

Article

A Survey of Consortium Blockchain and Its Applications

Xiaotong Chen^{1,2}, Songlin He^{1,2}, Linfu Sun^{1,2}, Yangxin Zheng^{1,2} and Chase Q. Wu^{3,*}

¹ School of Computing and Artificial Intelligence, Southwest Jiaotong University (SWJTU), Chengdu 610031, China; c0672@my.swjtu.edu.cn (X.C.); sohe@swjtu.edu.cn (S.H.); sunlf@vip.163.com (L.S.); zhengyx555@163.com (Y.Z.)

² Sichuan Provincial Key Laboratory of Manufacturing Industry Chain Collaboration and Information Support Technology, Southwest Jiaotong University, Chengdu 610031, China

³ Department of Computer Science, New Jersey Institute of Technology (NJIT), Newark, NJ 07102, USA

* Correspondence: chase.wu@njit.edu

Abstract: Blockchain is a revolutionary technology that has reshaped the trust model among mutually distrustful peers in a distributed network. While blockchain is well-known for its initial usage in a public manner, such as the cryptocurrency of Bitcoin, consortium blockchain, which requires authentication of all involved participants, has also been widely adopted in various domains. Nevertheless, there is a lack of comprehensive study of consortium blockchain in terms of its architecture design, consensus mechanisms, comparative performance, etc. In this study, we aim to fill this gap by surveying the most popular consortium blockchain platforms and assessing their core designs in a layered fashion. Particularly, Byzantine fault tolerant (BFT) state machine replication (SMR) is introduced to act as a basic computational model of consortium blockchain. Then the consortium blockchain is split into the hardware layer, layer-0 (network layer), layer-I (data layer, consensus layer and contract layer), layer-II protocols, and application layer. Each layer is presented with closely related discussion and analysis. Furthermore, with the extraction of the core functionalities, i.e., robust storage and guaranteed execution, that a consortium blockchain can provide, several typical consortium blockchain-empowered decentralized application scenarios are introduced. With these thorough studies and analyses, this work aims to systematize the knowledge dispersed in the consortium blockchain, highlight the unsolved challenges, and also indicate the propitious avenues of future work.



Citation: Chen, X.; He, S.; Sun, L.; Zheng, Y.; Wu, C.Q. A Survey of Consortium Blockchain and Its Applications. *Cryptography* **2024**, *8*, 12. <https://doi.org/10.3390/cryptography8020012>

Academic Editors: Mohsen Toorani
and Josef Pieprzyk

Received: 31 December 2023

Revised: 11 March 2024

Accepted: 16 March 2024

Published: 22 March 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Blockchain has emerged and revolutionized the conventional trust model amongst mutually untrusted users in a decentralized network. As an infrastructure technology, the applications of blockchain are far beyond its initial usage in cryptocurrency [1]. Complementary to the message transmission network built by the Internet, blockchain plays the critical role of value transmission network atop the widely used Internet. Until now, the advancement of blockchain has mainly experienced three stages [2], i.e., stage 1.0 initialized with cryptocurrency is represented by Bitcoin [3,4], stage 2.0 in the financial domain is represented by Ethereum [5], which supports Turing-complete programming languages to execute pre-agreed logic via smart contract, the ongoing stage 3.0 refers to various application fields including decentralized finance (Defi) [6], Internet of Things (IoT) [7], cyber security [8], content delivery networks [9], healthcare [10], smart city [11], metaverse [12], etc., which meet diversified and more complex real-world demands. With the blueprint and the gradual stepping into web3.0 [13] where data with semantic meanings are interconnected in a decentralized manner, blockchain sustainably shows its great potential in reshaping trust amongst individuals, and therefore, it is worth retrospectively investigating its core functionalities and primary applications.

Blockchain can be defined as an immutable ledger to record transactions and is maintained by distrustful peer nodes in a distributed network [14]. The wide adoption of blockchain is essentially owed to its multiple advantageous security properties [14,15]. Specifically, the *availability* property ensures that the blockchain network stays available even though partial nodes become unreachable; the *immutability* property guarantees that the recorded transactions cannot be reverted assuming the number of simultaneously corrupted nodes is upper-bounded. This *consistency* property assures that all peer nodes in the blockchain network remain a globally consistent ledger upon invocation, and the *accountability* property enables to take some corresponding actions, e.g., monetary punishment, if any peer performs malicious activities; the *provenance* property indicates that blockchain provides tamper-proof information about the origin of data records. Several studies [16–18] extract the core functions of a blockchain and formalize the blockchain model of cryptography. By constructing the *ideal functionality*, the cryptographic blockchain model is built in the Generalized Universal Composability (UC) framework [19], yielding the essential result of “fair Multi-Party Computation (MPC) with public deposits” aiming to comprehensively specify and reason about the security of blockchain-empowered protocols, and facilitate the designing of decentralized applications atop blockchains. As illustrated in Figure 1, $\mathcal{F}_{blockchain}[\text{succ}]$ defines a general-purpose append-only ledger implemented by common blockchain protocols [20].

$\mathcal{F}_{blockchain}[\text{succ}]$
1. Parameter: successor relationship $\text{succ} : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}$
2. Upon receiving (“init”): $\text{Storage} := \emptyset$
3. Upon receiving (“read”, id): output $\text{Storage}[\text{id}]$, or \perp if not found
4. Upon receiving (“write”, id , inp) from \mathcal{P} :
5. let $\text{val} := \text{Storage}[\text{id}]$, set to \perp if not found
6. if $\text{succ}(\text{val}, \text{inp}) = 1$ then $\text{Storage}[\text{id}] := \text{val} \parallel (\text{inp}, \mathcal{P})$; output (“receipt”, id)
7. else output (“reject”, id)
8. Upon receiving (“ \in ”, id , val):
9. if $\text{val} \in \text{Storage}[\text{id}]$ then output true else output false

Figure 1. Blockchain ideal functionality [20]. The parameter succ models the validity check of a transaction, \mathcal{P} refers to a client, and inp means the input data submitted by \mathcal{P} .

One primary thinking in designing a blockchain-based decentralized application lies in choosing a proper type of blockchain [8] in light of the concrete application setting. Specifically, there are usually three categories: the *public/permissionless* blockchain allows anyone to join in or leave at their discretion; the permissioned *consortium* blockchain requires the participants to be authorized before accessing the blockchain network and the servers constructing the blockchain network are provided by multiple organizations; the permissioned *private* blockchain differentiates from the consortium blockchain in that the peers constructing the blockchain network belong to one organization. In practice, consortium blockchains enable participants of common interests to authenticate and collaborate, thereby reducing the trust risk to some degree since the peers are known to each other. Moreover, consortium blockchains provide greater control over the governance and decision-making processes than public blockchains. Hence, this study examines the consortium blockchain and demonstrates the potential for enhancing various existing applications.

To comprehensively survey the consortium blockchain and its applications, we first provide a layered architecture, as depicted in Figure 2, and elaborate on the core designs and comparisons in each layer. In particular, the bottom layer is hardware which involves traditional physical servers, switches, routers and trusted hardware. The network layer (referred to as *layer 0*) considers the distributed setting where communication models,

services, and typical algorithms are investigated. *Layer I* relates to the blockchain itself, containing a data layer, a consensus layer and a smart contract layer. The *layer II* protocols mainly function to improve the scalability and performance of layer I. Based on these layers, blockchain empowers a wide range of applications. Such a layered architecture of blockchain works as a main guide to survey related techniques in later sections.

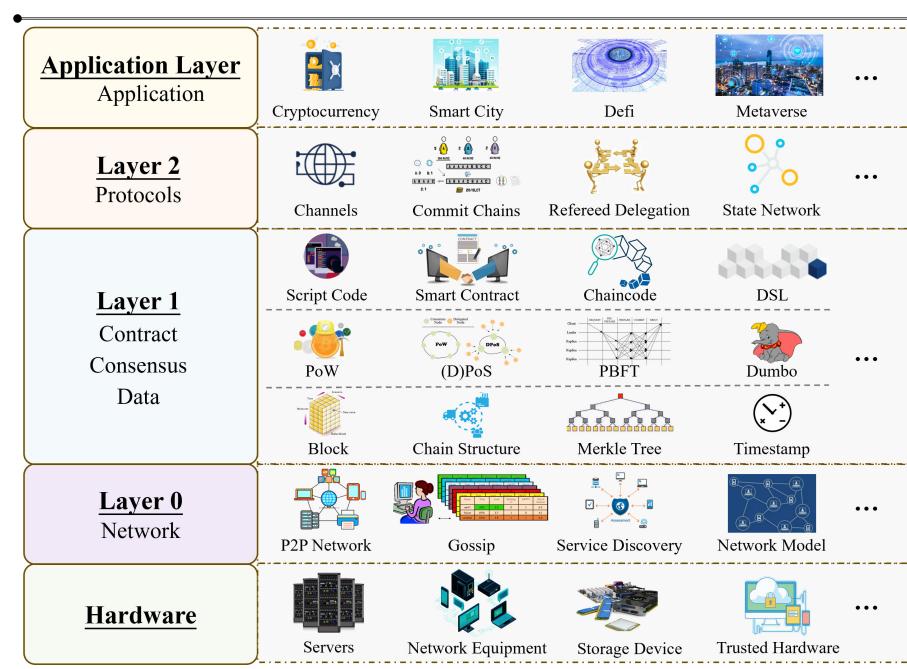


Figure 2. Consortium blockchain relevant layers.

Contributions. Overall, our contributions can be summarized as follows.

- We survey the consortium blockchain and present its core in a layered manner, thus aiming to comprehensively cover pertinent studies.
- We summarize the academic advancements and the usage in practical applications of consortium blockchain and suggest a few potential future research directions in this field for enhancing its design and practicalization.

The rest of the paper is structured as follows: Section 2 concretely presents the consortium blockchain regarding its computational model, typical platforms and relevant layers; Section 3 introduces the closely related decentralized applications atop consortium blockchain; Section 4 provides reflections about consortium blockchain and its potential research directions and we conclude in Section 5.

2. Consortium Blockchain

In this section, we introduce the computational model of a consortium blockchain and popular consortium blockchain platforms and then present the consortium blockchain in a layered fashion including the hardware layer, network layer, layer I and layer II protocols.

2.1. State Machine Replication

As a fundamental distributed computing model, *state machine replication* (SMR) [21] aims to provide an abstract state machine distributed over the network and replicated by many *peers* (or called *replicas*) [22]. In particular, as illustrated in Figure 3, an SMR protocol starts with an *initial state* S^0 , every *client* (or called *process*) can submit a request containing several consecutive execution *commands*, e.g., $C = \{c_1, c_2, c_3\}$ and perceive a sequence of *commits*. Each commit refers to the execution state that is produced via the execution of the commands atop its previous commit. Obviously, as the execution process is deterministic, i.e., with the same input, an algorithm would always generate the same output, and every

peer node would reach the new identical state. From a global viewpoint, all the replicas transfer their state S^0 to S^1 . It is worth pointing out that invalid commands would be rejected during execution. Typically a SMR protocol meets the following requirements [23]: (i) any honest replica r_i starts with the state S^0 ; (ii) The honest replicas r_1, r_2, \dots applying the same sequence of execution operations $\{c_1, c_2, \dots\}$ would all reach the new state S^j , $j \in \{1, 2, \dots\}$. Such a process satisfies two key security properties: (i) *liveness*, every submitted valid command would be added to the commit; (ii) *consistency*, all peer nodes perceive the same commits even though there is a communication delay.

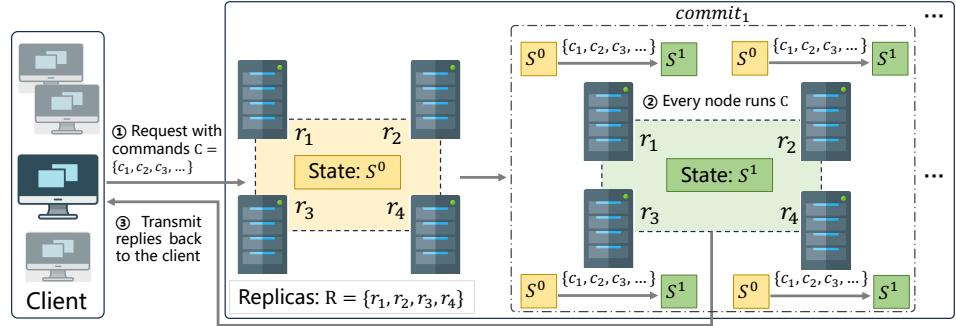


Figure 3. Illustration of state machine replication model.

