A Survey on the Efficiency, Reliability, and Security of Data Query in Blockchain Systems

Qizhi Zhang[a,b,c,*], Yale He[a,b,c,*], Ruilin Lai[a,b,c], Zhihao Hou[a,b,c], Gansen Zhao[a,b,c,**]

[a] *School of Computer Science, South China Normal University (Shipai Campus), Guangzhou, Guangdong 51063, China*
[b] *Key Lab on Cloud Security and Assessment technology of Guangzhou, Guangzhou, Guangdong 51063, China*
[c] *SCNU & VeChina Joint Lab on BlockChain Technology and Application, Guangzhou, Guangdong 51063, China*

Abstract

With the rise of digital currencies, blockchain systems work for storing data efficiently and securely, creating a decentralized and tamper-proof digital platform. However, less works focus on data backtracking and supporting for many complex query functions. Therefore, we explore blockchain queries from both a technical and an application standpoint. Current methods to deliver are based on using distributed databases, data indexing structures, and various cryptographic algorithms to implement rich, verifiable, and secure query functions. As a result, the purpose of this survey is to provide some guidance on blockchain-related query technology. We analyze the issues of query efficiency, query reliability, and privacy protection of queries in blockchain. In addition we summarize some classic scenarios of query schemes in blockchain and discuss future research challenges that still need to be addressed by blockchain query technology.

Keywords: Blockchain Technology, Data Query, Data Structures, Query Security, Privacy Protection, Blockchain Applications

1. Introduction

With the rapid development of the Internet of Things (IoT) [1], supply chain management [2], Telematics [3] and other industries, data is needed as an event-driven carrier, but the authenticity, integrity, and data security cannot be guaranteed during data interaction. security. Therefore, new technologies are urgently needed to manage data effectively, that is, to guarantee that data are not leaked or stolen by others while verifying the integrity and correctness of data. Blockchain as a trusted machine in an untrustworthy environment has been proven to be valid in terms of data integrity, decentralization and distributed ledger technology [4]. After reviewing the literature, the conclusion can be easily reached that blockchain can be a reliable database that can be efficiently queried, verifiable and secure.

Firstly, blockchain is a series of data stored in a chained structure with a tamper-proof timestamp and the chain is connected to the previous block in the form of a hash pointer, which can be easily detected by other nodes once the block content is tampered with. Furthermore, blockchains use distributed ledgers and consensus algorithms (such as PoW [5], PoS [6], PBFT [7], etc.) mechanisms to bypass data storage fraud that guarantee data consistency. Despite blockchain's enormous promise for driving the cyber era's future, experts have discovered that its data traceability function is still far from ideal. To compensate for the absence of data querying functionality, two techniques are commonly used. The first is to copy the blockchain file to a SQL Server database [8, 9], such as Abe [10]. Another typical technique currently is to create a full node on a local server and then use the full node's RPC protocol to obtain data or to obtain all of the transaction data of the matching token through the log file generated by each transaction, such as etherscan [11]. The first method has the advantage of providing comprehensive blockchain data query capabilities, but the drawback is that SQL query performance declines or even becomes unloadable as the volume of data grows considerably, and data migration consistency is not assured. The second idea provides query performance, but it is limited by the fact that users are unable to verify the data's quality and give sophisticated query capabilities.

Additionally, as blockchain technology advances, many industries are adopting it as their underlying technology architecture. Walmart, for example, has implemented an IBM blockchain-based supply chain [12] tracking system to integrate retail food into a tamper-proof, transparent, and distributed network. Auditors must validate the data flow in the case of a food safety traceability event in order to determine the cause and scope of the food concern. Furthermore, in healthcare, patients frequently employ IoT-based wearable technology, with sensor end devices gathering, recording, and querying patient data, which can cause major security issues or even endanger patients' lives if the data is compromised or tampered with. The preceding example demonstrates that, while blockchain has a wide range of applications, it lacks verifiable and privacy-protecting capabilities for data searches by default, necessitating the use of other methods to address the aforesaid shortcomings.

We divide the existing issues into three categories: query ef-

*Q. Zhang and Y. He contributed equally to this work.
**Corresponding author
Email addresses: zhangqizhi@m.scnu.edu.cn (Qizhi Zhang), heyale@m.scnu.edu.cn (Yale He), 1139656140@qq.com (Ruilin Lai), houzhihao@m.scnu.edu.cn (Zhihao Hou), gzhao@m.scnu.edu.cn (Gansen Zhao)

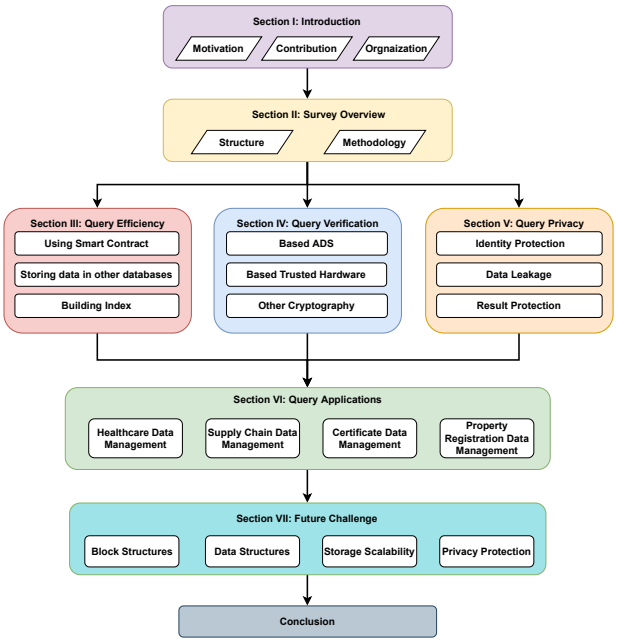

Figure 1: The roadmap of this survey

ficiency, query verifiability, and query privacy. To this end, we make three contributions:

- The survey evaluats the existing blockchain technologies and latest advances from the technical aspects of query efficiency, verifiable query, and query privacy, and thoroughly investigated their similarities and differences.
- Analyze the various applications of blockchain query technology and summarize existing technical solutions in some cases.
- The survey explores the latest technologies for query processing in blockchain to identify future trends and challenges.

