Practical comparison of distributed ledger technologies for IoT

Val A. Red

Val A. Red, "Practical comparison of distributed ledger technologies for IoT," Proc. SPIE 10206, Disruptive Technologies in Sensors and Sensor Systems, 102060G (4 May 2017); doi: 10.1117/12.2262793

SPIE.

Event: SPIE Defense + Security, 2017, Anaheim, California, United States

Practical Comparison of Distributed Ledger Technologies for IoT

Val A. Red*
Air Force Research Laboratory Information Directorate, 525 Brooks Rd, Rome, NY 13441

ABSTRACT

Existing distributed ledger implementations – specifically, several blockchain implementations – embody a cacophony of divergent capabilities augmenting innovations of cryptographic hashes, consensus mechanisms, and asymmetric cryptography in a wide variety of applications. Whether specifically designed for cryptocurrency or otherwise, several distributed ledgers rely upon modular mechanisms such as consensus or smart contracts. These components, however, can vary substantially among implementations; differences involving proof-of-work, practical byzantine fault tolerance, and other consensus approaches exemplify distinct distributed ledger variations. Such divergence results in unique combinations of modules, performance, latency, and fault tolerance. As implementations continue to develop rapidly due to the emerging nature of blockchain technologies, this paper encapsulates a snapshot of sensor and internet of things (IoT) specific implementations of blockchain as of the end of 2016. Several technical risks and divergent approaches preclude standardization of a blockchain for sensors and IoT in the foreseeable future; such issues will be assessed alongside the practicality of IoT applications among Hyperledger, Iota, and Ethereum distributed ledger implementations suggested for IoT. This paper contributes a comparison of existing distributed ledger implementations intended for practical sensor and IoT utilization. A baseline for characterizing distributed ledger implementations in the context of IoT and sensors is proposed. Technical approaches and performance are compared considering IoT size, weight, and power limitations. Consensus and smart contracts, if applied, are also analyzed for the respective implementations' practicality and security. Overall, the maturity of distributed ledgers with respect to sensor and IoT applicability will be analyzed for enterprise interoperability.

Keywords: blockchain, distributed ledgers, Internet of Things (IoT), directed acyclic graphs, Hyperledger, Iota, Ethereum

1. INTRODUCTION

Distributed ledgers enable efficient data decentralization and unprecedented traceability by recording linked records of cryptographically secure hashes validated and propagated across nodes and peers. This enables disintermediation, efficient recording of interdependent transactions, variable validators, and data sharing in environments of distrust. Disintermediation is accomplished in that there are traditionally no centralized third parties required to facilitate transactions; rather, a multitude of nodes and peers broadcast and record transactions' state. As hashes are recorded in a linked fashion within blocks, distributed ledgers are effective in capturing interdependent transactions. Depending on the specific application of a blockchain or distributed ledger and the network of peers it involves, disparate validators may be defined in implementation to minimize collusion or other malicious activities among peers. Securely sharing data in an environment with varying degrees of trust becomes achievable as every transaction becomes immutable once recorded in a block spread across multiple nodes maintaining a distributed ledger; furthermore, each transaction is guaranteed nonrepudiation and a high degree of integrity via digital signatures and asymmetric cryptography. Several of these properties are extensible towards assuring communications involving sensors and the internet of things (IoT), which currently face risks such as compromise of data in transit and overall device exploitation.

IoT generally refers to physical devices and sensors or actuators connecting and communicating via the Internet; several are edge devices [1]. Many IoT devices include capabilities of remote access, control, and other cyber-physical interactions; however, such convenience also comes with the risk of malicious targeting of IoT data in transit and exploitation of the devices themselves for lack of robust security and integrity features. Consequences of realizing such risks and device limitations have made trust difficult to implement for IoT overall [2]. Many IoT devices are limited in functionality and security due to constraints of size, weight, and power. Integrity, traceability of data, and nonrepudiation are all security features of blockchain and distributed ledger technologies that can potentially be leveraged towards mitigating IoT risks at low cost. As both IoT assurance and distributed ledgers are both areas of emerging capability, however, possible combinations of these new technologies demand due diligence in scrutinizing existing implementations, analyzing performance, and assessing the practicality of integrating such technologies. Hyperledger, Iota, and Ethereum are three distributed ledgers endeavoring towards IoT applicability.