Blockchain is essentially a state machine where transactions are submitted by peer nodes in the blockchain network and the distributed ledger records the transactions and the latest state upon transaction execution in a consistent and finalized manner. In a consortium blockchain, the distributed SMR ensures consensus, fault tolerance and fairness. For instance, guaranteeing consensus can be realized by propagating the status of a contract to all nodes in the network. Furthermore, SMR in the blockchain setting is highly pertinent to Byzantine fault tolerance (BFT), i.e., the peers can reach consensus even in the presence of adversaries who may corrupt or control part of the replicas. Typical BFT SMR protocols include Practical Byzantine Fault Tolerance (PBFT) [24] and its descendants [25–32]. However, there are several subtle but important differences [23] between the BFT SMR approach and (consortium) blockchain: (i) blockchain applications require maintaining a verifiable persistent ledger containing the executed transactions and also supporting reconfigurations on the replicas; these two features are not present in the SMR implementations; (ii) though most literature about BFT SMR assumes a static set of processes, in a blockchain consortium, peer nodes are expected to be allowed to join and leave at any time, without the need for an additional trusted party.

2.2. Consortium Blockchain Platforms

Choosing a proper consortium blockchain platform can greatly improve the efficiency and practicality of a blockchain-enabled application. To this end, we first analyze the core functionalities of a consortium blockchain and present the designs of each layer with detailed analysis. We select the representative consortium blockchain platforms based on several relevant sources [33–37], which indicate the popularity and practicality of these platforms in various domains.

2.2.1. Core Functionalities of Consortium Blockchain

Consortium Blockchain-Enabled Robust Storage. One core functionality empowered by consortium blockchain lies in providing a robust storage capability. From such a perspective, it resembles a traditional distributed database which has been well-studied for decades. However, there are several key [8,38] differences between a consortium blockchain and a database system:

- *Trust model.* Traditional database systems require trusting all participants where even the malicious behaviors occurring from only one node can make the whole

- system collapse; a blockchain system can tolerate partial, e.g., one-third of the nodes misbehaving arbitrarily. Essentially, a consortium blockchain-enabled storage system supports more robust byzantine fault tolerance instead of merely crash fault tolerance.
- *Transaction processing.* Though most consortium blockchain systems also support parallel transaction processing capability, they differ from the database systems in several aspects: (i) consortium blockchain commits transactions at the block level while the database commits at the individual transaction level; (ii) consortium blockchain is a distributed system where the state is fully replicated across the network and the transaction operates on all nodes, while for a database system, the transactions usually operate on a subset of network nodes; (iii) during transaction execution, the state on different nodes may differ in a consortium blockchain network, while for a database, a transaction is executed once against the only state present in the system; (iv) the factors dominating the performance of a consortium blockchain and a database is distinct, i.e., cryptographic primitive computations, network communication for the former, and the locking mechanism of concurrency control for the latter.

It is strongly believed that transitioning the technology of storage from a traditional database system to a consortium blockchain is the right direction, and we have seen several works [38,39] that are dedicated to research in such a field. Consortium blockchains indeed possess the potential to improve storage robustness. However, the storage cost is huge as all nodes need to store a full copy of the data submitted to the consortium blockchain network. A group of work also concentrated on reducing the on-chain storage costs, as tabulated in Table 1.

Table 1. The methods for reducing consortium blockchain on-chain storage costs.

Methods	Literature	Highlights	Year
Sharding	Wang et al. [40]	Sharding Technology	2023
	Wu et al. [41]	KBFT	2023
	Shen et al. [42]	A Node Reliable Shard Model	2023
	Zheng et al. [43]	Replay-epoch & Cross-call	2022
	Zhou. et al. [44]	Dynamic Sharding	2020
	Qi et al. [45]	Erasure Coding	2020
Compression	Farahat et al. [46]	The LZ4 Algorithm	2023
	Karthik et al. [47]	Lempel-Ziv-Welch	2023
	Yu et al. [48]	PoW-BC	2021
Deduplication	Liu et al. [49]	A New Refreshable Encryption Algorithm	2022
Compressed Indexing	Zhou et al. [50]	Merkle Semantic Trie	2023
	Chen. et al. [51]	Index Pointers	2023
	Shafarenko [52]	Tunstall's Method	2022

Consortium Blockchain-Enabled Guaranteed Computing. Another core functionality empowered by consortium blockchain is the capability of guaranteed execution. Underpinned by the smart contract which can execute pre-determined programs without any interruption, consortium blockchain can faithfully execute any deployed executable code, e.g., business agreement [53], and thus provide the great potential to revolutionize many existing centralized applications [16].

2.2.2. Hyperledger Fabric

Hyperledger Fabric [54] is a popular open-source consortium blockchain system introduced by IBM. The typical message flow in Hyperledger Fabric is depicted in Figure 4, where a client submits transactions to a group of peer nodes called *endorsing* nodes or *endorsers*, which simulate the transaction execution and send back the endorsed results, then the signed/endorsed results are forwarded to the *ordering* nodes or *orderers*, which accumulate transactions into blocks and output blocks abiding by certain rules, e.g., following a certain time interval or an approximate block size. The generated blocks are sent

to *committing nodes* or *committers*, which would write to the distributed ledger upon the validation of the generated blocks pass. Note, that the orderers can either be the full nodes in the blockchain network or extra servers can be employed. Surrounding Hyperledger Fabric, research mainly focusing on the following aspects is conducted: (i) performance modeling and analysis [55–57]; (ii) privacy and security [58–60]; (iii) interoperability [34,61]; (iv) benchmark and visualization [57,62,63].

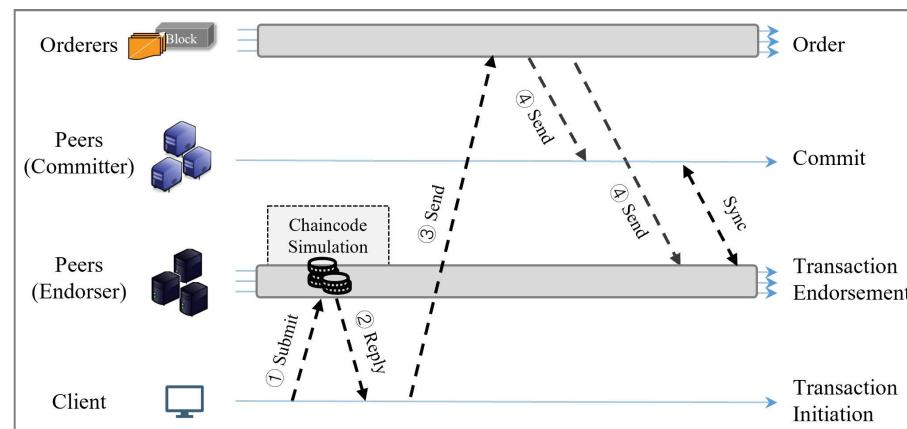


Figure 4. A typical transaction workflow in Hyperledger Fabric.

2.2.3. Ethereum

Ethereum [5] is blockchain 2.0 since it supports the Turing-complete smart contract, which enables the execution of much more complex logic in various application scenarios beyond cryptocurrency. In contrast to the unspent transaction output (UTXO) model in Bitcoin, Ethereum is designed with the account-based model where the latest balance for a user equipping with a public-private key pair is bound with his/her address (hash of the user's public key). In addition, Ethereum has transitioned its consensus mechanism from Proof of Work (PoW) to Proof of Stake (PoS) in 2022, i.e., the so-called *Merge* [64], to significantly reduce the criticized energy cost. Ethereum is usually characterized by three features: (i) smart contract [65]; (ii) Ethereum virtual machine (EVM) [66] and (iii) decentralized applications (DAPPs) [67]. Apart from the wide application in public settings, Ethereum can also be deployed in a consortium [68] or private [69] manner. It is worth pointing out that Ethereum is popularized as a public blockchain protocol, yet it also supports the customization of building a consortium blockchain environment for diversified application scenarios [70,71].

2.2.4. FISCO BCOS

FISCO BCOS [36] is an enterprise-grade permissioned blockchain system developed by the Financial Blockchain Shenzhen Consortium which is compatible with Ethereum in terms of account management and smart contracts [72]. FISCO BCOS aims to significantly improve the consortium blockchain's performance. Such an ambition is mainly achieved via two designs: a block level pipelining workflow designed to break the serial dependency of blocks; the blocks are processed in a pipeline with four stages. Blocks can be processed at different stages simultaneously. Meanwhile, a deterministic multi-contract mechanism is designed to execute transactions in parallel within a block. Transactions are also dispatched into multiple shards and processed in parallel by a set of executors. Consequently, FISCO BCOS realizes both intra-block and inter-block parallelism, and therefore, enables considerable performance and scalability enhancement [33].

2.2.5. Corda

Corda [73] is a distributed ledger platform designed specifically for the financial sector. It is utilized by over 60 companies, banks such as HSBC, J.P. Morgan, and institutions like

Intel, Microsoft and NASDAQ [74]. There exist several key differences [75] between Corda and conventional blockchains: (i) there are no transactions structured in blocks; (ii) the occurred transactions are only shared with involved parties instead of all participants; (iii) a set of trusted parties called notaries are introduced in order to prevent a double-spending attack. In light of these characteristics, Corda highlights three key properties:

- *Privacy.* Corda prioritizes privacy by design, ensuring that transactions are only shared on a need-to-know basis. Unlike many other blockchains, Corda achieves a weaker security notion *partial consistency* where parties in the blockchain network may only see part of the state but accumulating all parties' states can result in the global view. This minimizes the exposure of sensitive data and reduces the network load and storage requirements [35].
- *Scalability.* Unlike many other blockchains, Corda does not rely on a global consensus mechanism that requires every node to validate every transaction but instead utilizes a pluggable notary service that can employ various consensus algorithms depending on the use case. Such a design allows Corda to handle high transaction volumes and complex business logic without compromising performance or security [76].
- *Interoperability.* Corda allows businesses to use their legacy infrastructure while benefiting from the advantages of distributed ledger technology. It also supports interoperability among different Corda networks, as well as with other blockchain platforms via the use of common standards and protocols. Such a design enables cross-industry and cross-border collaboration and innovation [77].

2.2.6. Quorum

Quorum [78] is an Ethereum-based, enterprise-focused, permissioned blockchain infrastructure specifically designed for financial use cases [37]. This open-source project was initiated by J.P. Morgan Chase and has been acquired by ConsenSys. Quorum contains two blockchain projects: one is based on GoQuorum [79], and the other one is based on Hyperledger Besu [80]. Each Quorum node is composed of two main services: (i) *Quorum client*, which is responsible for executing the Ethereum p2p protocol and the consensus algorithm; (ii) *privacy manager*, which enables private transactions and smart contract operations. As a consortium blockchain, Quorum is mainly introduced to satisfy the following demands [81]: (i) empowered by the privacy manager to execute private transactions and smart contract operations; (ii) supporting multiple pluggable consensus mechanisms; (iii) enabling flexible and expressive network permissions management.

2.2.7. Ripple

Ripple [82] is a real-time gross settlement (RTGS) system aiming at fast global payments, asset exchange, and settlement [83]. Ripple maintains a ledger of transactions where participants can trade user-issued currencies along with the native cryptocurrency of Ripple, i.e., XRP. The *Ripple protocol consensus algorithm* (RPCA) allows for a flexible security assumption for consensus protocols (typically tolerating $< 1/3$ faulty nodes) in the sense that each node can declare which nodes it trusts instead of using a global assumption about how many faulty nodes may collude simultaneously and misbehave. In practical usage, a company that wants to employ the Ripple network can develop software and interact with it, e.g., SAP developed a Ripple-enabled application for cross-border payments between two banks which significantly decreased time costs, i.e., from six business days to only 20 s [84].

Table 2 presents the comparison of the aforementioned consortium blockchain platforms in terms of their data models, supported consensus mechanisms, state databases, highlighted properties and performance under specific experimental environments.

Table 2. The comparison of different consortium blockchain platforms.

Platform	Data Model	Consensus	State Database	Highlights	Performance Tested Net.	TPS (tx/s)
Hyperledger Fabric [54]	Account Based	Raft, PBFT	CouchDB, LevelDB	Pluggable Consensus, Scalability	4 nodes	~3000
Ethereum [85]	Account Based	PoW, PoS	MPT	Turing-Complete Smart Contract	10 nodes	~6000
FISCO BCOS [33]	Account Based	Raft, PBFT	LevelDB	Efficiency, Flexibility	6 nodes	~3000
Corda [86]	UTXO Based	Raft, PBFT	H2	Privacy, Scalability	4 node	~2500
Quorum [87]	Account Based	Raft, PBFT	Go-Ethereum	Modularity, Privacy	3 nodes	~2000
Ripple [88]	Ripple Data Model	XRP	Rippled Database	RTGS, Native Token	16 nodes	~1000

2.3. Hardware Layer

The hardware layer refers to the underlying hardware equipment and infrastructure, including computers, servers, network equipment and so forth. Especially, trusted hardware provides a *trusted execution environment* (TEE) [89], i.e., a hardware architecture that enables code execution in an isolated, tamper-free environment (called a secure *enclave*) [90]. A secret key hidden in the enclave and possibly only known to the hardware manufacturer can be used to encrypt incoming and outgoing data, thus ensuring data confidentiality. Meanwhile, TEE can attest that an output represents the result of code execution, and allows remote users to make sure the execution is correct. In recent years, TEEs designed with a secure counter and supporting complex stateful functions are more preferred, popular products including Intel SGX [91], Intel TXT [92], ARM's TrustZone [93], AMD SEV [94], Sanctum [95], KeyStone [96], etc. TEEs exhibit a set of security features [97,98], and therefore, are adopted in many complicated system designs [99]. However, TEEs are also vulnerable to a few attacks, e.g., side-channel attacks [100] and rewind attacks [101], and plenty of work [98,102–104] focuses on such a direction to defend against potential attacks. The hardware layer provides the physical resources and support required for the operation of the consortium blockchain systems and plays a crucial role in affecting the performance, security and reliability.