The following is the format of this paper: in Section 2, we describe the core idea of our survey and the survey methodology. Section 3 covers the most recent strategies and advances for making blockchain queries more efficient. Verifiable query strategies are divided into numerous categories in Section 4. We offer privacy-preserving query approaches in blockchain in Section 5. Section 6 then presents blockchain query applications based on various scenarios. In Section 7 we identify future directions for research and challenges and conclude the paper in Section 8. The roadmap of the paper is depicted in Figure 1.

2. Survey Overview

This survey classified blockchain query technology into three categories: effective query technology, verifiable query technology, and query process privacy protection technology.

2.1. Survey Structure

The query efficiency issue is the first of three parts of our survey on the blockchain's query technology. The underlying data storage system, LevelDB [13], experiences excessive write performance but insufficient read performance when handling frequent query functions. Based on the presumption that both the full node and the client are honest nodes, existing query approaches provide more effective query services as well as richer query functions. Utilizing smart contracts, making the underlying data accessible to other databases, or altering the internal design of the blockchain to include suitable index structures are the three main strategies to increase query efficiency. While introducing the characteristics of other data structures, such as B-trees, B+-trees, prefix trees, and so on, adding an index structure typically preserves the Merkle Tree's attributes. Although this structure has the benefit of permitting real-time querying and guaranteeing that the query structure is unaltered, it frequently sacrifices storage space and query feature richness. The issue of unreliable query results is introduced by the other two strategies, which both leverage smart contracts and outreach databases.

The second point is the verifiability of the query results. Proposing authenticated data structures (ADS) [14] based on cryptographic techniques or other cryptographic-based techniques like multiparty signatures, database fingerprinting, generating trusted execution environments, etc., is currently the dominant trend. Verifiable inquiries guarantee the reliability of the query results, but they have a marginally greater storage and processing overhead.

With the assumption that both the full node and the client are semi-honest, identity leakage, data leakage, query statements, and query results can all be taken by attackers during transmission during the query process as the volume of data grows and the application scales. In order to protect data privacy, mainstream approaches frequently include storing encrypted data in cloud databases, indexing their files into blockchains, utilizing searchable encryption, private information retrieval methods, and differential privacy methods. This has the benefit of preventing the leakage of cryptographic data while ensuring the integrity of the data.

In each section, we will explore and review the literature from these three areas, listing the assumptions, technologies involved, and similarities and contrasts among the approaches. We conclude by listing the pertinent blockchain technology applications, outlining the unique issues and directions that were resolved in that scenario by blockchain technology, and providing the pertinent solutions.

2.2. Methodology

In this study, we have opted to analyze the overview of many features of blockchain inquiry technology using a systematic literature review approach [15], which involves the subsequent steps. In the beginning, we broke down the fundamental concept of blockchain query technology into three progressive categories: query efficiency, query verifiability, and query privacy. Finding relevant academic or research articles on blockchain

query technologies is the next step. We concentrate on peer-reviewed, pertinent articles of the highest caliber from reputable journals, conferences, publications, etc. Our sources for this work are from renowned publishers like IEEE, Elsevier, Springer, and others, as well as well-known academic search databases like Google Scholar, arXiv, and DBLP. To find relevant publications, use the search terms "blockchain query verifiable" "blockchain query privacy" and "blockchain query applications". In the following stage, we carefully sorted out the high-quality articles and excluded those that were not pertinent to this inquiry after filtering all the retrieved papers according to their titles. Finally, we organize and analyze these references methodically, organizing them into three progressive categories and corresponding application scenarios, and summarizing their contributions and deficiencies to give academics with precise proposals for the next stage of research.

3. Analysis in Blockchain Query Efficiency

LevelDB is currently the mainstream database in blockchain systems, based on the LSM Tree (Log-Structured Merge-Tree) [16] storage structure. However, Level-DB only supports simple key-value pair queries and does not support relational queries, equal-value queries, etc. Blockchain systems only support hash-based keyword queries with a single query method and need to follow a specific format (e.g. with 128-bit hexadecimal elements). Research in recent years has been divided into three major directions to solve these problems, the first is to drive data queries through smart contracts, the second is to store blockchain data in other databases, and the third is to build an index structure inside the blockchain.

3.1. Assumptions

The system has two types of entities: full nodes and light clients. Assume that both full nodes and light clients are honest. In this part, we only discuss the ideal case, in which the full node returns all data efficiently and honestly when the light client makes a request, and we ignore the possibility of harmful attacker activity.

3.2. Using Smart Contract

Abuhashim et al. [17] designed to put data into smart contracts on the blockchain by programming different smart contract function to support blockchain indexing and retrieval. This approach was not suitable for real-time transaction systems with large amounts of data, and when blockchain transactions reach 10 million records, the cost of storing in a smart contract is about 200 Ether, which is uneconomical.