*red@val.moe; https://val-red.com

Disruptive Technologies in Sensors and Sensor Systems, edited by Russell D. Hall, Misty Blowers, Jonathan Williams,
Proc. of SPIE Vol. 10206, 102060G · © 2017 SPIE · CCC code: 0277-786X/17/$18 · doi: 10.1117/12.2262793

Hyperledger represents several open source blockchain implementations variably backed by many organizations across industry; one implementation specifically focuses on IoT [3,4]. Ethereum is a distributed ledger that began as a blockchain cryptocurrency, later integrating directed acyclic graph elements for additional security as applications expanded from cryptocurrency features to other potential applications, such as IoT [5]. Finally, Iota is a distributed ledger built specifically for IoT purposes; Iota departs entirely from specifically implementing blockchain and instead utilizes directed acyclic graphs to link transactions rather than grouping them into blocks [6]. All these distributed ledgers share common characteristics, but include distinct approaches of modules and features intended for security and IoT applicability.

Current research involving either the practicality or comparison of distributed ledgers predominantly focuses on cryptocurrencies far more than IoT applicability, while even security research was fractional – out of 41 papers published up until late 2016, 80.5% of papers focus on bitcoin while 34% address blockchain assurance [7]. Other research focuses on variations of modules or low-level comparisons of consensus implementations [1, 8]. Distributed ledger implementations and security of IoT will continue to evolve dramatically in the in-term future; this paper contributes a benchmark of current state of the art.

A benchmark is accomplished in a practical comparison of three distributed ledger implementations with known IoT associations. First, common distributed ledger characteristics will be established as a baseline for implementations discussed. IoT devices will be discussed in the context distributed ledgers. Fundamental differences in application and performance among Hyperledger, Iota, and Ethereum will be assessed. Finally, all three will specifically be analyzed for applicability towards IoT devices. Overall, the summary and conclusions shall point towards the maturity of current distributed ledger technologies in the context of IoT interoperability.

2. DISTRIBUTED LEDGER CHARACTERISTICS

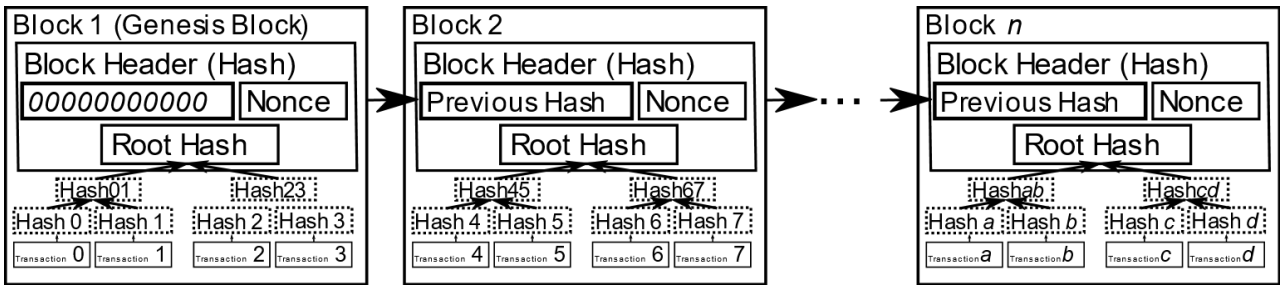

Figure 1. A representation of a blockchain based on the bitcoin implementation; note that to achieve nonrepudiation, every single transaction hashed under the root hash of a block are digitally signed by the subsequent sender and contain the public key of the transaction recipient. Graphic adapted from [9].

Distributed ledger technologies developed into a practical means of immutably and securely recording hashes of data with the creation of the Bitcoin cryptocurrency and its underlying blockchain implementation in 2008 [9]. Specifically, blockchain can be considered a subset of distributed ledger technologies defined by the linking of one block representing several transactions to a single, chronologically preceding block; following these links, every block can trace back to the original genesis block – or first block in the blockchain – as in figure 1. This arrangement of hashes is nontrivial to maliciously corrupt at scale. A single compromised node would practically be unable to maliciously redefine or delete a transaction already recorded several blocks deep across a network of hundreds or thousands of peers.

