	Chaincode	None	Smart contract application engine	Chaincode	
Smart contract type	On-chain and Installed	Installed	None	On-chain	On-chain	
Smart contract language	C++, Go, Java, JavaScript, Python, Rust, or Solidity (through Seth)	Go, Javascript, Solidity	Java, None	Native language code	Native language code	Java
State storage	Central database, Ildb	CouchDB or leveldb	RocksDB	Google's Protocol Buffers	Kura	RocksDB

A. Mode of Participation

Based on restrictions on who can view or read transaction data from the blockchain network, public and private blockchains can be distinguished. In public blockchains, no restrictions exist on reading transaction data, and essentially everyone can download the blockchain ledger and view all transactions. Bitcoin and Ethereum fall into this category. Private blockchains, conversely, enable direct access to blockchain data that is limited to predefined users, and only participants that are registered on the blockchain network can download the ledger. All Hyperledger projects discussed previously are private blockchains.

Permission restrictions decide whether transaction processors (often called miners in Bitcoin and Ethereum) who submit data and are eligible to create blocks of data can do so without permission or are restricted to do so and need permission from a central authority. Whereas transaction processing in permissioned blockchains such as Hyperledger is performed by predefined users, Bitcoin and Ethereum are permissionless blockchains where there are no restrictions on the identities of processors, enabling everyone to start mining in order to create blocks.

B. Consensus Mechanism

The consensus mechanism is a means to determine consensus about all transactions and the current state of the system. The mechanism ensures that transactions will only be added to the blockchain if valid and never recorded more than once. Ethereum and Bitcoin use PoW,

which means that miners have to solve a computational difficult problem to ensure the validity of new transactions. Various modifications to the Byzantine Fault Tolerance (BFT) consensus mechanism are often used in permissioned blockchains. For instance, in Practical Byzantine Fault Tolerance (PBFT), a new block is added if more than 2/3 of all validating peers submit the same response. Hyperledger Fabric, as one of the most common used Hyperledger Projects, does not provide PBFT support natively, but offers its users to add this consensus mechanism in a modular manner.

C. Limits

Scalability represents a main challenge, specifically for the Bitcoin and Ethereum networks where a copy of the complete history needs to be stored in each node. As for the permissioned Hyperledger technology, this network can scale independently for each node without any disruption because peers are abridged into endorsers, committers and consenters. Also, the parallel transactions processing in the Hyperledger Blockchain leads to a higher throughput. Principally, the one-megabyte block size in a Bitcoin network not only causes delays (7 transactions/second) but also leads to the drop of not conform blocks. To overcome this limitation a Request Management System based on advertising requests was introduced as well as static time-outs. Also, in Ethereum, many functions are duplicated leading to scalability issues. Thus, the introduction of state channels and plasma chan-

TABLE II
KEY DIFFERENCES BETWEEN BITCOIN, ETHEREUM, AND HYPERLEDGER [14]

Features	Bitcoin	Ethereum	Hyperledger
Community	Bitcoin developers	Ethereum developers	Linux foundation
Mode of Peer Participation	Public and Permissionless Network	Public/Private and Permissionless Network	Private and Permissioned Network
Cryptocurrency	Built-in, Bitcoin	Built-in, Ether	No built-in cryptocurrency
Consensus Mechanism	PoW: Consensus is reached by mining	PoW: Consensus is reached by mining	Pluggable (No-op – no consensus needed and PBFT – Practical Byzantine Fault Tolerance): No mining required
Confidentiality	Transparent	Transparent	Confidential transactions
Private transaction mode	No	No	Yes
Stimulus	Economics incentive, fees and rewards	Economics incentive, fees and rewards	Reputational Risk
Limit	7 transactions/sec	20 transactions/sec	No
Applications	Digital Registry, Crypto Currency	Currency, Smart Contracts	Digital Registry, Smart Contracts
Smart contract languages	No	Solidity, Serpent, Mutan, LLL	Chaincode
Languages	C++	Golang, C++, Python	GoLang, Java
Variants	700+ variants	Homestead, Metropolis, Serenity	Burrow, Fabric, Iroha, Indy, Sawthooth, Besu

nels in Ethereum networks to perform some transactions offline and conduct off-chain activities which leads also to enforce anonymity.