Thabet et al. [18] proposed a smart contract-based query engine layer using Solidity mappings linked by unique integer identifiers, as it supports querying by directly accessing the mappings without traversing the entire dataset. The approach successfully solved the problem of storing and querying data, but required a large number of gas to execute on-chain queries with over 1000 entries. And no specific experimental data was given in the paper to support the conclusion that the queries are efficient.

Table 1: Comparison of various metrics for smart contract

Approach	Query Types	Gas Consumption(in Gwei)
Abuhashim et al. [17]	Key	315,000 ~ 335,000
Thabet et al. [18]	Key	800,000
Gürsoy et al. [19]	Key, Range, Aggregate	-
Chishti et al. [20]	Key, Range, Aggregate	2,248,593

Notations: Gas Consumption: Average of Gas required per query execution;

Gürsoy et al. [19] adopted to use the FastQeury smart contract on Ethereum to store and query pharmacogenomics data on the Ethereum blockchain, a scheme that exploited the uniqueness of attribute values for storage mapping, similar to a hash table, to achieve efficient key-value queries. From a storage perspective, FastQuery scales up to 10,000 inserted entries in the test; from a query perspective, the average query time to process 10,000 pieces of data toke 165 milliseconds. The amount of data stored in a smart contract was difficult to land in a large application. Also, due to the vulnerability of smart contracts, the limitations of gas and other features developed by Ethereum, storing and querying with smart contracts was by no means an easy task.

Currently, smart contracts on Ethereum are assisted in accessing on-chain data through indexing and search tools provided by external prophecy machines, and smart contracts can not directly access information on the blockchain. Chishti et al. [20] proposed an efficient, decentralized search mechanism that improves search efficiency by using the SMPT (Search Merkle-Patricia Trie) data structure and dividing blockchain data into disjoint subsets The search efficiency was improved by processing the data in parallel to support direct access to the blockchain data by smart contracts. The cost and limitations of gas are probably taken into account in this approach, which only goes up to a maximum number of 5000 block queries, which is impractical in a practical and growing blockchain.

In general, the use of smart contracts for querying data is still in its infancy, as there are too many limitations in smart contracts, including potential vulnerabilities in the smart contract itself, limitations in gas, blurring of function boundaries when converting Solidity language to Bytecode, and the small amount of data that can be stored in the contract itself. All of these issues affect the efficiency of smart contract queries, and we can see from the experimental section of these articles that smart contracts can only be used in certain scenarios with small amounts of data and are not generalizable. Table 1 compares the complexity of various smart contracts, mainly measured in terms of Gas. A more efficient way of querying data is described below.

3.3. Storing Blockchain data in other databases

EtherQL [21] was the first efficient query layer designed for Ethernet that supports SQL-like query statements, the system consists of four modules: a synchronization manager, a handler chain, a persistence framework, and a developer interface that

Table 2: Comparison of various metrics for storing to other database methods

Approach	Platform	Conccurent	Query Types	External Database
Li et al. [21]	Ethereum	T	Account, Transaction, Block	MongoDB
Pratama et al. [22]	Ethereum	T	Account, Transaction, Block	MongoDB
Wang et al. [23]	Ethereum	N	Account, Transaction, Block	SQLDB
S.Bragagnolo et al. [24]	Hyperledger Fabric	N	Account, Transaction, Block	Hadoop
Zhang et al. [25]	Ethereum	N	Account, Transaction, Block	ForkBase

Notations: Conccurent: Support for concurrent queries;

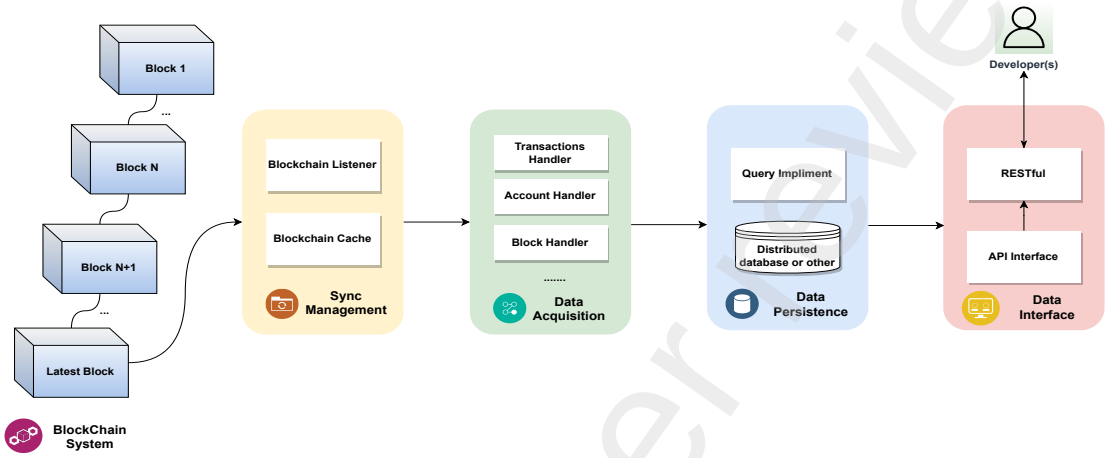

Figure 2: Architecture diagram for storing blockchain data to other databases

provided efficient query primitives for blockchain data including range queries and top-k queries. The data synchronization manager is used to fetch blockchain data from the Ethernet network, parse out the blockchain fields, store them in MongoDB and then provide a query interface to implement the query function with the help of an external database.

Pratama et al. [22] enhanced and extended the queries defined in EtherQL to support keyword queries, range search queries, aggregation queries, sorting queries (ascending or descending) and other functions. The system provided a RESTful API query interface to support DaaS (Data as a Service).