All transactions in a block, represented by the root hash included in the block header, are assured nonrepudiation by means of asymmetric cryptography: the transaction's sender digitally signs the previous hash with the sender's private key and adds the recipient's public key. The recipient, in turn, validates the incoming transaction with the sender's public key and uses the recipient's private key to digitally sign a subsequent transaction. Storage efficiency is achieved by means of the root hash representing a Merkle Tree, such that completed transactions can be safely excluded from the final block propagating among peers [10]. Overall, the immutability, integrity, and ownership of data recorded in a blockchain is assured at low cost.

There are variations to linking among blocks that depart from blockchain's simple association by previous block header hash. Directed acyclic graphs, for example, are an alternative arrangement of a distributed ledger that links hashes to multiple other hashes instead of a one-to-one linking; this is illustrated in figure 2. Overall, however, the fundamental, cryptographically secure linking among blocks in general unifies all distributed ledger implementations.

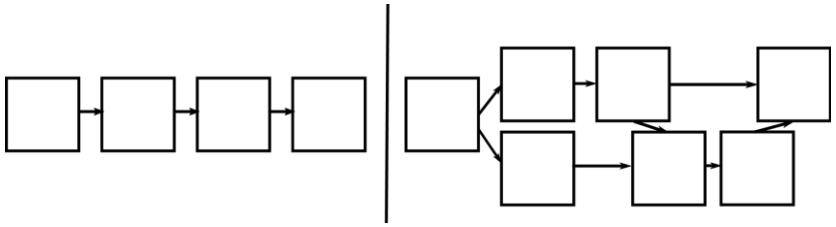

Figure 2. A simplified representation of a blockchain (left) compared with a directed acyclic graph implementation of a distributed ledger (right); compared with blockchain, blocks created in a directed acyclic graph can link to multiple previous blocks.

Most blockchain and distributed ledger implementations can be distinguished by what type of data is being hashed, such as records of cryptocurrency transactions, in addition to the application of additional modules, such as combinations of consensus or smart contracts, on top of the fundamental linking of blocks described. In addition to this typical blockchain structure, most distributed ledgers enhance integrity and prevent failures – such as attempting to spend the same cryptocurrency in two different transactions at once – by means of enforcing consensus.

Consensus embodies a modular process of deciding amongst peers how exactly transactions are grouped into deconflicted blocks and accepted into a blockchain. Bitcoin, for example, utilizes proof-of-work – a computationally-expensive process in which multiple nodes collect transactions and probabilistically produce a block header hash value – which is impractical with IoT specifications and can potentially be abused to prevent consensus in small scale networks conflicted on dependent transactions [9, 11]. Additionally, the modular feature of smart contracts built on top of some distributed ledger implementations enable automatically executing scripts to conditionally validate a transaction for properties not already deconflicted through consensus, such as legal terms [8,12]. Chaincode, implemented in the Go programming language, exemplifies a practical means of creating smart contracts [13]. Flawed implementations of smart contracts, however, are also an avenue for further malicious exploitation [14, 18]. Nevertheless, mature modules of consensus and smart contracts can enhance the security of a distributed ledger implementation by ensuring data integrity amongst peers. Table 1 breaks down distributed ledger characteristics and modules.

Table 1. Distributed ledger characteristics and modules and examples varying across implementations.

Characteristic	Example
Linking of blocks	Blockchain, directed acyclic graphs
Secure hashing	SHA-2, SHA3
Asymmetric cryptography	Diffie-Hellman, RSA
Digital Signatures	ECDSA
Consensus	Proof-of-work (PoW), Practical Byzantine Fault Tolerance (PBFT)
Smart Contracts	Chaincode

3. INTERNET OF THINGS AND DISTRIBUTED LEDGERS

Trust across IoT devices presents a difficult problem, predominantly due to divergence in implementation. Not unlike IoT, distributed ledger implementations can vary significantly. Regardless of differences among distributed ledgers, all implementations predominantly apply linking of data structures, secure hashing, asymmetric cryptography, and digital signatures in a way that mitigates trust risks faced by IoT devices.