2.4. Network Layer

The network layer not only includes the complete network stack of the conventional network architecture which concentrates on Internet routing, but it also forms a dedicated peer-to-peer network for blockchain nodes. This layer is of great importance since it impacts the scalability [105], security [106] and privacy [107] of a blockchain network. In a consortium blockchain network, full nodes are required to participate in consensus, and also periodically communicate with each other to maintain the connection [108], e.g., via gossip protocol [7]. When designing consortium blockchain-enabled decentralized applications, it is necessary to consider different communication models, e.g., *synchronous*, *partial synchronous* and *asynchronous*. A main assumption regarding the network layer is that such a layer should provide reliable communication among peer nodes in a blockchain network.

2.5. Layer I: Data, Consensus Mechanism, and Smart Contract

Layer I refers to consortium blockchain *per se* (As illustrated in Figure 2, layer-I refers to the design of the consortium blockchain itself regardless of the underlying hardware, network environment, or other protocols that are designed atop consortium blockchain). It hosts an append-only chain of blocks that accumulate transactions in the blockchain network for public verifiability [109]. It can further be categorized into the sub-layers, i.e., the datalayer, consensus layer and contract layer.

2.5.1. Data Layer

The data layer mainly concerns the data structure and data storage. Specifically, blockchain is typically a succession of blocks with a starting block called *genesis block*. Transactions in each block can simply be a transaction list, e.g., in Bitcoin, or a more intricate structure such as state trie in Ethereum. For the consortium blockchain, we highlight the following components in the data layer.

- **Block.** A block contains two parts, i.e., the *block head* and the *block body*, where the head part typically includes the block version, the merkle root of the involved transactions, timestamp, nonce and the hash of the previous block. The body part is mainly composed of a transaction counter and a bunch of transactions. The number of transactions is related to the block size, which is restricted due to the communication overhead. Meanwhile, asymmetric cryptography, i.e., digital signatures such as the *elliptic curve digital signature algorithm* (ECDSA) [110] is used to ensure the validity of transactions, where usually the digital signature requires *existential unforgeability under chosen message attack* (EU-CMA) security [111].
- **Chain Structure.** From the data structure viewpoint, the architecture of a consortium blockchain is essentially a hash chain where the unique hash value of each block is computed based on its previous one. Such a design fully hinges on the security properties such as one-way, collision resistance of the hash function $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^\lambda$ with the security parameter λ , and can further be modeled as a random oracle [112]. In addition, to improve the scalability and reduce the latency of the hash-chain based blockchain system, some works [113–115] have explored the *directed acyclic graphs* (DAG) architecture of blockchain, exemplified by IOTA [116], an open source distributed ledger designed for the IoT.
- **Merkle Tree.** A Merkle tree MT is constructed from the leaf nodes level all the way up to the Merkle root level by grouping nodes in pairs and calculating the hash of each pair of nodes in that particular level [117]. Specifically, the MT scheme contains a tuple of three algorithms (BuildMT, GenMTP, VerifyMTP), as illustrated in Algorithms 1–3. BuildMT accepts as input a sequence of elements $m := (m_1, m_2, \dots, m_n)$, and outputs the Merkle tree MT with root. GenMTP takes as input the Merkle tree MT and the hash of the i -th message in m , i.e., $\mathcal{H}(m_i)$, and outputs a proof π_i to attest the inclusion of m_i at the position i of m . VerifyMTP takes as input the Merkle tree proof π_i , the root of MT and the message hash $\mathcal{H}(m_i)$, and outputs either *true* or *false* indicating whether it succeeds in verifying or not. The security of the Merkle tree scheme ensures that: for any *probabilistic polynomial-time* (P.P.T.) adversary \mathcal{A} , any sequence m and any index i , conditioned on MT is a Merkle tree built for m , \mathcal{A} cannot produce a fake Merkle tree proof fooling VerifyMTP to accept $m'_i \neq m_i \in m$ except with negligible probability given m , MT and security parameters. For a consortium blockchain, its advantage lies in allowing efficient comparison and verification of transactions with viable computational power.

Algorithm 1 BuildMT Algorithm

```

1: Input :  $m = (m_1, \dots, m_n)$ 
2: Output : Merkle tree MT with root
3: if  $|m| = 1$  then
4:    $label(root) = \mathcal{H}(m_1)$ 
5: else
6:    $lchild = \text{BuildMT}(m_1, \dots, m_{\lceil n/2 \rceil})$ 
7:    $rchild = \text{BuildMT}(m_{\lceil n/2 \rceil + 1}, \dots, m_n)$ 
8:    $label(root) = \mathcal{H}(root(lchild) || root(rchild))$ 
9: end if
10: return Merkle tree MT with root
  
```

Algorithm 2 GenMTP Algorithm

```

1: Input : MT,  $\mathcal{H}(m_i)$ 
2: Output : Merkle tree proof  $\pi_i$ 
3: while  $\mathcal{H}(m_i) \neq \text{label}(\text{root}(\text{MT}))$  do
4:    $lchild \leftarrow \mathcal{H}(m_i).\text{parent}.lchild$ 
5:    $rchild \leftarrow \mathcal{H}(m_i).\text{parent}.rchild$ 
6:   if  $\mathcal{H}(m_i) = lchild$  then
7:      $b_j \leftarrow 0, l_j = \text{label}(rchild)$ 
8:   else
9:      $b_j \leftarrow 1, l_j = \text{label}(lchild)$ 
10:  end if
11:   $\mathcal{H}(m_i) \leftarrow \mathcal{H}(m_i).\text{parent}$ 
12: end while
13: return  $\pi_i = ((l_j, b_j))_{j \in [1, n]}$ 

```

Algorithm 3 VerifyMTP Algorithm

```

1: Input : root(MT),  $\pi_i, \mathcal{H}(m_i)$ 
2: Output : true or false
3: parse  $\pi_i$  as a list  $((l_j, b_j))_{j \in [1, n]}$ , where  $l_j$  is a node label,  $b_j$  is a binary bit
4: for  $j$  in  $[n]$  do
5:   if  $b_j == 0$  then
6:      $\mathcal{H}(m_i) \leftarrow \mathcal{H}(\mathcal{H}(m_i) || l_j)$ 
7:   else
8:      $\mathcal{H}(m_i) \leftarrow \mathcal{H}(l_j || \mathcal{H}(m_i))$ 
9:   end if
10:  end for
11:  if  $\mathcal{H}(m_i) \neq \text{label}(\text{root})$  then
12:    return false
13:  else
14:    return true
15:  end if

```

2.5.2. Consensus Mechanism

The consensus mechanism lies at the heart of a consortium blockchain network, ensuring that all the peer nodes reach the same state. Typically, a consensus mechanism satisfies three key properties: (i) *termination*, each peer node locally outputs the result within a limited amount of time; (ii) *agreement*, all honest peer nodes agree on the same value; (iii) *safety*, the agreed value for all honest peer nodes comes from an honest node. There are multiple perspectives to categorize different consensus mechanisms, e.g., crash fault tolerance (CFT) vs. byzantine fault tolerance (BFT), and different communication models, e.g., synchronous, partial synchronous and asynchronous. We direct readers to related surveys in Table 3 and compare several commonly used consensus mechanisms for consortium blockchains in Table 4.

Table 3. Related surveys involving consensus mechanisms (Fully: ✓, Partial: *, Not Applicable: ×).

Surveys	Consensus Comparison	Consortium Blockchain-Specific	Year
Du et al. [118]	✓	*	2017
Nguyen et al. [119]	✓	*	2018
Alsunaidi et al. [120]	✓	*	2020
Fu et al. [121]	✓	*	2020

Table 3. Cont.

Surveys	Consensus Comparison	Consortium Blockchain-Specific	Year
Wan et al. [122]	✓	*	2020
Ferdous et al. [123]	✓	✗	2021
Lashkari et al. [124]	✓	✓	2021
Bouraga [125]	✓	*	2021
Divya et al. [126]	✓	*	2021
Khan et al. [127]	✓	*	2022
Yao et al. [128]	✓	✓	2023
Guru et al. [129]	✗	✗	2023
Morais et al. [130]	✓	*	2023

Table 4. Consensus comparison (○: High; ●: Mid; ⊙: Low; →: Transition through timeline; n: the number of nodes in blockchain network).

Consensus Mechanisms	Supportive Blockchain Type	Safety	Scalability	Costs	Tolerance Threshold
Raft [130,131]	Consortium	●	○	●	n/2
PoS [132]	Consortium, Public	○	○	●→●	n/2
PoA [133]	Consortium, Private	●→○	●	●	-
PoET [132]	Consortium, Private	●	○	●	n/2
PoC [134]	Consortium, Public	○	○	●	-
DPoS [135]	Consortium, Public	●	○	●	n/2
FBA [136]	Consortium, Private	○	○	●	-
PBFT [32]	Consortium	○	●	●	n/3
RBFT [136]	Consortium	○	●	●	n/3
BFT-SMART [136]	Consortium	○	●	●→○	n/3
RPCA [136]	Consortium	○	○	●	n/5
SCP [136]	Consortium, Public	○	○	●	n/3
HotStuff [136]	Consortium, Private	○	○	●	n/3
Tendermint [137]	Consortium, Public	○	○	●	n/2
HoneyBadger [138]	Consortium	○	○	●	n/2
Dumbo [139]	Public, Consortium, Private	○	○	●	n/3

2.5.3. Smart Contract

The term smart contract was popularized by Szabo in a 1994 essay [140]. In consortium blockchain, the smart contract refers to a piece of program that is pre-determined by involved parties, and the logic would be automatically executed without any interruption. When designing a consortium blockchain-enabled application, the Turing complete smart contract can be defined as a stateful ideal functionality [16], i.e., a stateful program that can transparently handle pre-specified functionalities and access the cryptocurrency ledger to faithfully tackle conditional payments once a certain event is triggered.

2.6. Layer II Protocols for Scalability

A consortium blockchain, as a distributed system, faces the critical issue of scalability. To overcome such an issue, a bevy of works focus on the layer-I, i.e., the blockchain

itself, via designing alternative consensus mechanisms [141] or adopting techniques such as sharding [142] and side-chains [143]. However, these layer-I solutions possess shortcomings, e.g., adopting new consensus means changing the core part of a blockchain network, leading to blockchain forking and making the blockchain system backward incompatible. Similarly, sharding implies significant changes in existing blockchain architecture, which seriously hinders its usage in practice.

Layer II protocols are meant to increase the scalability of the underlying blockchain network, thus considerably improving the performance without modifying anything in layer-I. The rationale behind the Layer-II protocols lies in enabling users to perform transactions *off-chain* via private and authenticated communication. Particularly, layer-II protocols can be divided into three categories: (i) *channels*, which establish a private p2p medium governed by pre-agreed rules that deployed as smart contracts, and allow users to consent to state updates with state transitions exchanged off-chain [109]. Channels can further be categorized as *state channel* [144] and *payment channel* [145,146], where the former is a generalized version and the latter is specific to payment-oriented applications; (ii) *commit chains*, where an operator can launch a commit-chain and users can join by contacting the operator and submitting transactions. The operator can then periodically submit a commitment to all collected transactions to the parent chain. The typical protocols include NOCUST [147] and Plasma [148]; (iii) *protocols for refereed delegation*, which function to tackle the disputes among participants, and typically include Truebit [149] and Arbitrum [150].

2.7. Performance Modeling for Consortium Blockchain

Improving the performance of consortium blockchain is undoubtedly of utmost importance. To this end, performance evaluation of the system by experiments is necessary. However, such a process is tedious and time-consuming [151]. It is, therefore, desired to design a model that can compute performance metrics as a function of various consortium blockchain system configurations and parameters. It would facilitate the comparison of different configurations and make design trade-off decisions and meanwhile, enabling users to compute performance for potential architectural updates that the software engineers can take into account for future releases. The existing modeling methods, as tabulated in Table 5 for consortium blockchain can be categorized into the following directions:

- **Queuing.** For consortium blockchain, processes like node competition for consensus transaction confirmation and block generation introduce potential issues such as transaction backlogs and congestion, resulting in increased delays and reduced throughput. Queuing theory can establish models considering interactions among nodes, block propagation times, and transaction confirmations, thus facilitating predicting system performance limits and identifying potential optimizations. The works [152–154] leverage queuing theory to model the different stages of Fabric and Ethereum.
- **Stochastic Petri Nets (SPNs).** SPNs offer a graphical representation that can effectively model the intricate interactions among peer nodes in the consortium blockchain network. Also, SPNs are adept at handling randomness and uncertainty, e.g., network latency, and transaction confirmation time. This stochastic capability is particularly powerful, which enables the analysis of blockchain system performance and stability under varying conditions, thereby facilitating system design optimization and resource utilization. The related works include [151,155,156].