S.Bragagnolo et al. [24] proposed the use of big data techniques to enable the extraction and analysis of information in the blockchain. They used a Map/Reduce model parallelized indexing algorithm to synchronise the blockchain data into a relational database. Subsequent queries to the relational database would give information from the Ethernet blockchain, but the synchronisation time of this method is intolerable due to the fact that Ethernet produces blocks at an average rate of one block every 12 seconds.

ForkBase [23] and UStore [26] made some optimizations at the blockchain storage engine layer, both providing query interfaces from the underlying storage layer. ForkBase was an optimised version of UStore that supports features such as data version control, forking and tamper-proofing, and aims to push these features to the blockchain data layer.

Zhang et al. [25] used ForkBase, organized by Pos-Tree which combined the concepts of Merkle trees and B+ trees, as an external database and adds an external cache (Redis) to improve query performance, this approach allow for quick synchronization of copies from the blockchain database to the local node.

Overall these approaches are efficient, but as seen in Figure 2. there are delays in updating the data from the blockchain to other databases, which do not meet the needs of users to query data in real-time well. Second, external databases are not tamper-proof. Blockchain can protect the data on the chain from tampering, but it cannot protect the external database. If the external database is attacked, it is difficult for querying users to discover it because the data obtained by the user cannot be validated. Table 2 lists the different indicators of this approach. Therefore, efficiency alone is not comprehensive and the consistency and integrity of the query results with the blockchain data need to be considered, which will be detailed in the verifiable queries presented in Section 4.

3.4. Building a internal index in Blockchain

The index query in blockchain is to locate the block and then find a specific transaction based on the location of the block, so the index in the article is generally divided into two kinds, one is intra-block index and one is inter-block index. Usually, in order to reduce the time complexity of the query, the article will combine the two kinds of indexes, and Figure 3. shows the general architecture of the index query. The inter-block index is divided into two parts in the figure because the index in the blockchain is generally composed of an optimized Merkle-Patricia Tree, but since the blockchain is a single-chain structure we can also

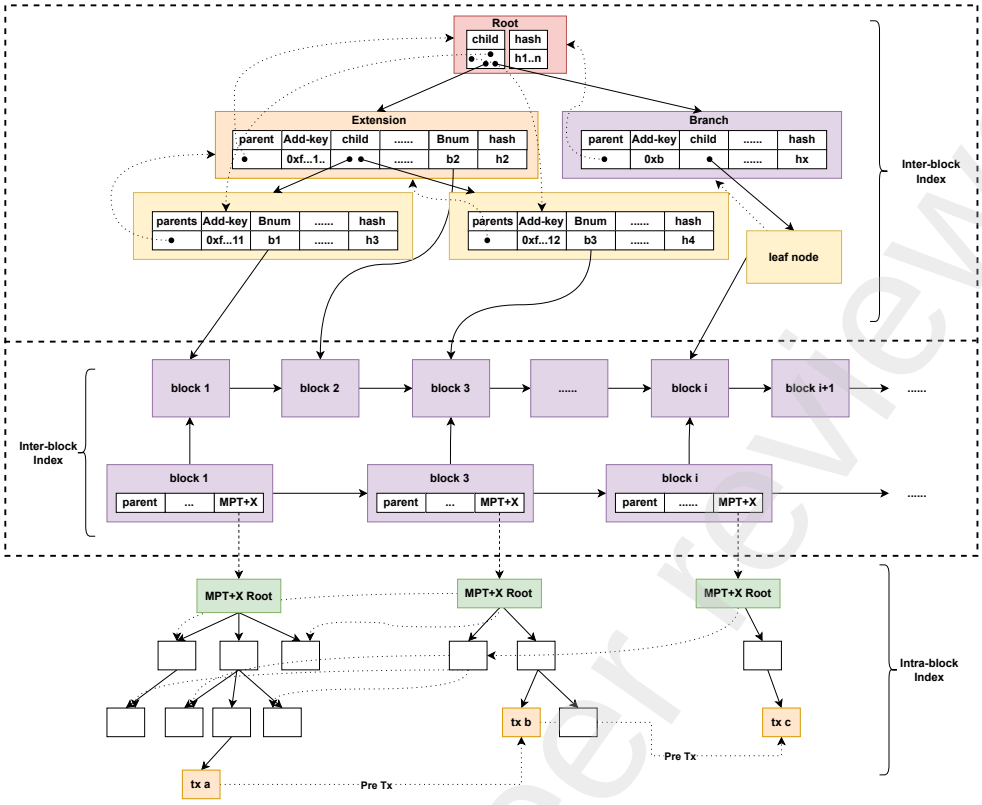

Figure 3: Architecture diagram for inter/intra index

use skip list or other forms of processing when dealing with the inter-block index.

Zeng et al. [27] proposed a static indexing approach by embedding indexed data in each transaction, which maintained the first 2^i transactions of the current transaction. This is a relatively simple, concurrency-supporting query method that does not involve modifying the Merkle Tree storage structure, but as the indexed data table may grow indefinitely it is necessary to consider compressing the fields to reduce space consumption.

While BlockchainDB [28] choosed to use RBTree, a combination of Red-black tree and Merkle tree, as the index structure. The updated status of transaction records can be checked through the blockchain, and when the updated version of a node exceeds a threshold, the article argued that even if the node have been deleted, it would not have an impact on the latest status, which is conducive to branch deletion to achieve pruning of indexes to compress disk space size.