Decentralizing communications to be both cryptographically secure, immutable, and auditable is a major advantage leveraged by distributed ledgers and their modules that can enhance IoT devices [8]. It is important to also consider IoT device limitations: size, weight, and power can constrain interoperability with distributed ledger technologies. Performance requirements for consensus can be a largely limiting factor; for example, proof-of-work would be prohibitively expensive from a computation perspective with IoT devices alone, and applying similarly expensive consensus mechanisms would be difficult to organize an IoT-centric network infrastructure around. Most IoT devices would be unable to efficiently collect and calculate possible hashes for several of other devices' communications at the same time as its own. Additionally, depending on how the distributed ledger is propagated and stored across IoT devices, storage can be another factor adversely affected by the size, weight, and power of IoT devices. Some distributed ledger implementations require complete ledger copies to be stored by every single peer; this would be difficult for an IoT distributed ledger network to scale [5]. Depending on how exactly blocks are formed and subsequently stored in a ledger, each device's copy can be gigabytes large in practice; within 5 years of its first, approximately 285-byte large genesis block, a copy of the entire Bitcoin ledger reached 30 gigabytes [15]. Thus, it is important to consider the combination of implementation and modules leveraged by distributed ledger technologies in the context of IoT.

IoT capabilities of remote access and cyber-physical interaction enable the possibility of programmatic execution of tasks based on sensor and actuator data; assuring integrity of information and automating subsequent tasks are both possible with distributed ledgers and, specifically, smart contract integration [1,8]. Smart contracts are self-executing scripts that can account for conditionally validating and procedurally reacting to data in a way that consensus would be unable to effect on its own. While consensus verifies integrity of communications themselves, smart contracts enable programmatic routines that efficiently leverage IoT data. Furthermore, when implemented with all other distributed ledger capabilities, smart contracts allow secure automation of otherwise time-consuming flows of cyber-physical tasks in-band. One example would be leveraging smart contracts to share actionable energy usage data among IoT devices to economically automate what power modes they operate in throughout a day or other periods of time. This data, due to the immutability and auditability of a distributed ledger, could even be used to directly transact between an electricity vendor and user. These capabilities are furthermore supported by distributed ledgers' inherent context of interdependent communications, disintermediation, validation amongst peers, and assured integrity in networks of varying trust.

Not all distributed ledger implementations, however, utilize smart contracts. There are also substantial differences even among distributed ledger or blockchain systems that do incorporate smart contracts. Hyperledger, Iota, and Ethereum are three widely used, and are closest to production or enterprise interoperability for IoT applications.

4. COMPARISON OF HYPERLEDGER, IOTA, AND ETHEREUM

Hyperledger, Iota, and Ethereum are three open source implementations of distributed ledgers. They all share the fundamental characteristics of linking among blocks, cryptographically secure for hashing of multiple transactions compactly in single blocks, asymmetric cryptography, digital signatures, consensus and smart contracts; the specific respective implementations vary significantly. Furthermore, the extent of applicability to IoT also differs among the three in how aspects – particularly smart contracts – are applied.

4.1 Hyperledger

Hyperledger is a permissioned blockchain: it uniquely applies access control, chaincode-based smart contracts, variable consensus with a current implementation of practical byzantine fault tolerance (PBFT), and includes anchors of trust to root certificate authorities as an enhancement to the asymmetric cryptography and digital signature features with SHA3 and ECDSA [16]. While most distributed ledgers enable open enrollment, the permissioned aspects of Hyperledger enhance security by means of preventing Sybil attacks, an attack in which consensus could potentially be threatened by a malicious entity creating and enrolling illegitimate peers – or Sybil identities – to adversely affect the network [17]. Furthermore, its implementation of smart contracts involves the chaincode implementation, which can self-execute conditions such as asset or resource transfers among peers in hundreds of milliseconds [1, 3]. This latency is low among comparative distributed ledger implementations.

Hyperledger application of PBFT prevents the probabilistic and computationally expensive mining of hashes as in proof-of-work; however, it trades off immediate computational overhead with network utilization. Multiple validating

peers are broadcast the transaction and converge upon a deterministic execution, and subsequently, the same block [16]. To accomplish this, the validating peers must also intercommunicate, causing more network overhead.