Besides, there are other analytical models proposed for analyzing blockchain performance. Papadis et al. [157] propose a stochastic network model to capture the blockchain dynamics and mainly analyze the impact of the block dissemination delay and hashing power of the member nodes on blockchain performance. Li et al. [158] consider the information propagation delays in the blockchain network and propose Athena, a Hyperledger Fabric-based tuning system that can automatically provide parameter configurations for optimal performance.

Table 5. Different performance modeling methods of consortium blockchain.

Methods	Literature	Consensus	Platform	Model Output
Queuing	[152]	Solo, PBFT	Hyperledger Fabric V1.4	Latency
	[153]	Kafka, Raft	Hyperledger Fabric V1.4	Throughput and latency
	[154]	POS	Ethereum	Throughput and memory-pool count
SPNs	[155]	PBFT	Hyperledger Fabric V1.2	Throughput and latency for each phase
	[156]	PBFT	Hyperledger Fabric V1.0	Throughput, utilization and mean queue length for each peer
	[151]	PBFT	Hyperledger Fabric V0.6	Mean Time for Consensus
Others	[157]	POW	Ethereum	TX processing rate
	[158]	Raft	Hyperledger Fabric V1.4	TX throughput and latency

3. Decentralized Applications Atop Consortium Blockchain

As mentioned earlier, the consortium blockchain essentially provides two vital functionalities, i.e., robust storage and guaranteed computing. Hereunder, we describe several popular application scenarios that consortium blockchain empowers.

3.1. Internet of Things

Spawned from the machine-to-machine (M2M) technology, the Internet of Things (IoT) emerges as a new dynamic global network infrastructure with self-configuring capabilities where physical and virtual “things” with identities, physical attributes, and virtual personalities are seamlessly integrated into the information network [7]. According to [159], the number of connected devices will reach 75 billion by the end of 2025. However, such a huge amount of connected devices makes it challenging in terms of the huge amount of collected data, intensive data exchange, security, privacy, centralized processing, and interoperability [160]. To mitigate these issues, (consortium) blockchain has been reckoned as a promising infrastructure. Specifically, (consortium) blockchain along with smart contracts brings advantages in the following aspects: (i) *reducing costs*, a consortium blockchain can replace the traditional multiple centralized parties charging a lot due to their services, thus reducing the intermediate costs [161]; (ii) *establishing trust*, a group of parties of interest can join together and establish a consortium blockchain, which enables the transparency and accountability of occurred activities, and therefore, builds trust among participants [162,163]; (iii) *privacy protection*, blockchain-based public key infrastructure (PKI) allows devices and users to conceal their physical-world identities. By combining with other blockchain-enabled technologies such as decentralized identity [164], privacy can be preserved [165]; (iv) *secure information management*, consortium blockchain-enabled architecture can realize secure data management by, e.g., issuing certificates for devices [166] and robust information storage [8].

3.2. Healthcare

With the wide adoption of the Internet of Medical Things (IoMT), a great deal of personal health data are collected as personal health records (PHRs). The global PHRs market was about 26.8 billion dollars in 2020 with a probable compound annual growth rate to reach 3.7% by 2028 [167]. Correspondingly, the sharing of PHRs becomes an urgent demand since it can help significantly improve the accuracy of diagnosis, and also be beneficial to disease study [168]. However, PHRs sharing is challenging since the data might be manipulated improperly or revealed during the operational process. Thus, protecting the data integrity and confidentiality is a basic requirement. To address these issues, consortium blockchain emerges as a promising way to build trust among involved hospitals and patients. In particular, [167] proposes a security-aware and privacy-preserved PHR management and sharing scheme based on consortium blockchain where IPFS is

involved to store PHR ciphertext, and zero-knowledge proof (ZKP) to provide evidence for verifying keyword index authentication on-chain. The works [168–170] mainly focus on privacy-preserved data sharing and access control based on consortium blockchain.

3.3. Supply Chain

Supply chain [171] is one of the most straightforward application scenarios of blockchain due to the well-known immutability feature that blockchain can provide, e.g., the *IBM Food Trust* project built on Hyperledger Fabric [172]. There are many advantages when incorporating blockchain with a supply chain ecosystem. For instance, supply chain management (SCM) involves the design, planning, and execution of all activities that result in the delivery of a product or service to the end customer; a consortium blockchain can increase the effectiveness and efficiencies of global supply chains by delivering relevant information quickly, securely, and efficiently to all participants in the chain and by facilitating the use of digital tokens to track goods as they move along the procurement, planning, production, and delivery phases of a supply chain [173]. Meanwhile, the consortium blockchain-based *supply chain finance* [174,175] application is a decentralized financial solution that aims to improve the reliability and efficiency of supply chain financial transactions by connecting supply chain participants and financial institutions on a trusted shared ledger. The application leverages the features of consortium blockchain technology, i.e., decentralization, tamper-proof transaction records and smart contracts, to provide more transparent, secure and efficient financial services to all parties in the supply chain.

3.4. Agriculture

Current agricultural advancement and reform are calling for new techniques and innovations to create a more transparent and accountable environment in the agriculture sector [176]. The promising answer lies in blockchain, which can meet the diverse demands in the ecosystem of agricultural products. Existing solutions hinging on a centralized management system suffer from several drawbacks: (i) the centralized server is readily hacked, causing damage to data integrity; (ii) the supply chain management of the agricultural products usually relies on centralized servers, resulting in a single point of failure (SPOF); (iii) high costs are involved to either maintain a set of necessary systems or for a third-party helping to verify and monitor the transactions. To this end, [177] propose a consortium blockchain-based agricultural machinery scheduling system, which optimizes the matching function and scheduling algorithm in the smart contract, and improves the scheduling efficiency. The work [178] proposes a food traceability system based on IoT and blockchain for agricultural products. [70] proposes a consortium blockchain-enabled food trading system, which sets permission for different roles in food transactions and helps choose an optimized trading portfolio for buyers. A set of literature [179–181] investigates the usage of consortium blockchain in the agricultural supply chain. Several works [176,182,183] also review the related techniques, security and privacy challenges and potential research directions in consortium blockchain-enabled agricultural applications.

3.5. Smart Grid

The concept of a smart grid represents a new vision of the traditional power grid that aims to integrate green and renewable energy technologies efficiently [184]. By generating electricity on a small, individual scale and selling it to the grid, it ensures the efficient distribution of electricity, the maintenance of low losses and high quality, and the security of electricity supply [185]. However, challenges such as serious security and privacy issues arise in the adoption of smart grid to consume and trade electricity data [186]. Blockchain technology, which offers a promising solution to these issues, facilitates the following aspects: (i) flexible and integral smart grid data aggregation and regulation [187]; (ii) secure data storage and sharing [188]; (iii) the balance between energy pricing and the amount of traded energy for demand response [189]; (iv) transaction immutability for generators and consumers [185]; (v) privacy preservation for trading users [190,191]; (vi) economic evaluation of blockchain-enabled local energy market [192]; (vii) access

control [193]. The key role of (consortium) blockchain is to act as a trusted third party in a smart grid which is scarce in practice. Furthermore, by integrating cryptographic primitives such as digital signature, encryption algorithms and additional designs such as enhanced consensus mechanisms, blockchain-based solutions can achieve the desired design goals such as reliability, efficiency, flexibility and security in smart grid trading.

4. Challenges and Potential Directions

4.1. Challenges of Consortium Blockchain

We have witnessed the wide adoption of consortium blockchain in many practical settings. Several interesting research directions are still worth further exploration. Hereunder, we highlight the following aspects:

- **Balancing decentralization and performance.** Consortium blockchain-based applications can gain the benefits of being more secure and robust. However, it also introduces extra overhead due to its distributed architecture. As depicted in Figure 5 [194], conventional data centers based on centralized servers can efficiently handle operations. However, the single point of failure issue becomes obvious. The permissionless/public blockchain-enabled systems possess the worst performance but the best robustness. In the middle, the consortium blockchain exhibits better robustness and scalability in comparison with the centralized data centers while having better performance than the fully decentralized public blockchain-enabled systems. Such results follow the so-called blockchain's impossible triangle, i.e., our current technology and understanding are insufficient to ensure *decentralization*, *scalability* and *security* simultaneously. Though various efforts [195–198] have been put to step towards such an ultimate goal, there still needs to be time to reach it; it is desired to consider the concrete demands when designing consortium blockchain-based systems.
- **Consortium blockchain-enabled provably secure protocol designs.** Following the paradigm of modern cryptography [199], it is indispensable to formally argue the security properties of consortium blockchain-based decentralized applications. Specifically, there are three basic principles needed for probably secure protocols, i.e., *formal definitions*, *precise assumptions* and *rigorous proofs*. In addition, in *game-based* security, we claim a protocol is secure if the adversary's advantage is at most negligible considering the security parameter. In the *simulation-based* security, the protocol is secure if the adversary cannot computationally distinguish between the real-world protocol execution and its simulated version of the security experiment in polynomial time. Moreover, another viewpoint for security proof distinguishes the *standalone* and the *universally composable* (UC) model [200], which captures the security of multiple concurrent execution or even composition among multiple secure protocols.

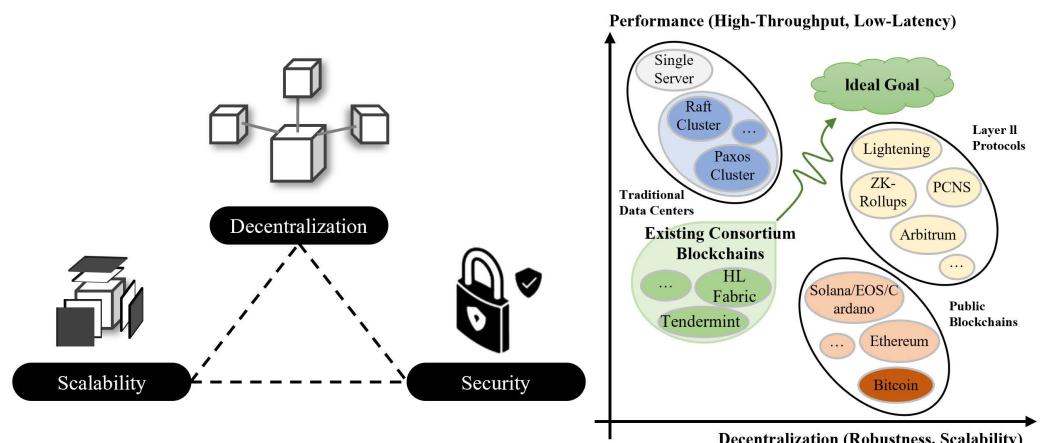


Figure 5. Blockchain's impossible triangle (left) and the balancing between decentralization and performance (right).

4.2. Potential Directions for Consortium Blockchain

For future research directions, there are opportunities in each layer of consortium blockchain that may influence the development of consortium blockchains.

- **TEE-enhanced designs for consortium blockchain.** Combining Trusted Execution Environments (TEEs) with consortium blockchain exhibits great potential in mitigating various security risks and providing significantly improved efficiency [201]. For instance, designing a more efficient consensus mechanism [202] on TEEs instead of wasting time collecting peer nodes' responses during reaching consensus, augmenting the confidentiality of smart contracts [203] for consortium blockchain, or designing more efficient and secure consortium blockchain-enabled applications based on TEEs [204]. However, the vulnerabilities [98] exposed by TEEs still require further exploration and solutions.
- **Layer-II protocols for the scalability of consortium blockchain.** Layer-II protocols undoubtedly play a vital role to improve the scalability of consortium blockchains. However, some open challenges, e.g., quantifying the specific cost of these protocols to offer more rationality in performing layer-II transactions, the quantification of layer-II protocols' decentralization similar to layer-I's decentralization [205], or providing a systematic method to develop security and privacy notions for layer-two protocols, faithfully including their interaction with layer-I, i.e., the consortium blockchain layer [109].
- **Post-quantum consortium blockchain.** The security properties such as transparency, reliability and consistency of consortium blockchains essentially rely on the underlying cryptographic primitives like public-key cryptography and hash functions [206]. However, the quick advancement of quantum computing has exhibited potential while serious security threats for consortium blockchains. To this end, existing consortium blockchains are expected to be post-quantum, quantum-proof, quantum-safe, or quantum-resistant. Though some efforts [207–209] have been witnessed, no widely recognized post-quantum consortium blockchain platforms are found.
- **Practical application-driven designs.** Consortium blockchain can empower the flourishing of diversified decentralized applications, and it is an ongoing topic to develop a killer application in different settings. Thus, it is worth considering the core functionalities of a consortium blockchain and the abstraction of centralized/decentralized applications [16,210]. The potentially interesting goal lies in building a generalized secure and efficient compiler that can seamlessly transmit the existing architectures to a consortium blockchain-based decentralized ones.