S.Bragagnolo et al. [29] proposed an Ether Query Language, one that allow users to query relevant information from the blockchain by writing SQL-like statements. The internal implementation relies on a binary search tree (BST) as the index structure, which uses a two-dimensional array to improve the performance of querying data by mapping stored attribute values to their corresponding hashes. This implementation has limited support for querying information about smart contracts, as it does not allow querying contract properties or parameters in function calls, and the discussion of improving query performance is not supported by concrete data.

Huang et al. [30] designed a blockchain based on a doubly linked list, combining a Threaded Binary Search Tree (TBST) and AVL tree to propose a new search structure within the block. With the new data structure supporting time-range search, the starting position of the search can be found quickly and all transaction records can be searched from this position. However, this method does not give detailed experimental data and the efficiency of the search is open to question.

Du et al. [31] argued that as a multi-path self-balancing search tree, B-tree has better performance than binary search tree, balanced Binary(AVL) tree and other tree structures, and proposed EtherH as a hybrid index based on B-Tree and Skiplist that can provide Single-V queries and Range queries on the Ethernet blockchain. To make index construction more user-friendly, EtherH supports elastic data construction by selecting part of the blockchain data for index construction, achieving higher query performance and making it more universal.

Wan et al. [32] built an improve account grouping algorithm using a deep learning model that employs the GMPT (Group Merkel Patricia Tree) to speed up the querying of account status. However, GMPT does not support queries on historical transactions. For this reason, they modify it and construct an index directory BKV (B-Key-Value) [33] in combination with a B-Tree index structure. However, the article does not give a comparison and analysis of other relevant metrics about indexing performance beyond the time dimension.

Table 3: Comparison of different articles in terms of index

Approach	Platfrom	Query Types	Index Types	Block Structure	Index Data Structure	AQAC	SQL Interface
Zeng et al. [27]	Ethereum	Range	Intra	SL	Ordered Array	$O(\log_2 N)$	F
Jiao et al. [28]	Bitcion	Key	Intra	SL	MRB Tree	$O(\log_2 N)$	F
S.Bragagnolo et al. [29]	Ethereum	like SQL	Intra	SL	BST	$O(\log_2 N)$	T
Huang et al. [30]	Ethereum	Range	Intra	DL	TBST + AVL	$O(\log_2 N)$	F
Du et al. [31]	Ethereum	Range, Single-V	Intra + Inter	SL	B Tree	$O(\log_p N)$	F
Wan et al. [32]	Ethereum	Account	Intra	SL	GMPT	-	F
Wan et al. [33]	Ethereum	Historical Tx	Intra	SL	BVK Tree	-	F
Liu et al. [34]	Hyperledger Fabric	Historical Tx	Intra + Inter	SL	Abstact-Trie	-	F
Jia et al. [35]	Hyperledger Fabric	Historical Tx	Intra	SL	AB-M Tree	$O(N) \sim (\log_2 N)$	F
Xiao et al. [36]	Ethereum	Top-K, Range	Intra + Inter	SL	B+ Tree + MHT	$O(\log_p N)$	F
Zhu et al. [37]	SEBDB	like SQL	Intra + Inter	SL	B+ Tree + MHT	$O(\log_p N)$	T
Ruan et al. [38]	Hyperledger Fabric	Key	Intra + Inter	SL	Merkle DAG	$O(\log_p N)$	F
Pei et al. [39]	Ethereum	Semantic, Range, Fuzzy	Intra + Inter	SL	Merkle Semantic Tree	-	F
Xu et al. [40]	Bitcoin	Account, Historical Tx	Inter	SL	MPT	$O(\log_p N)$	F
You et al. [41]	Ethereum	Account, Historical Tx	Inter	SL	B+ Tree + MHT	O(N)	F
Xing et al. [42]	Ethereum	Account, Historical Tx	Intra + Inter	SL	B+ Tree + MHT	O(N)	F

Notations: AQAC:Average Query Algorithm Complexity; Intra: Intra-Block Index; Inter: Inter-Block Index; $O(\log_p N)$: p refers to the maximum number of index numbers inside each node; SL: Singly-linked List; DL: Doubly-linked List; MRB: Merkle Red-black; BST: Binary Search Tree; TBST: Threaded Binary Search Tree; GMPT: Group Merkle Patricia Tree; AB-M T: Adaptive Balanced Merkle Tree; MHT: Merkle Hash Tree; F: False; T: True;

Liu et al. [34] contributed a novel block structure that divides the block body into an index layer and a data layer. Two types of indexes are proposed at the index layer, aggregation of data within blocks and efficient querying of data between blocks. This approach maintains the association of the original data in each block, but at the cost of increasing the time cost of constructing the blocks.

SE-Chain [35] system represented to store each thing (transaction event) in the data layer as AB-M tree (Adaptive Balanced Merkle Tree). AB-M tree is a combination of Balanced Binary Tree and Merkle Tree, which is a variation of the Merkle Tree. The data processing layer reduces the data redundancy by de-duplication algorithm and improves the AB-M tree storage scalability.

Both EBTree [36] and SEDBD [37] combined B+ tree indexes, and the B+ tree was introduced in EBTree to achieve orderly insertion and management of nodes, and the nodes that meet the requirements are written to LevelDB through the persistence module, then the query module locates, processes the data and returns the results. SEDBD introduced B+ trees to achieve fast indexing in chronological order and proposed three index structures: block index, table attributes index, and hierarchical index to achieve support for SQL statement query and multi-table join query on-chain.

LineageChain [38] is a fine-grained, secure and efficient blockchain traceability system implemented on a Forkbase based storage system. Merkle DAG is established to store the historical evolution state of all transactions during the block generation process, and it is inefficient to traverse all nodes on the graph during the query process so the paper proposes to build Deterministic Append-only Skip List (DASL) index with a skip table structure on top of DAG to replace the storage layer of Fabric.