In an IoT context, the scale at which network utilization increases with the number of devices on a network must be investigated and measured further. Overall, between applying chaincode smart contracts and a unique PBFT implementation that offsets computational overhead for increased networking among peers, Hyperledger offers robust versatility for IoT applications.

4.2 Iota

Iota is a unique distributed ledger in that it does not utilize an explicit blockchain at all; instead, it implements a directed acyclic graph of transactions – instead of blocks of multiple transactions that link together, each transaction approves and links back to two other transactions. This is intended to be especially lightweight, as consensus does not require several peers intercommunicating or exhausting computational effort validating additions to the directed acyclic graph; instead, two transactions can be validated by single peers committing a transaction themselves. This minimizes transaction time and overhead; however, this does present a few attack scenarios. With several of Sybil identities and enough computing power, an attacker could potentially orchestrate a simultaneous, conflicting transactions at once [6]. The lightweight nature and unique application of directed acyclic graphs makes it among the most versatile implementations of a distributed ledger; however, there are some tradeoffs.

The lightweight and versatile nature of Iota offsets both computational and network overhead, making it especially extensible towards a range of IoT applications with some tradeoffs. Some elements of robustness, such as a built-in smart contract mechanism, are excluded to keep Iota as lightweight as possible. This limits the full range of distributed ledger capabilities that could extend to IoT; furthermore, due to Iota being among the most recent of emerging implementations, its latency and scalability have yet to be tested at scale. Iota may be newer and not as robust as other distributed ledgers; however, its unique application of directed acyclic graphs and its lightweight means of propagating transactions can potentially make it among the fastest IoT implementations.

4.3 Ethereum

Ethereum, although predominantly intended for cryptocurrency, is an adaptable blockchain implementation with an implementation of smart contracts and a derivative of proof-of-work consensus known as Ethash. This also applies directed acyclic graphs to manage probabilistic hash generation such that it prevents potential abuse from specialized hardware that other proof-of-work algorithms are vulnerable to [5, 11]. In addition to implementing smart contracts, Ethereum transactions can also store custom data. This increases the potential for auditability and immutability of IoT data beyond cryptocurrency transactions. This allows robust extensibility for IoT applications with some performance tradeoffs.

Due to Ethash being based upon proof-of-work, Ethereum may require between 10 to 20 seconds to produce a block. High-frequency and time-sensitive IoT device operations may not support such delays [1, 8]. While Ethash prevents abuses from potential specialized hardware, it does not necessarily enhance fault tolerance. At scale, IoT devices would need to rely on trusted and computationally powerful peers to ensure fault handling. Storage also presents another problem; Ethereum requires all peers store a blockchain that is also in the order of tens of gigabytes large. IoT devices will either need to intercommunicate with a proxy server that acts as a peer in the Ethereum network or accommodate large storage.

Ethereum, in active use as a cryptocurrency longer than most distributed ledger implementations, does already also have IoT prototypes. For example, Ethereum is handles tokens and contracts for electronic lock sharing and supply chain assurance prototypes [8]. To become practical, individual IoT device performance with Ethereum would be desirable, however; additionally, smart contract security and maturity should be reassessed as it has encountered significant changes since a theft of funds due to a smart contract implementation flaw [18]. Nevertheless, Ethereum boasts practicality, user acceptance, and the ability to apply smart contracts – all of which extend towards robustness in IoT potential.

5. SUMMARY

Distributed ledger implementations and IoT technologies are both divergent and disruptive areas of development. Research on capabilities and interoperability will continue to develop rapidly as both distributed ledgers and IoT devices transform over time. Hyperledger, for example, is under development to expand from PBFT into more modular combinations of consensus approaches to accommodate nondeterministic transactions and smart contracts [16]. Hyperledger, Iota, and Ethereum, as represented in table 2, may be substantially different over time.