5. Conclusions

Consortium blockchain has been widely applied to many practical scenarios such as finance, IoT, cyber security and the metaverse. It provides two core functionalities of secure robust storage and guaranteed computation, thus bringing many advantages, including a more flexible trust model in comparison with traditional infrastructure. We proposed a layered consortium blockchain architecture and surveyed the pertinent technologies in each layer. Furthermore, the challenges in consortium blockchain itself and building consortium blockchain-enabled decentralized applications are discussed, and the potential research directions are also sketched.

Author Contributions: Conceptualization, X.C. and S.H.; methodology, X.C., S.H. and Y.Z.; investigation, X.C. and S.H.; writing—original draft preparation, X.C. and S.H.; writing—review and editing, X.C., S.H., Y.Z. and C.Q.W.; supervision, S.H., L.S. and C.Q.W.; funding acquisition, S.H., L.S. and C.Q.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key R&D Project of China (No. 2023YFB3308400, 2023YFB3308500, and 2023YFB3308600). Songlin He is also supported in part by NSFC (No. 62302403), the Fundamental Research Funds for the Central Universities (No. A0920502052301-186), and the

New Interdisciplinary Cultivation Fund (No. YH15001124322133) with Southwest Jiaotong University, Sichuan, China.

Data Availability Statement: The data is collected from the sources: Elsevier's Library, IEEE Xplore, Multidisciplinary Digital Publishing Institute (MDPI), ACM Digital Library, and Springer.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Defi	Decentralized Finance
IoT	Internet of Things
UC	Universal Composability
MPC	Multi-Party Computation
SMR	State Machine Replication
CFT	Crash Fault Tolerance
PBFT	Practical Byzantine Fault Tolerance
UTXO	Unspent Transaction Output
RTGS	Real-Time Gross Settlement
PoW	Proof of Work
PoS	Proof of Stake
EVM	Ethereum Virtual Machine
DAPP	Decentralization Application
TEE	Trusted Execution Environment
P.P.T.	Probabilistic Polynomial Time
ECDSA	Elliptic Curve Digital Signature Algorithm
EU-CMA	Unforgeability Under Chosen Message Attack
DAG	Directed Acyclic Graphs
SPNs	Stochastic Petri Nets
M2M	Machine to Machine
PHRs	Personal Health Records
IPFS	InterPlanetary File System
ZKP	Zero-Knowledge Proof
SCM	Supply Chain Management
SPOF	Single Point of Failure

References

- Yadav, S.P.; Agrawal, K.K.; Bhati, B.S.; Al-Turjman, F.; Mostarda, L. Blockchain-based cryptocurrency regulation: An overview. *Comput. Econ.* **2022**, *59*, 1659–1675. [[CrossRef](#)]
- Li, X.; Jiang, P.; Chen, T.; Luo, X.; Wen, Q. A survey on the security of blockchain systems. *Future Gener. Comput. Syst.* **2020**, *107*, 841–853. [[CrossRef](#)]
- Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: <https://bitcoin.org/bitcoin.pdf> (accessed on 6 February 2024).
- Garay, J.; Kiayias, A.; Leonardos, N. The bitcoin backbone protocol: Analysis and applications. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 26–30 April 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 281–310.
- Wood, G. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Proj. Yellow Pap.* **2014**, *151*, 1–32.
- Werner, S.; Perez, D.; Gudgeon, L.; Klages-Mundt, A.; Harz, D.; Knottenbelt, W. SoK: Decentralized finance (defi). In Proceedings of the 4th ACM Conference on Advances in Financial Technologies, New York, NY, USA, 19–21 September 2022; pp. 30–46.
- He, S.; Tang, Q.; Wu, C.Q.; Shen, X. Decentralizing IoT management systems using blockchain for censorship resistance. *IEEE Trans. Ind. Inform.* **2019**, *16*, 715–727. [[CrossRef](#)]
- He, S.; Ficke, E.; Pritom, M.M.A.; Chen, H.; Tang, Q.; Chen, Q.; Pendleton, M.; Njilla, L.; Xu, S. Blockchain-based automated and robust cyber security management. *J. Parallel Distrib. Comput.* **2022**, *163*, 62–82. [[CrossRef](#)]
- He, S.; Lu, Y.; Tang, Q.; Wang, G.; Wu, C.Q. Blockchain-based P2P content delivery with monetary incentivization and fairness guarantee. *IEEE Trans. Parallel Distrib. Syst.* **2022**, *34*, 746–765. [[CrossRef](#)]
- Merlo, V.; Pio, G.; Giusto, F.; Bilancia, M. On the exploitation of the blockchain technology in the healthcare sector: A systematic review. *Expert Syst. Appl.* **2023**, *213*, 118897. [[CrossRef](#)]
- Ullah, Z.; Naeem, M.; Coronato, A.; Ribino, P.; De Pietro, G. Blockchain applications in sustainable smart cities. *Sustain. Cities Soc.* **2023**, *97*, 104697. [[CrossRef](#)]

12. Huynh-The, T.; Gadekallu, T.R.; Wang, W.; Yenduri, G.; Ranaweera, P.; Pham, Q.V.; da Costa, D.B.; Liyanage, M. Blockchain for the metaverse: A Review. *Future Gener. Comput. Syst.* **2023**, *143*, 401–419. [[CrossRef](#)]
13. Lin, Y.; Gao, Z.; Tu, Y.; Du, H.; Niyato, D.; Kang, J.; Yang, H. A blockchain-based semantic exchange framework for web 3.0 toward participatory economy. *IEEE Commun. Mag.* **2023**, *61*, 94–100. [[CrossRef](#)]
14. Zhang, R.; Xue, R.; Liu, L. Security and privacy on blockchain. *ACM Comput. Surv.* **2019**, *52*, 1–34. [[CrossRef](#)]
15. Leng, J.; Zhou, M.; Zhao, J.L.; Huang, Y.; Bian, Y. Blockchain security: A survey of techniques and research directions. *IEEE Trans. Serv. Comput.* **2020**, *15*, 2490–2510. [[CrossRef](#)]
16. Kosba, A.; Miller, A.; Shi, E.; Wen, Z.; Papamanthou, C. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In Proceedings of the IEEE Symposium on Security and Privacy, San Jose, CA, USA, 22–26 May 2016; pp. 839–858.
17. Kiayias, A.; Zhou, H.S.; Zikas, V. Fair and robust multi-party computation using a global transaction ledger. In Proceedings of the 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, 8–12 May 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 705–734.
18. Juels, A.; Kosba, A.; Shi, E. The ring of gyges: Investigating the future of criminal smart contracts. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 283–295.
19. Canetti, R.; Dodis, Y.; Pass, R.; Walfish, S. Universally composable security with global setup. In Proceedings of the 4th Theory of Cryptography Conference, Amsterdam, The Netherlands, 21–24 February 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 61–85.
20. Cheng, R.; Zhang, F.; Kos, J.; He, W.; Hynes, N.; Johnson, N.; Juels, A.; Miller, A.; Song, D. Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy, Stockholm, Sweden, 17–19 June 2019; pp. 185–200.
21. Schneider, F.B. Implementing fault-tolerant services using the state machine approach: A tutorial. *ACM Comput. Surv.* **1990**, *22*, 299–319. [[CrossRef](#)]
22. Baudet, M.; Ching, A.; Chursin, A.; Danezis, G.; Garillot, F.; Li, Z.; Malkhi, D.; Naor, O.; Perelman, D.; Sonnino, A. *State Machine Replication in the Libra Blockchain*; The Libra Assn. Tech. Report; The Diem Association: Geneva, Switzerland, 2019.
23. Bessani, A.; Alchieri, E.; Sousa, J.; Oliveira, A.; Pedone, F. From byzantine replication to blockchain: Consensus is only the beginning. In Proceedings of the 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, Valencia, Spain, 29 June–2 July 2020; pp. 424–436.
24. Castro, M.; Liskov, B. Practical byzantine fault tolerance. In Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, LA, USA, 22–25 February 1999; pp. 173–186.
25. Cowling, J.; Myers, D.; Liskov, B.; Rodrigues, R.; Shrira, L. HQ replication: A hybrid quorum protocol for byzantine fault tolerance. In Proceedings of the 7th Symposium on Operating Systems Design and Implementation, Seattle, WA, USA, 6–8 November 2006; pp. 177–190.
26. Kotla, R.; Alvisi, L.; Dahlin, M.; Clement, A.; Wong, E. Zyzzyva: Speculative byzantine fault tolerance. In Proceedings of the 21st ACM SIGOPS Symposium on Operating Systems Principles, Stevenson, WA, USA, 14–17 October 2007; pp. 45–58.
27. Veronese, G.S.; Correia, M.; Bessani, A.N.; Lung, L.C. Spin one's wheels? Byzantine fault tolerance with a spinning primary. In Proceedings of the 28th International Symposium on Reliable Distributed Systems, Niagara Falls, NY, USA, 27–30 September 2009; pp. 135–144.
28. Amir, Y.; Coan, B.; Kirsch, J.; Lane, J. Prime: Byzantine replication under attack. *IEEE Trans. Dependable Secur. Comput.* **2010**, *8*, 564–577. [[CrossRef](#)]
29. Veronese, G.S.; Correia, M.; Bessani, A.N.; Lung, L.C.; Verissimo, P. Efficient byzantine fault-tolerance. *IEEE Trans. Comput.* **2011**, *62*, 16–30. [[CrossRef](#)]
30. Aublin, P.L.; Mokhtar, S.B.; Quéma, V. RBFT: Redundant byzantine fault tolerance. In Proceedings of the 33rd International Conference on Distributed Computing Systems, Philadelphia, PA, USA, 8–11 July 2013; pp. 297–306.
31. Sankar, L.S.; Sindhu, M.; Sethumadhavan, M. Survey of consensus protocols on blockchain applications. In Proceedings of the 4th International Conference on Advanced Computing and Communication Systems, Coimbatore, India, 6–7 January 2017; pp. 1–5.
32. Li, W.; Feng, C.; Zhang, L.; Xu, H.; Cao, B.; Imran, M.A. A scalable multi-layer PBFT consensus for blockchain. *IEEE Trans. Parallel Distrib. Syst.* **2020**, *32*, 1146–1160. [[CrossRef](#)]
33. Wang, R.; Ye, K.; Meng, T.; Xu, C.Z. Performance evaluation on blockchain systems: A case study on Ethereum, Fabric, Sawtooth and Fisco-Bcos. In Proceedings of the Services Computing—SCC 2020, Honolulu, HI, USA, 18–20 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 120–134.
34. Belchior, R.; Vasconcelos, A.; Guerreiro, S.; Correia, M. A survey on blockchain interoperability: Past, present, and future trends. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–41. [[CrossRef](#)]
35. 101 Blockchains. Hyperledger vs. Corda vs. Ethereum: The Ultimate Comparison. 2021. Available online: <https://101blockchains.com/hyperledger-vs-corda-r3-vs-ethereum/> (accessed on 6 February 2024).
36. Li, H.; Chen, Y.; Shi, X.; Bai, X.; Mo, N.; Li, W.; Guo, R.; Wang, Z.; Sun, Y. FISCO-BCOS: An enterprise-grade permissioned blockchain system with high-performance. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Denver, CO, USA, 12–17 November 2023; pp. 1–17.
37. Capocasale, V.; Gotta, D.; Perboli, G. Comparative analysis of permissioned blockchain frameworks for industrial applications. *Blockchain Res. Appl.* **2023**, *4*, 100113. [[CrossRef](#)]