Pei et al. [39] argued that it is not only important to focus on on-chain data queries and present a real-time query scheme for hybrid storage blockchain systems. A new semantic keyword extraction algorithm is proposed to extract key semantic information from the under-chain to map with on-chain data and construct a backward index by sorting according to the weight of the lexical items, and the index is generated together with the creative block. As building the index in the block does not rely on any specific underlying structure, it is compatible with various blockchain systems. This indexing technique is called Merkle Semantic Trie and supports semantic queries, range queries, and even fuzzy queries.

Xu et al. [40] constructed the index structure of MPTChain for the education certificate management problem, which supports both the efficient query of historical transactions and the query of account history. You et al. [41] improved the index structure of MPTChain to form the Account Status Tree (AST) while proposing two-hybrid index structures for transaction chain indexing to improve the transaction retrieval performance of the blockchain. Both index structures use a compressed prefix tree structure to reduce the storage space of the index and are actually inspired by the Merkle-Patricia Tree structure of Ethereum.

Xing et al. [42] proposed an efficient indexing structure based on subchain account transaction chains (SCATC), being for it divides N blocks into multiple subchains. They suggested creating an optimized inter-AST block index structure for transaction information of the same account via hash pointers, and the information status of all accounts is included in the last block of each sub-chain. By this method, only the last block of the sub-chain needs to be queried during the query process to obtain the existence of the account in the sub-chain. Converting the query method of traversing the transaction chain to a sub-chain query method (similar to a jump table structure) effectively reduces the access to irrelevant block data and reduces the computational overhead.

In general, the method of storing blockchain data in other databases is simple to implement, scalable, and query capable. However, it does not support real-time searches, and because

blockchain updates are asynchronous, frequent writes may result in the data consistency and integrity of many copies that are not secured by the blockchain itself. Building an index structure within the blockchain is difficult to implement and weak in scalability, but it supports real-time queries. Because the data on the chain is synchronized with the update of the index, the next operation can only start after the data is written to the index and LevelDB, so the index state and data of the blockchain are kept consistent after the system crash and restart. Since the index only needs to save the corresponding attributes, it takes up less storage space than storing blockchain data in other databases. Table 3 lists the different indexing approaches to address query efficiency.

4. Verifiable Queries in Blockchain System

Verifiable query processing means that light nodes can verify the correctness of the query data and ensure that the resulting data has not been tampered with. Verifiable query techniques are generally implemented based on cryptography, which includes ADS, trusted hardware implementations and other cryptographic methods.

4.1. Assumptions

The full nodes and the light clients are the two types of entities in this area of the system. This section's premise is that some full nodes are dishonest and light nodes are trustworthy. Full nodes may listen to the blockchain network but not perform synchronous block updates, and they may even maliciously tamper with certain information in a block to achieve a specific goal. We must evaluate not only the query efficiency of the full node in this section, but also the validity of the result after the query, and the consistency of the data in the full node and the blockchain network, because malicious conduct of full nodes is a possibility.

4.2. Based Authenticated Data Structure

Xu et al. [43] first investigated the problem of verifiable query processing for blockchain databases, proposing the vChain framework to ensure the integrity of Boolean range queries for lightweight users. An accumulator-based ADS scheme is proposed to transform numerical attributes into aggregate values, so that dynamic aggregation can be performed for arbitrary query attributes. However, the solutions are still not perfectly suited to data-intensive scenarios, as they still do not support efficient access to the history of operations on raw data.

Zhu et al. [44] proposed a method to support verifiable aggregation queries on the Blockchain system, which can alleviate storage and computation costs for users while ensuring the integrity of query results, and is the first paper to support aggregation queries. A general and capacity-efficient authenticated aggregate tree (GCA^2 Tree) based on the accumulator ADS is proposed to support various verifiable multidimensional aggregation (count, maximum, minimum, maximum, average) queries. The GCA^2 tree is applied to the blockchain system to satisfy the immutability characteristics of the blockchain, on which efficient verifiable multidimensional aggregation queries are supported. This further extends the diversity of verifiable queries.

Dai et al. [45] proposed LVQ, a lightweight verifiable query method that reduces both storage requirements and network overhead. By storing the BF (Bloom Filter) hashes in headers, the amount of data stored for verification is reduced. They designe the BMT (BF Integrated Merkle Tree) structure, which reduces the query, storage and network overheads by merging multiple contiguous BFs so that ideally only one BF is transmitted in the query result to indicate the existence of an address in thousands of blocks.

Matteo et al. [46] Unlike vChain and GCA^2-tree, this article's solution did not rely on set accumulators. The article choosed to use MR-trees, i.e. authenticated data structures designed for spatial data, so that queries can be verified by simpler cryptographic hash computations. Furthermore, with this approach, we can exploit the spatial locality of the data to reduce the size of the authenticated object. However, the limitations of the article only support queries within blocks, inter-block queries are still not extended.

Zhang et al. [47] took the first step towards investigating authenticated range queries in hybrid storage blockchains.A new ADS was proposed that can not only be maintained by smart contracts with optimized gas performance, called the Gas Efficient Merkel Merge Tree (GEM^2-tree), which can be efficiently maintained in a blockchain while efficiently supporting authentication range queries.