V. DISCUSSION AND CONCLUSION

Our comparative study showed that although considered a single product (or a variation of a single product), the Hyperledger family of distributed ledgers varies in many different aspects. Indeed, the different Hyperledger projects can be used for different applications; however, Hyperledger projects can also be combined when considering a more complex use of Blockchain-based services. In addition, the Hyperledger product umbrella features different tools and libraries, such as Aries for digital certification, Ursa cryptographic support, Quilt for payment support, and Transact for smart contract support. The authors concluded that Hyperledger projects, tools, and libraries present a complete environment for Blockchain development

ACKNOWLEDGMENT

This paper was supported by the Blockchain Technology Laboratory at Belgrade Metropolitan University, Belgrade, Serbia.

REFERENCES

- [1] S. Nakamoto and A. Bitcoin, "A peer-to-peer electronic cash system," *Bitcoin*.—URL: <https://bitcoin.org/bitcoin.pdf>, vol. 4, 2008.
- [2] G. Wood *et al.*, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, no. 2014, pp. 1–32, 2014.
- [3] J. Mattila, "The blockchain phenomenon—the disruptive potential of distributed consensus architectures," ETLA working papers, Tech. Rep., 2016.
- [4] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Business & Information Systems Engineering*, vol. 59, no. 3, pp. 183–187, 2017.
- [5] V. J. Morkunas, J. Paschen, and E. Boon, "How blockchain technologies impact your business model," *Business Horizons*, vol. 62, no. 3, pp. 295–306, 2019.
- [6] C. Cachin *et al.*, "Architecture of the hyperledger blockchain fabric," in *Workshop on distributed cryptocurrencies and consensus ledgers*, vol. 310, no. 4. Chicago, IL, 2016.
- [7] N. Zdravkovic, M. Damjanovic, D. Domazet, V. Ponnusamy, and M. Trajanovic, "Implementing Blockchain Technology for Data Protection in Health-IoT and Diagnostic Devices: Overview, Potential, and Issues," in *Proceedings of the 10th International Conference on Information Society and Technology (ICIST 2020)*, Mar 2020, pp. 211–215.
- [8] V. Grkovic, J. Jovic, N. Zdravkovic, M. Trajanovic, D. Domazet, and V. Ponnusamy, "Usage of Blockchain Technology for Sensitive Data Protection – Medical Records Use Case," in *Proceedings of the 10th International Conference on Information Society and Technology (ICIST 2020)*, Mar 2020, pp. 216–221.
- [9] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of medical systems*, vol. 42, no. 7, pp. 1–7, 2018.
- [10] M. P. Caro, M. S. Ali, M. Vecchio, and R. Giaffreda, "Blockchain-based traceability in agri-food supply chain management: A practical implementation," in *2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany)*. IEEE, 2018, pp. 1–4.
- [11] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*. IEEE, 2017, pp. 557–564.
- [12] J. Moubarak, E. Filiol, and M. Chamoun, "Comparative analysis of blockchain technologies and tor network: Two faces of the same reality?" in *2017 1st Cyber Security in Networking Conference (CSNet)*. IEEE, 2017, pp. 1–9.
- [13] B. R. Hiranman *et al.*, "A study of apache kafka in big data stream processing," in *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*. IEEE, 2018, pp. 1–3.
- [14] J. Moubarak, E. Filiol, and M. Chamoun, "On blockchain security and relevant attacks," in *2018 IEEE Middle East and North Africa Communications Conference (MENACOMM)*. IEEE, 2018, pp. 1–6.