38. Sharma, A.; Schuhknecht, F.M.; Agrawal, D.; Dittrich, J. Blurring the lines between blockchains and database systems: The case of hyperledger fabric. In Proceedings of the International Conference on Management of Data, Amsterdam, The Netherlands, 30 June–5 July 2019; pp. 105–122.
39. Chacko, J.A.; Mayer, R.; Jacobsen, H.A. Why do my blockchain transactions fail? a study of hyperledger fabric. In Proceedings of the 2021 International Conference on Management of Data, Virtual, 20–25 June 2021; pp. 221–234.
40. Wang, J.; Wang, S.; Zhang, Q.; Deng, Y. A two-layer consortium blockchain with transaction privacy protection based on sharding technology. *J. Inf. Secur. Appl.* **2023**, *74*, 103452. [[CrossRef](#)]
41. Wu, X.; Jiang, W.; Song, M.; Jia, Z.; Qin, J. An efficient sharding consensus algorithm for consortium chains. *Sci. Rep.* **2023**, *13*, 20. [[CrossRef](#)] [[PubMed](#)]
42. Tao, S.; Li, T.; Zhuo, Y.; Bai, F.; Chi, Z. GT-NRSM: Efficient and scalable sharding consensus mechanism for consortium blockchain. *J. Supercomput.* **2023**, *79*, 20041–20075.
43. Zheng, P.; Xu, Q.; Zheng, Z.; Zhou, Z.; Yan, Y.; Zhang, H. Meepo: Multiple execution environments per organization in sharded consortium blockchain. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 3562–3574. [[CrossRef](#)]
44. Zhou, Z.; Qiu, Z.; Yu, Q.; Chen, H. A dynamic sharding protocol design for consortium blockchains. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020.
45. Qi, X.; Zhang, Z.; Jin, C.; Zhou, A. BFT-Store: Storage partition for permissioned blockchain via erasure coding. In Proceedings of the 2020 IEEE 36th International Conference on Data Engineering (ICDE), Dallas, TX, USA, 20–24 April 2020.
46. Farahat, I.S.; Aladrousy, W.; Elhoseny, M.; Elmougy, S.; Tolba, A.E. Secure medical blockchain model. *Information* **2023**, *14*, 80. [[CrossRef](#)]
47. Soundarapandian, K.; Ambrose, A.K. Lossless data compression and blockchain-assisted aggregation for overlapped-clusters sensor networks. *Wirel. Pers. Commun. Int. J.* **2023**, *131*, 1313–1337. [[CrossRef](#)]
48. Yu, B.; Li, X.; Zhao, H. PoW-BC: A PoW consensus protocol based on block compression. *KSII Trans. Internet Inf. Syst.* **2021**, *15*, 4.
49. Liu, L.; Liu, X.; Wan, J. Design of updating encryption algorithm for privacy big data based on consortium blockchain technology. *J. Math.* **2022**, *2022*, 7138173. [[CrossRef](#)]
50. Zhou, E.; Hong, Z.; Xiao, Y.; Zhao, D.; Pei, Q.; Guo, S.; Akerkar, R. MSTDB: A hybrid storage-empowered scalable semantic blockchain database. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 8228–8244. [[CrossRef](#)]
51. Chen, X.; Lin, S.; Yu, N. Bitcoin blockchain compression algorithm for blank node synchronization. In Proceedings of the 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), Xi'an, China, 23–25 October 2019.
52. Shafarenko, A. Indexing structures for the PLS blockchain. *Cybersecurity* **2022**, *5*, 19. [[CrossRef](#)]
53. He, S.; Sun, T.; Tang, Q.; Wu, C.; Lipka, N.; Wigington, C.; Jain, R. Secure and efficient agreement signing atop blockchain and decentralized identity. In Proceedings of the International Conference on Blockchain and Trustworthy Systems, Chengdu, China, 4–5 August 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 3–17.
54. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; De Caro, A.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger Fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference, EuroSys '18, Porto, Portugal, 23–26 April 2018; Association for Computing Machinery: New York, NY, USA, 2018.
55. Javaid, H.; Hu, C.; Brebner, G. Optimizing validation phase of hyperledger fabric. In Proceedings of the 2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Rennes, France, 21–25 October 2019; pp. 269–275.
56. Kwon, M.; Yu, H. Performance improvement of ordering and endorsement phase in hyperledger fabric. In Proceedings of the 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), Granada, Spain, 22–25 October 2019; pp. 428–432.
57. Nakaike, T.; Zhang, Q.; Ueda, Y.; Inagaki, T.; Ohara, M. Hyperledger fabric performance characterization and optimization using goleveldb benchmark. In Proceedings of the International Conference on Blockchain and Cryptocurrency (ICBC), Toronto, ON, Canada, 2–6 May 2020; pp. 1–9.
58. Kang, H.; Dai, T.; Jean-Louis, N.; Tao, S.; Gu, X. Fabzk: Supporting privacy-preserving, auditable smart contracts in hyperledger fabric. In Proceedings of the 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Portland, OR, USA, 24–27 June 2019; pp. 543–555.
59. Graf, M.; Küsters, R.; Rausch, D. Accountability in a permissioned blockchain: Formal analysis of hyperledger fabric. In Proceedings of the European Symposium on Security and Privacy (EuroS&P), Genoa, Italy, 7–11 September 2020; pp. 236–255.
60. Dharani, J.; Sundarakantham, K.; Singh, K. A privacy-preserving framework for endorsement process in hyperledger fabric. *Comput. Secur.* **2022**, *116*, 102637.
61. Bu, G.; Haouara, R.; Nguyen, T.S.L.; Potop-Butucaru, M. Cross hyperledger fabric transactions. In Proceedings of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems, London, UK, 25 September 2020; pp. 35–40.
62. Kuzlu, M.; Pipattanasomporn, M.; Gurses, L.; Rahman, S. Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 536–540.

63. Shuaib, M.; Hassan, N.H.; Usman, S.; Alam, S.; Bakar, N.A.A.; Maarop, N. Performance evaluation of DLT systems based on hyper ledger fabric. In Proceedings of the 4th International Conference on Smart Sensors and Application (ICSSA), Kuala Lumpur, Malaysia, 26–28 July 2022; pp. 70–75.
64. Kushwaha, S.S.; Joshi, S.; Singh, D.; Kaur, M.; Lee, H.N. Systematic review of security vulnerabilities in ethereum blockchain smart contract. *IEEE Access* **2022**, *10*, 6605–6621. [CrossRef]
65. Atzei, N.; Bartoletti, M.; Cimoli, T. A survey of attacks on ethereum smart contracts (sok). In Proceedings of the European Joint Conferences on Theory and Practice of Software, Uppsala, Sweden, 22–29 April 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 164–186.
66. Hildenbrandt, E.; Saxena, M.; Rodrigues, N.; Zhu, X.; Daian, P.; Guth, D.; Moore, B.; Park, D.; Zhang, Y.; Stefanescu, A.; et al. Kevm: A complete formal semantics of the ethereum virtual machine. In Proceedings of the 31st Computer Security Foundations Symposium, Oxford, UK, 9–12 July 2018; pp. 204–217.
67. Zheng, P.; Jiang, Z.; Wu, J.; Zheng, Z. Blockchain-based decentralized application: A survey. *IEEE Open J. Comput. Soc.* **2023**, *4*, 121–133. [CrossRef]
68. Zhang, H.; Jin, C.; Cui, H. A method to predict the performance and storage of executing contract for ethereum consortium-blockchain. In Proceedings of the International Conference on Blockchain, Seattle, WA, USA, 25–30 June 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 63–74.
69. Rouhani, S.; Deters, R. Performance analysis of ethereum transactions in private blockchain. In Proceedings of the International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 24–26 November 2017; pp. 70–74.
70. Mao, D.; Hao, Z.; Wang, F.; Li, H. Novel automatic food trading system using consortium blockchain. *Arab. J. Sci. Eng.* **2019**, *44*, 3439–3455. [CrossRef]
71. Al-Shaibani, H.; Lasla, N.; Abdallah, M. Consortium blockchain-based decentralized stock exchange platform. *IEEE Access* **2020**, *8*, 123711–123725. [CrossRef]
72. Li, Z.; Hao, J.; Liu, J.; Wang, H.; Xian, M. An IoT-applicable access control model under double-layer blockchain. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *68*, 2102–2106. [CrossRef]
73. Brown, R.; Carlyle, J.; Grigg, I.; Hearn, M. Corda: An introduction. *R3 CEV* **2016**, *1*, 14.
74. R3. R3’s Corda Partner Network Grows to over 60 Companies Including Hewlett Packard Enterprise. 2017. Available online: <https://r3.com/press-media/r3s-corda-partner-network-grows-to-over-60-companies-including-hewlett-packard-enterprise-intel-and-microsoft/> (accessed on 29 December 2023).
75. Graf, M.; Rausch, D.; Ronge, V.; Egger, C.; Küsters, R.; Schröder, D. A security framework for distributed ledgers. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Virtual, 15–19 November 2021; pp. 1043–1064.
76. DLT Magazine. An in-Depth Guide to Corda: Exploring Its Features and Benefits. 2023. Available online: <https://distributed-ledger.tech/articles/an-in-depth-guide-to-corda-exploring-its-features-and-benefits/> (accessed on 5 February 2024).
77. 4IRE. Why Choose Corda for Your Project? 2021. Available online: <https://4irelabs.com/articles/why-choose-corda-for-your-project/> (accessed on 7 February 2024).
78. ConsenSys. Build on Quorum, the Complete Open Source Blockchain Platform for Business. 2021. Available online: <https://consensys.io/quorum/> (accessed on 27 December 2023).
79. ConsenSys. ConsenSys GoQuorum. 2023. Available online: <https://docs.goquorum.consenSys.io/> (accessed on 29 December 2023).
80. Hyperledger Besu Community. Hyperledger Besu Ethereum Client. 2023. Available online: <https://besu.hyperledger.org/> (accessed on 29 December 2023).
81. Mazzoni, M.; Corradi, A.; Di Nicola, V. Performance evaluation of permissioned blockchains for financial applications: The ConsenSys Quorum case study. *Blockchain Res. Appl.* **2022**, *3*, 100026. [CrossRef]
82. Armknecht, F.; Karame, G.O.; Mandal, A.; Youssef, F.; Zenner, E. Ripple: Overview and outlook. In Proceedings of the International Conference on Trust and Trustworthy Computing, Heraklion, Greece, 24–26 August 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 163–180.
83. Amores-Sesar, I.; Cachin, C.; Mićić, J. Security analysis of ripple consensus. *arXiv* **2020**, arXiv:2011.14816.
84. De Rossi, L.M.; AbbateMarco, N.; Salvietti, G. Towards a comprehensive blockchain architecture continuum. In Proceedings of the 52nd Hawaii International Conference on System Sciences, Maui, HI, USA, 8–11 January 2019; Volume 6, pp. 4605–4614.
85. Pandey, S.; Ojha, G.; Shrestha, B.; Kumar, R. BlockSIM: A practical simulation tool for optimal network design, stability and planning. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Republic of Korea, 14–17 May 2019.
86. R3. Performance Benchmarking Results. Available online: <https://docs.r3.com/en/platform/corda/4.8/enterprise/performance-testing/performance-results.html> (accessed on 7 February 2024).
87. Baliga, A.; Solanki, N.; Verekar, S.; Pednekar, A.; Kamat, P.; Chatterjee, S. Performance characterization of hyperledger fabric. In Proceedings of the Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland, 20–22 June 2018; pp. 65–74.
88. Han, R.; Gramoli, V.; Xu, X. Evaluating blockchains for IoT. In Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 26–28 February 2018.
89. Li, X.; Zhao, B.; Yang, G.; Xiang, T.; Weng, J.; Deng, R.H. A survey of secure computation using trusted execution Environments. *arXiv* **2023**, arXiv:2302.12150.