Zhe et al. [48] proposed VFChain, a verifiable and auditable federal learning framework based on a blockchain system. They provide verifiability by blockchain selecting a committee to collectively aggregate models and record verifiable proofs in the blockchain. To improve the efficiency of verifiable proof search and to support secure rotation of committees a Dual Skip Chain (DSC) is proposed, an ADS, can be traversed forward and backward from any block establishing two types of interconnections.

4.3. Based Trusted Hardware

Trusted hardware referred to providing a trusted execution environment on an trustless node where clients can interact in a trusted environment. Shao et al. [49] considered Intel SGX (Software Guard Extensions) to be trusted execution environments that provide cryptographic and integrity protection for security-sensitive computations, enabling important code and data to run securely on untrusted system software. If a blockchain system that focuses on data trust can be combined with SGX, which provides a trusted execution environment, it is bound to enhance and optimise trusted data sharing in blockchain systems. They propose an approach based on Intel SGX trusted hardware to provide trusted query services for blockchain light nodes, which enables zero-cost trusted queries by combining the Merkle Tree with ADS (Authenticated Data-Structure) so that light nodes do not have to handle any authentication work.

Table 4: Comparison of various metrics on verifiable query methods in blockchain

Approach	Platform	Verifiable Method	Data Structure	Query Type	Supported Index
Xu et al. [43]	Ethereum	acc-ADS	MHT + Skip List	Boolean Range	T
Zhu et al. [44]	Ethereum	acc-ADS	GCA^2-Tree	Aggregate	T
Dai et al. [45]	Bitcoin	acc-ADS	BMT	Historical Tx	T
Matteo et al. [46]	Ethereum	tree-ADS	MRT	Range	T
Zhang et al. [47]	Ethereum	tree-ADS	GEM^2-Tree	Range	T
Zhe et al. [48]	Hyperledger Fabric	Federal learning	DSC	Historical Tx, Block	T
Shao et al. [49]	Ethereum	Intel SGX, ADS	MHT	Range	T
Pang et al. [50]	Ethereum	Intel SGX, ADS	MHT	Range	T
Jade et al. [51]	Bitcoin	Intel SGX	ODS-TX-IDX	Key	T
Zhou et al. [52]	Ethereum	Intel SGX	VeriDB	like SQL	T
Wu et al. [53]	Ethereum	Database fingerprint	MPT	Historical Tx	F
Rahman et al. [54]	Bitcoin	multi-signature	B+ tree	Key, Range	T
Han et al. [55]	Hyperledger Fabric	Vassago framework	Multi-chain	Historical Tx	F

Notations: acc-ADS:Accumulator-ADS; MHT: Merkle Hash Tree; GCA^2-Tree:General and capacity-efficient authenticated aggregate Tree; BMT: Bloom Filter Integrated Merkle Tree; MRT:;GEM^2-tree: Gas Efficient Merkle Merge Tree; DSC: Dual Skip Chain; ODS-TX-IDX: Oblivious Data Structure-Transaction-TransactionID; MPT: Merkle Patricia Tree; F: False; T: True;

Pang et al. [50] organized data hierarchically in trusted (enclave) and untrusted memory, using ADS to provide query validation for lightweight clients. Jade et al. [51] proposed an efficient transaction query scheme for lightweight clients by running Intel SGX Enclave on the full node. The scheme combined the oblivious data structure [56] technique with prefix trees to provide efficient transaction search and verification functionality for Bitcoin clients.

Zhou et al. [52] designed and implemented VeriDB, a verifiable database based on trusted hardware SGX that supports relational tables, multiple access methods, and general SQL queries. VeriDB further provides verifiable query execution support for general SQL queries, and virtual container technology in a trusted hardware environment provides a secure and isolated environment for blockchain [57]. However, the security of query execution in trusted environments relies on the approval of trusted hardware vendors, and their performance can be limited in some ways. For example, the memory available in the enclave provided by SGX is small and as the number and size of programs increases, memory pages need to be swapped to ensure security, but this will result in reduced efficiency.

4.4. Others Cryptography Methods

In addition to verifiable queries based on ADS, some articles proposed the use of database fingerprinting, multiple signatures, smart contracts and other schemes to achieve data query integrity.

Wu et al. [53] provided an efficient and verifiable query service for blockchain, a Verifiable Query Layer (VQL) architecture is proposed, a three-tier architecture cloud query service. To ensure the validity of the data in the middleware, the blockchain generates a fingerprint which is a cryptographic hash value based on the different content and attributes of each database, and only needs to verify the correctness of the fingerprint can prevent any forgery in the middleware. At the same time the system provides a simplified solution for validating the query results, the user only needs to check the validity of the database fingerprints based on the Merkle proofs.

Rahman et al. [54] introduced a framework based on search privacy protection and providing verifiability of search results on the client side. It supports precision and range search through the use of order-preserving encryption (OPE) technology, using a Guillou-Quisquater (GQ) based multi-signature scheme to validate query results. The key benefit of the article proposed framework is that it can verify the authenticity of the query result regardless of the underlying structure of the blockchain.

Furthermore, in addition to verifiable queries between the client and the blockchain, Han et al. [55] proposed an efficient and authenticated approach across multiple blockchains. Vassago is a system consisting of a four-layer architecture that guarantees data provenance queries for trusted cross-chain transactions by defining dependencies connected into a complete path. The key to achieving trusted associations is that the cross-chain behaviour of each node should be consistent with each other.

4.5. Summary

The main methods of verifiable queries in blockchain are ADS, trusted environment based approaches, and cryptography based approaches. Tree-based ADS and cryptographic accumulator-based ADS are the two most common types of ADS. To secure the integrity and correctness of data, verifiable queries based on trusted environments often use Intel SGX paired with ADS. Asymmetric encryption techniques are used to implement other cryptographic methods such as database fingerprinting, digital signatures, and multi-party signatures. The data structure, query type, and index support metrics of various approaches are compared in Table 4.