Table 2. Hyperledger, Iota, and Ethereum in the context of IoT

Distributed Ledger	Consensus	Transaction time	Smart Contracts	Other Characteristics
Hyperledger	PBFT	10s-100s ms	Yes	Computationally light, network-intensive
Iota	N/A (DAG)	10s ms	No	Computationally light, low network use
Ethereum	Ethash (PoW-based)	10,000 ms	Yes	Computationally expensive, low network use

Overall, different types of IoT applications may be most compatible with specific characteristics among distributed ledger implementations. IoT sensor and actuator data that may not require frequent updates but does require a large degree of integrity assurance via consensus and the ability to include custom data may warrant Ethereum. Time-sensitive sensor and actuator data involving quick transactions and not requiring additional functionality of smart contract enforcement may be extensible towards Iota and its directed acyclic graph approach. For fault tolerance, quick transactions, smart contract compatibility, and high-bandwidth networks – Hyperledger may be preferable. Overall, distributed ledgers' have the potential to assure integrity of IoT data in a robust diversity of practical applications; how exactly the introduction of new distributed ledgers and updates to existing implementations may transform the performance and interoperability with IoT devices may be elucidated in future research.

REFERENCES

[1] Samaniego, M. and Deters, R., "Hosting Virtual IoT Resources on Edge-Hosts with Blockchain," Proc. IEEE CIT 2016, 116-119 (2016).

[2] Yan, Z., Zhang, P. and Vasilakos, A.V., "A survey on trust management for Internet of Things," Journal of network and computer applications 42, 120-134 (2014).

[3] Smith, B. and Christidis, K., "IBM Blockchain: An Enterprise Deployment of a Distributed Consensus-based Transaction Log," Proc. Fourth International IBM Cloud Academy Conference, 140-143 (2016).

[4] Gantait, A., Patra, J. and Mukherjee, A., "Implementing blockchain for cognitive IoT applications, Part 1," IBM DeveloperWorks, 9 January 2017, <https://www.ibm.com/developerworks/cloud/library/cl-blockchain-for-cognitive-iot-apps-trs/index.html> (14 January 2017).

[5] Wood, G., "Ethereum: A Secure Decentralised Generalised Transaction Ledger," Ethereum Project, 2014, <http://gavwood.com/paper.pdf> (10 October 2016).

[6] Popov, S., "The tangle," IOTA, 3 April 2016, <https://iotatoken.com/IOTA_Whitepaper.pdf> (8 June 2016).

[7] Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K., "Where Is Current Research on Blockchain Technology?—A Systematic Review," PloS one, 11(10), 1-27 (2016).

[8] Christidis, K. and Devetsikiotis, M., "Blockchains and Smart Contracts for the Internet of Things," IEEE Access 4(1), 2292-2303 (2016).

[9] Nakamoto, S., "Bitcoin: A peer-to-peer electronic cash system," *Consulted* 1(2016), 28 (2008).

[10] Merkle, R.C., "Protocols for public key cryptosystems," Proc. IEEE Computer Society 1980 Symposium on Security and Privacy, 122-133 (1980).

[11] Natoli, C. and Gramoli, V., "The Blockchain Anomaly," Proc. IEEE 15th International Symposium on Network Computing and Applications (NCA), 310-317 (2016).

[12] Frantz, C. K. and Nowostawski, M., "From institutions to code: Towards automated generation of smart contracts," Proc. IEEE International Workshops on Foundations and Applications of Self* Systems, 210-215 (2016).

[13] Seijas, P. L., Thompson, S. and McAdams, D., "Scripting smart contracts for distributed ledger technology," Cryptology ePrint Archive, 2016(1156), 1-28 (2016).

[14] Atzei, N., Bartoletti, M. and Cimoli, T., "A survey of attacks on Ethereum smart contracts," Cryptology ePrint Archive, 2016(1007), 1-24 (2016).

[15] O'Hara, K., "Authority Printed Upon Emptiness," IEEE Internet Computing, 19(6), 72-76 (2015).

[16] Cachin, C., "Architecture of the Hyperledger blockchain fabric," Proc. Workshop on Distributed Cryptocurrencies and Consensus Ledgers, 1-4 (2016).

[17] Douceur, J. R., "The sybil attack," Proc. International Workshop on Peer-to-Peer Systems, 251-260 (2002).

[18] Frantz, C.K. and Nowostawski, M., "From Institutions to Code: Towards Automated Generation of Smart Contracts," Proc. 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems, 210-215 (2016)