90. Bentov, I.; Ji, Y.; Zhang, F.; Breidenbach, L.; Daian, P.; Juels, A. Tesseract: Real-time cryptocurrency exchange using trusted hardware. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 1521–1538.
91. Costan, V.; Devadas, S. *Intel SGX Explained*; Cryptology ePrint Archive: Cambridge, MA, USA, 2016.
92. Futral, W.; Greene, J.; Futral, W.; Greene, J. Fundamental principles of intel® txt. In *Intel® Trusted Execution Technology for Server Platforms: A Guide to More Secure Datacenters*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 15–36.
93. Pinto, S.; Santos, N. Demystifying arm trustzone: A comprehensive survey. *ACM Comput. Surv.* **2019**, *51*, 1–36. [CrossRef]
94. Zhao, S.; Li, M.; Zhangyz, Y.; Lin, Z. Vsgx: Virtualizing sgx enclaves on amd sev. In Proceedings of the Symposium on Security and Privacy, San Francisco, CA, USA, 27 July 2022; pp. 321–336.
95. Costan, V.; Lebedev, I.; Devadas, S. Sanctum: Minimal hardware extensions for strong software isolation. In Proceedings of the 25th USENIX Security Symposium, Austin, TX, USA, 10–12 August 2016; pp. 857–874.
96. Lee, D.; Kohlbrenner, D.; Shinde, S.; Asanović, K.; Song, D. Keystone: An open framework for architecting trusted execution environments. In Proceedings of the 15th European Conference on Computer Systems, Heraklion Greece, 27–30 April 2020; pp. 1–16.
97. Pass, R.; Shi, E.; Tramer, F. Formal abstractions for attested execution secure processors. In Proceedings of the 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, 30 April–4 May 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 260–289.
98. Fei, S.; Yan, Z.; Ding, W.; Xie, H. Security vulnerabilities of SGX and countermeasures: A survey. *ACM Comput. Surv.* **2021**, *54*, 1–36. [CrossRef]
99. Choi, J.I.; Butler, K.R. Secure multiparty computation and trusted hardware: Examining adoption challenges and opportunities. *Secur. Commun. Netw.* **2019**, *2019*, 1368905. [CrossRef]
100. Sasy, S.; Gorbunov, S.; Fletcher, C.W. ZeroTrace: Oblivious memory primitives from Intel SGX. In Proceedings of the Annual Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 18–21 February 2018.
101. Bellare, M.; Fischlin, M.; Goldwasser, S.; Micali, S. Identification protocols secure against reset attacks. In Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, 6–10 May 2001; Springer: Berlin/Heidelberg, Germany, 2001; pp. 495–511.
102. Fleischer, F.; Busch, M.; Kuhrt, P. Memory corruption attacks within Android TEEs: A case study based on OP-TEE. In Proceedings of the 15th International Conference on Availability, Reliability and Security, Virtual, 25–28 August 2020; pp. 1–9.
103. Cerdeira, D.; Santos, N.; Fonseca, P.; Pinto, S. Sok: Understanding the prevailing security vulnerabilities in trustzone-assisted tee systems. In Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 18–21 May 2020; pp. 1416–1432.
104. Ghaniyoun, M.; Barber, K.; Xiao, Y.; Zhang, Y.; Teodorescu, R. TEEsec: Pre-Silicon Vulnerability Discovery for Trusted Execution Environments. In Proceedings of the 50th Annual International Symposium on Computer Architecture, Orlando, FL, USA, 17–21 June 2023; pp. 1–15.
105. Decker, C.; Wattenhofer, R. Information propagation in the bitcoin network. In Proceedings of the IEEE P2P 2013 Proceedings, Trento, Italy, 9–11 September 2013; pp. 1–10.
106. Gervais, A.; Karame, G.O.; Wüst, K.; Glykantzis, V.; Ritzdorf, H.; Capkun, S. On the security and performance of proof of work blockchains. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 3–16.
107. Gervais, A.; Capkun, S.; Karame, G.O.; Gruber, D. On the privacy provisions of bloom filters in lightweight bitcoin clients. In Proceedings of the 30th Annual Computer Security Applications Conference, New Orleans, LA, USA, 8–12 December 2014; pp. 326–335.
108. Manevich, Y.; Barger, A.; Tock, Y. Service discovery for hyperledger fabric. In Proceedings of the 12th ACM International Conference on Distributed and Event-Based Systems, Hamilton, New Zealand, 25–29 June 2018; pp. 226–229.
109. Gudgeon, L.; Moreno-Sánchez, P.; Roos, S.; McCorry, P.; Gervais, A. Sok: Layer-two blockchain protocols. In Proceedings of the Financial Cryptography and Data Security, Kota Kinabalu, Malaysia, 10–14 February 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 201–226.
110. Basha, S.J.; Veerasam, V.S.; Ammannamma, T.; Navudu, S.; Subrahmanyam, M. Security enhancement of digital signatures for blockchain using EdDSA algorithm. In Proceedings of the 3rd International Conference on Intelligent Communication Technologies and Virtual Mobile Networks, Tirunelveli, India, 4–6 February 2021; pp. 274–278.
111. Aumasson, J.P.; Hamelink, A.; Shlomovits, O. *A Survey of ECDSA Threshold Signing*; Cryptology ePrint Archive: Cambridge, MA, USA, 2020.
112. Canetti, R.; Jain, A.; Scafuro, A. Practical UC security with a global random oracle. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AR, USA, 3–7 November 2014.
113. Pervez, H.; Muneeb, M.; Irfan, M.U.; Haq, I.U. A comparative analysis of DAG-based blockchain architectures. In Proceedings of the 12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 19–21 December 2018.
114. Wu, H.Y.; Yang, X.; Yue, C.; Paik, H.Y.; Kanhere, S.S. Chain or DAG? Underlying data structures, architectures, topologies and consensus in distributed ledger technology: A review, taxonomy and research issues. *J. Syst. Archit.* **2022**, *131*, 102720. [CrossRef]
115. Wang, Q.; Yu, J.; Chen, S.; Xiang, Y. Sok: Dag-based blockchain systems. *ACM Comput. Surv.* **2023**, *55*, 1–38. [CrossRef]

116. Fan, C.; Ghaemi, S.; Khazaei, H.; Chen, Y.; Musilek, P. Performance analysis of the IOTA DAG-based distributed ledger. *ACM Trans. Model. Perform. Eval. Comput. Syst.* **2021**, *6*, 1–20. [[CrossRef](#)]
117. Dziembowski, S.; Eckey, L.; Faust, S. Fairswap: How to fairly exchange digital goods. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 967–984.
118. Mingxiao, D.; Xiaofeng, M.; Zhe, Z.; Xiangwei, W.; Qijun, C. A review on consensus algorithm of blockchain. In Proceedings of the International Conference on Systems, Man, and Cybernetics, Banff, AB, Canada, 5–8 October 2017; pp. 2567–2572.
119. Nguyen, G.T.; Kim, K. A survey about consensus algorithms used in blockchain. *J. Inf. Process. Syst.* **2018**, *14*, 101–128.
120. Alsunaidi, S.J.; Alhaidari, F.A. A survey of consensus algorithms for blockchain technology. In Proceedings of the International Conference on Computer and Information Sciences, Sakaka, Saudi Arabia, 3–4 April 2019; pp. 1–6.
121. Fu, X.; Wang, H.; Shi, P. A survey of blockchain consensus algorithms: Mechanism, design and applications. *Sci. China Inf. Sci.* **2020**, *64*, 121101. [[CrossRef](#)]
122. Wan, S.; Li, M.; Liu, G.; Wang, C. Recent advances in consensus protocols for blockchain: A survey. *Wirel. Netw.* **2020**, *26*, 5579–5593. [[CrossRef](#)]
123. Ferdous, M.S.; Chowdhury, M.J.M.; Hoque, M.A. A survey of consensus algorithms in public blockchain systems for cryptocurrencies. *J. Netw. Comput. Appl.* **2021**, *182*, 103035. [[CrossRef](#)]
124. Lashkari, B.; Musilek, P. A comprehensive review of blockchain consensus mechanisms. *IEEE Access* **2021**, *9*, 43620–43652. [[CrossRef](#)]
125. Bouraga, S. A taxonomy of blockchain consensus protocols: A survey and classification framework. *Expert Syst. Appl.* **2021**, *168*, 114384. [[CrossRef](#)]
126. Guru, D.; Perumal, S.; Varadarajan, V. Approaches towards blockchain innovation: A survey and future directions. *Electronics* **2021**, *10*, 1219. [[CrossRef](#)]
127. Khan, M.; den Hartog, F.; Hu, J. A survey and ontology of blockchain consensus algorithms for resource-constrained IoT systems. *Sensors* **2022**, *22*, 8188. [[CrossRef](#)] [[PubMed](#)]
128. Yao, W.; Deek, F.P.; Murimi, R.; Wang, G. SoK: A taxonomy for critical analysis of consensus mechanisms in consortium blockchain. *IEEE Access* **2023**, *11*, 79572–79587. [[CrossRef](#)]
129. Guru, A.; Abhishek, M.; Mohanta, B.K.; Mohapatra, H.; Al-Turjman, F.; Altrjman, C.; Yadav, A. A survey on consensus protocols and attacks on blockchain technology. *Appl. Sci.* **2023**, *13*, 2604. [[CrossRef](#)]
130. De Morais, A.M.; Lins, F.A.A.; Rosa, N.S. Survey on integration of consensus mechanisms in IoT-based blockchains. *JUICS J. Univers. Comput. Sci.* **2023**, *29*, 1139–1160. [[CrossRef](#)]
131. Luo, H.; Yu, H.; Luo, J. PRAFT and RPBFT: A class of blockchain consensus algorithm and their applications in electric vehicles charging scenarios for V2G networks. *Internet Things Cyber-Phys. Syst.* **2023**, *3*, 61–70. [[CrossRef](#)]
132. Ahmad, A.; Saad, M.; Kim, J.; Nyang, D.; Mohaisen, D. Performance evaluation of consensus protocols in blockchain-based audit systems. In Proceedings of the 2021 International Conference on Information Networking (ICOIN), Jeju Island, Republic of Korea, 13–16 January 2021.
133. Bentov, I.; Lee, C.; Mizrahi, A.; Rosenfeld, M. Proof of Activity: Extending bitcoin’s proof of work via proof of stake [Extended Abstract]. *SIGMETRICS Perform. Eval. Rev.* **2014**, *42*, 34–37. [[CrossRef](#)]
134. Song, H.; Zhu, N.; Xue, R.; He, J.; Zhang, K.; Wang, J. Proof-of-Contribution consensus mechanism for blockchain and its application in intellectual property protection. *Inf. Process. Manag.* **2021**, *58*, 102507. [[CrossRef](#)]
135. BitShares. Delegated Proof of Stake (DPOS). Available online: <https://how.bitshares.works/en/master/technology/dpos.html> (accessed on 6 February 2024).
136. Yao, W.; Ye, J.; Murimi, R.; Wang, G. A survey on consortium blockchain consensus mechanisms. *arXiv* **2021**, arXiv:2102.12058.
137. Han, R.; Shapiro, G.; Gramoli, V.; Xu, X. On the performance of distributed ledgers for internet of things. *Internet Things* **2020**, *10*, 100087. [[CrossRef](#)]
138. Miller, A.; Xia, Y.; Croman, K.; Shi, E.; Song, D. The honey badger of BFT protocols. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 31–42.
139. Guo, B.; Lu, Z.; Tang, Q.; Xu, J.; Zhang, Z. Dumbo: Faster asynchronous BFT protocols. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual, 9–13 November 2020; Association for Computing Machinery: New York, NY, USA, 2020.
140. Zhang, F.; Cecchetti, E.; Croman, K.; Juels, A.; Shi, E. Town crier: An authenticated data feed for smart contracts. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS), Vienna, Austria, 24–28 October 2016; pp. 270–282.
141. Kiayias, A.; Russell, A.; David, B.; Oliynykov, R. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 20–24 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 357–388.
142. Kokoris-Kogias, E.; Jovanovic, P.; Gasser, L.; Gailly, N.; Syta, E.; Ford, B. Omnipledger: A secure, scale-out, decentralized ledger via sharding. In Proceedings of the Symposium on Security and Privacy, San Francisco, CA, USA, 20–24 May 2018; pp. 583–598.
143. Gangwal, A.; Gangavalli, H.R.; Thirupathi, A. A survey of Layer-two blockchain protocols. *J. Netw. Comput. Appl.* **2023**, *209*, 103539. [[CrossRef](#)]

144. Miller, A.; Bentov, I.; Bakshi, S.; Kumaresan, R.; McCorry, P. Sprites and state channels: Payment networks that go faster than lightning. In Proceedings of the International Conference on Financial Cryptography and Data Security, Frigate Bay, St. Kitts and Nevis, 18–22 February 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 508–526.
145. Decker, C.; Wattenhofer, R. A fast and scalable payment network with bitcoin duplex micropayment channels. In Proceedings of the Symposium on Self-Stabilizing Systems, Edmonton, AB, Canada, 18–21 August 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 3–18.
146. Dziembowski, S.; Eckey, L.; Faust, S.; Malinowski, D. Perun: Virtual payment hubs over cryptocurrencies. In Proceedings of the Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 106–123.
147. Khalil, R.; Zamyatin, A.; Felley, G.; Moreno-Sánchez, P.; Gervais, A. *Commit-Chains: Secure, Scalable Off-Chain Payments*; Cryptology ePrint Archive: Cambridge, MA, USA, 2018.
148. Dziembowski, S.; Fabiański, G.; Faust, S.; Riahi, S. *Lower Bounds for Off-Chain Protocols: Exploring the Limits of Plasma*; Cryptology ePrint Archive: Cambridge, MA, USA, 2020.
149. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to scalability of blockchain: A survey. *IEEE Access* **2020**, *8*, 16440–16455. [[CrossRef](#)]
150. Kalodner, H.; Goldfeder, S.; Chen, X.; Weinberg, S.M.; Felten, E.W. Arbitrum: Scalable, private smart contracts. In Proceedings of the 27th USENIX Security Symposium, Baltimore, MD, USA, 15–17 August 2018; pp. 1353–1370.
151. Sukhwani, H.; Martínez, J.M.; Chang, X.; Trivedi, K.S.; Rindos, A. Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric). In Proceedings of the 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong, China, 26–29 September 2017; pp. 253–255.
152. Xu, X.; Sun, G.; Luo, L.; Cao, H.; Yu, H.; Vasilakos, A.V. Latency performance modeling and analysis for hyperledger fabric blockchain network. *Inf. Process. Manag.* **2021**, *58*, 102436. [[CrossRef](#)]
153. Meng, T.; Zhao, Y.; Wolter, K.; Xu, C.Z. On consortium blockchain consistency: A queueing network model approach. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *32*, 1369–1382. [[CrossRef](#)]
154. Memon, R.A.; Li, J.P.; Ahmed, J. Simulation model for blockchain systems using queuing theory. *Electronics* **2019**, *8*, 234. [[CrossRef](#)]
155. Yuan, P.; Zheng, K.; Xiong, X.; Zhang, K.; Lei, L. Performance modeling and analysis of a Hyperledger-based system using GSPN. *Comput. Commun.* **2020**, *153*, 117–124. [[CrossRef](#)]
156. Sukhwani, H.; Wang, N.; Trivedi, K.S.; Rindos, A. Performance modeling of hyperledger fabric (permissioned blockchain network). In Proceedings of the 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 1–3 November 2018; pp. 1–8.
157. Papadis, N.; Borst, S.; Walid, A.; Grissa, M.; Tassiulas, L. Stochastic models and wide-area network measurements for blockchain design and analysis. In Proceedings of the IEEE INFOCOM 2018–IEEE Conference on Computer Communications, Honolulu, HI, USA, 16–19 April 2018; pp. 2546–2554.
158. Li, M.; Wang, Y.; Ma, S.; Liu, C.; Huo, D.; Wang, Y.; Xu, Z. Auto-tuning with reinforcement learning for permissioned blockchain systems. *Proc. VLDB Endow.* **2023**, *16*, 1000–1012. [[CrossRef](#)]
159. Atlam, H.F.; Wills, G.B. Technical aspects of blockchain and IoT. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 2019; Volume 115, pp. 1–39.
160. Mathur, S.; Kalla, A.; Gür, G.; Bohra, M.K.; Liyanage, M. A Survey on Role of Blockchain for IoT: Applications and Technical Aspects. *Comput. Netw.* **2023**, *227*, 109726. [[CrossRef](#)]
161. Rathee, G.; Sharma, A.; Saini, H.; Kumar, R.; Iqbal, R. A hybrid framework for multimedia data processing in IoT-healthcare using blockchain technology. *Multimed. Tools Appl.* **2020**, *79*, 9711–9733. [[CrossRef](#)]
162. Lahbib, A.; Toumi, K.; Laouiti, A.; Laube, A.; Martin, S. Blockchain based trust management mechanism for IoT. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–8.
163. Kumar, R.; Sharma, R. Leveraging blockchain for ensuring trust in IoT: A survey. *Comput. Inf. Sci.* **2022**, *34*, 8599–8622. [[CrossRef](#)]
164. Maram, D.; Malvai, H.; Zhang, F.; Jean-Louis, N.; Frolov, A.; Kell, T.; Lobban, T.; Moy, C.; Juels, A.; Miller, A. Candid: Can-do decentralized identity with legacy compatibility, sybil-resistance, and accountability. In Proceedings of the Symposium on Security and Privacy (SP), San Francisco, CA, USA, 24–27 May 2021; pp. 1348–1366.
165. Reyna, A.; Martín, C.; Chen, J.; Soler, E.; Díaz, M. On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **2018**, *88*, 173–190. [[CrossRef](#)]
166. Thakker, J.; Chang, I.; Park, Y. Secure data management in internet-of-things based on blockchain. In Proceedings of the International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 23 March 2020; pp. 1–5.
167. Wang, Y.; Zhang, A.; Zhang, P.; Qu, Y.; Yu, S. Security-aware and privacy-preserving personal health record sharing using consortium blockchain. *IEEE Internet Things J.* **2021**, *9*, 12014–12028. [[CrossRef](#)]
168. Zhang, A.; Lin, X. Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain. *J. Med. Syst.* **2018**, *42*, 140. [[CrossRef](#)]
169. Ni, W.; Huang, X.; Zhang, J.; Yu, R. Healchain: A decentralized data management system for mobile healthcare using consortium blockchain. In Proceedings of the Chinese Control Conference (CCC), Guangzhou, China, 27–30 July 2019; pp. 6333–6338.
170. Du, M.; Chen, Q.; Chen, J.; Ma, X. An optimized consortium blockchain for medical information sharing. *IEEE Trans. Eng. Manag.* **2020**, *68*, 1677–1689. [[CrossRef](#)]

171. Jabbar, S.; Lloyd, H.; Hammoudeh, M.; Adebisi, B.; Raza, U. Blockchain-enabled supply chain: Analysis, challenges, and future directions. *Multimed. Syst.* **2021**, *27*, 787–806. [[CrossRef](#)]
172. Nguyen, H.; Do, L. The Adoption of Blockchain in Food Retail Supply Chain: Case: IBM Food Trust Blockchain and the Food Retail Supply Chain in Malta 2018. Bachelor’s Thesis, Lahti University of Applied Science, Lahti, Finland, 2018.
173. Bajwa, N.; Prewett, K.; Shavers, C.L. Is your supply chain ready to embrace blockchain? *J. Corp. Account. Financ.* **2020**, *31*, 54–64. [[CrossRef](#)]
174. Monrat, A.A.; Schelén, O.; Andersson, K. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access* **2019**, *7*, 117134–117151. [[CrossRef](#)]
175. Ahram, T.; Sargolzaei, A.; Sargolzaei, S.; Daniels, J.; Amaba, B. Blockchain technology innovations. In Proceedings of the 2017 IEEE Technology & Engineering Management Conference (TEMSCON), Santa Clara, CA, USA, 8–10 June 2017.
176. Lin, W.; Huang, X.; Fang, H.; Wang, V.; Hua, Y.; Wang, J.; Yin, H.; Yi, D.; Yau, L. Blockchain technology in current agricultural systems: From techniques to applications. *IEEE Access* **2020**, *8*, 143920–143937. [[CrossRef](#)]
177. Yang, H.; Xiong, S.; Frimpong, S.A.; Zhang, M. A consortium blockchain-based agricultural machinery scheduling system. *Sensors* **2020**, *20*, 2643. [[CrossRef](#)] [[PubMed](#)]
178. Guo, J.; Cengiz, K.; Tomar, R. An IOT and blockchain approach for food traceability system in agriculture. *Scalable Comput. Pract. Exp.* **2021**, *22*, 127–137. [[CrossRef](#)]
179. Salah, K.; Nizamuddin, N.; Jayaraman, R.; Omar, M. Blockchain-based soybean traceability in agricultural supply chain. *IEEE Access* **2019**, *7*, 73295–73305. [[CrossRef](#)]
180. Borah, M.D.; Naik, V.B.; Patgiri, R.; Bhargav, A.; Phukan, B.; Basani, S.G. Supply chain management in agriculture using blockchain and IoT. *Adv. Appl. Blockchain Technol.* **2020**, *60*, 227–242.
181. Eluubek kyzzy, I.; Song, H.; Vajdi, A.; Wang, Y.; Zhou, J. Blockchain for consortium: A practical paradigm in agricultural supply chain system. *Expert Syst. Appl.* **2021**, *184*, 115425. [[CrossRef](#)]
182. Yadav, V.S.; Singh, A. A systematic literature review of blockchain technology in agriculture. In Proceedings of the International Conference on Industrial Engineering and Operations Management, Pilsen, Czech Republic, 23–26 July 2019; IEOM Society International: Southfield, MI, USA, 2019; pp. 973–981.
183. Torky, M.; Hassanein, A.E. Integrating blockchain and the internet of things in precision agriculture: Analysis, opportunities, and challenges. *Comput. Electron. Agric.* **2020**, *178*, 105476. [[CrossRef](#)]
184. Mollah, M.B.; Zhao, J.; Niyato, D.; Lam, K.Y.; Zhang, X.; Ghias, A.M.; Koh, L.H.; Yang, L. Blockchain for future smart grid: A comprehensive survey. *IEEE Internet Things J.* **2020**, *8*, 18–43. [[CrossRef](#)]
185. Agung, A.A.G.; Handayani, R. Blockchain for smart grid. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 666–675. [[CrossRef](#)]
186. Alladi, T.; Chamola, V.; Rodrigues, J.J.; Kozlov, S.A. Blockchain in smart grids: A review on different use cases. *Sensors* **2019**, *19*, 4862. [[CrossRef](#)]
187. Fan, M.; Zhang, X. Consortium blockchain based data aggregation and regulation mechanism for smart grid. *IEEE Access* **2019**, *7*, 35929–35940. [[CrossRef](#)]
188. Wu, Z.; Liang, Y.; Kang, J.; Yu, R.; He, Z. Secure data storage and sharing system based on consortium blockchain in smart grid. *J. Comput. Appl.* **2017**, *37*, 2742.
189. Aggarwal, S.; Kumar, N. A consortium blockchain-based energy trading for demand response management in vehicle-to-grid. *IEEE Trans. Veh. Technol.* **2021**, *70*, 9480–9494. [[CrossRef](#)]
190. Zheng, D.; Deng, K.; Zhang, Y.; Zhao, J.; Zheng, X.; Ma, X. Smart grid power trading based on consortium blockchain in Internet of Things. In Proceedings of the International Conference on Algorithms and Architectures for Parallel Processing, Guangzhou, China, 15–17 November 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 453–459.
191. Gai, K.; Wu, Y.; Zhu, L.; Qiu, M.; Shen, M. Privacy-preserving energy trading using consortium blockchain in smart grid. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3548–3558. [[CrossRef](#)]
192. Mengelkamp, E.; Notheisen, B.; Beer, C.; Dauer, D.; Weinhardt, C. A blockchain-based smart grid: Towards sustainable local energy markets. *Comput. Sci. Res. Dev.* **2018**, *33*, 207–214. [[CrossRef](#)]
193. Bera, B.; Saha, S.; Das, A.K.; Vasilakos, A.V. Designing blockchain-based access control protocol in IoT-enabled smart-grid system. *IEEE Internet Things J.* **2020**, *8*, 5744–5761. [[CrossRef](#)]
194. He, S. Towards Practicalization of Blockchain-Based Decentralized Applications. Ph.D. Thesis, New Jersey Institute of Technology, Newark, NJ, USA, 2022.
195. Kim, J. Blockchain technology and its applications: Case studies. *J. Syst. Manag. Sci.* **2020**, *10*, 83–93.
196. Wang, H.; Li, H.; Smahi, A.; Xiao, M.; Li, S.Y.R. GBT-CHAIN: A System Framework for Solving the General Trilemma in Permissioned Blockchains. In *Distributed Ledger Technologies: Research and Practice*; ACM: New York, NY, USA, 2023.
197. Papadis, N. Stochastic Modeling and Optimization of Blockchain Networks. Ph.D. Thesis, Yale University, New Haven, CT, USA, 2023.
198. Guo, Z.; Qin, B.; Guan, Z.; Wang, Y.; Zheng, H.; Wu, Q. A High-Efficiency and Incentive-Compatible Peer-to-Peer Energy Trading Mechanism. *IEEE Trans. Smart Grid* **2023**, *15*, 1075–1088. [[CrossRef](#)]
199. Katz, J.; Lindell, Y. *Introduction to Modern Cryptography: Principles and Protocols*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2007.

200. Badertscher, C.; Ciampi, M.; Kiayias, A. Agile cryptography: A universally composable approach. In Proceedings of the Theory of Cryptography Conference, Taipei, Taiwan, 29 November 29–2 December 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 480–509.
201. Karanjai, R.; Collier, R.; Gao, Z.; Chen, L.; Fan, X.; Suh, T.; Shi, W.; Xu, L. Decentralized Translator of Trust: Supporting Heterogeneous TEE for Critical Infrastructure Protection. In Proceedings of the 5th ACM International Symposium on Blockchain and Secure Critical Infrastructure, Melbourne, VIC, Australia, 10–14 July 2023; pp. 85–94.
202. Andreina, S.; Bohli, J.M.; Karame, G.O.; Li, W.; Marson, G.A. Pots: A secure proof of tee-stake for permissionless blockchains. *IEEE Trans. Serv. Comput.* **2020**, *15*, 2173–2187. [[CrossRef](#)]
203. Li, R.; Wang, Q.; Wang, Q.; Galindo, D.; Ryan, M. SoK: TEE-assisted confidential smart contract. *arXiv* **2022**, arXiv:2203.08548.
204. Xie, H.; Zheng, J.; He, T.; Wei, S.; Hu, C. TEBDS: A Trusted Execution Environment-and-Blockchain-supported IoT data sharing system. *Future Gener. Comput. Syst.* **2023**, *140*, 321–330. [[CrossRef](#)]
205. Gencer, A.E.; Basu, S.; Eyal, I.; Van Renesse, R.; Sirer, E.G. Decentralization in bitcoin and ethereum networks. In Proceedings of the the 22nd International Conference on Financial Cryptography and Data Security, Nieuwpoort, Curaçao, 26 February–2 March 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 439–457.
206. Fernandez-Carames, T.M.; Fraga-Lamas, P. Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks. *IEEE Access* **2020**, *8*, 21091–21116. [[CrossRef](#)]
207. Gao, Y.; Chen, X.; Chen, Y.; Sun, Y.; Niu, X.; Yang, Y. A secure cryptocurrency scheme based on post-quantum blockchain. *IEEE Access* **2018**, *6*, 27205–27213. [[CrossRef](#)]
208. Chen, J.; Gan, W.; Hu, M.; Chen, C.M. On the construction of a post-quantum blockchain for smart city. *J. Inf. Secur. Appl.* **2021**, *58*, 102780. [[CrossRef](#)]
209. Saha, R.; Kumar, G.; Devgun, T.; Buchanan, W.; Thomas, R.; Alazab, M.; Kim, T.; Rodrigues, J. A blockchain framework in post-quantum decentralization. *IEEE Trans. Serv. Comput.* **2021**, *16*, 1–12. [[CrossRef](#)]
210. Acay, C.; Recto, R.; Gancher, J.; Myers, A.C.; Shi, E. Viaduct: An extensible, optimizing compiler for secure distributed programs. In Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation, Virtual, 20–25 June 2021; pp. 740–755.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.