5. Privacy Preserving Queries

Blockchain technology has to make public some information within blocks as verification in order to maintain data consistency among decentralized nodes and to reach consensus on

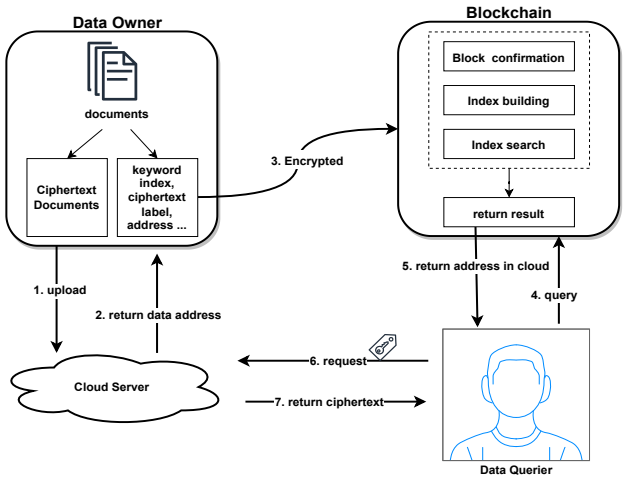

Figure 4: Privacy preserving query structure

transactions. This public information often leads to privacy issues such as identity leakage of auditors and leakage of user transaction information during the query process.

The privacy protection in the query process is divided into two stages, which are the privacy protection in the query process and the privacy protection in the query result. Issues such as identity privacy as well as data leakage occur during the query process. Identity privacy leakage refers to the possibility that the client (light node) may leak some sensitive information when making a query, which can be used to infer the true identity corresponding to the blockchain address. Data leaking occurs when a user's query statement and the returned search results are shown in plain text, potentially exposing the data owner's sensitive information. Privacy protection of query results means that the client (light node) may infer sensitive information about a specific person's transaction information after getting the query results, for example, certain transactions reflect the user's consumption level, living habits, etc.

5.1. Assumptions

The full nodes, the light clients (users), and the data owners are often included in the systems in this part. The basis of this section is that both full nodes and light clients are semi-honest or even malicious. The full node listens to the blockchain network and synchronizes the most recent blocks, providing transactions or accounts to be queried. During the query process, the full node and light client may exchange any information they obtain, such as transaction details or other node addresses, in order to assess their true identities. As well as the full node may return partial information or even incorrect results of the search results. Following the retrieval of the query result, the light client may use the known partial information to deduce the identity or sensitive information of a person of interest.

5.2. Identity privacy protection

Chen et al. [58] were the first to employ a blockchain-based user management scheme for searchable cryptosystems that use smart contracts to enable decentralized multi-user dynamic management and data sharing schemes for several users in a group. Users (clients) do not have to worry about third parties stealing, tampering with, or leaking the user's identity, key, or other data because they record the index of encrypted data into the blockchain in advance in the form of contracts. When the volume of transactions surpasses 3,000 per second, or when 250 users continuously launch a query within 300 seconds, the system stability deteriorates and the likelihood of query failure rises.

Linov et al. [59] proposed to protect the identity of the light client (auditor), query statements, and query results using Hadoop and synchronized Ethernet clients at a semi-trusted server using symmetric cryptography. When users submit like-SQL queries, these statements are transformed into MapReduce tasks and locate blocks through a B+ tree-based inter-block index for efficient querying. To achieve user identity privacy protection, the client and server share the key, extend the query scope and protect the query statements in cipher text during query processing, and then hide the real identity of the client through the proxy service; when the query is returned, it is also sent to the client in cipher text, and the client uses the key to decrypt and filter the extended information for data processing.

Ge et al. [60] introduced a privacy-preserving algorithm based on variable factor perturbation of Bloom filters to dynamically adjust the privacy-preserving performance. Combining the query history and the threshold of privacy leakage tolerability of light nodes, a privacy-preserving technique based on hybrid policy Nash equilibrium is developed to successfully safeguard the privacy information of light nodes. However, the research does not define user privacy requirements or node computing and storage resources, and the resource consumption issue is not examined in depth.

5.3. Data leakage protection

Access control management for users on the blockchain network is accomplished by providing a valid Membership Service Provider (MSP) on the federated chain to produce, verify, and revoke identity-related certificates. These outsourced node data are encrypted and then stored on the blockchain to strengthen MSP security. Tahir et al. [61] proposed a Permissioned Blockchain-Based Searchable Encryption framwork (PBSE) to support keyword search and data insertion functions for encrypted data, which generates different search tokens for searches of the same keyword through probabilistic trapdoor-based hidden search, resisting distinguishability attacks and also preventing passive attacks.

Cai et al. [62] introduced keyword search of encrypted data using a searchable symmetric encryption (SSE) approach, and the query is confirmed using a hash digest. The metadata in the encrypted file uploaded by the client is anchored as evidence on the blockchain by the server node, and it can be located using the index built on the file identity. However, in practice, this strategy is unsuitable because the smart contract deployment cost is approximately $5.24 and the retrieval cost is approximately $4.42, using an exchange rate of 1 ether = USD$ 2841 at the time of writing.

Table 5: Comparison of various metrics for privacy preserving queries

Privacy stage	Authors	Platform	Crypto.M.	Query Types	Index Content	Ciphertext Location
Identity	Chen et al. [58]	Hyperledger Fabric	SE	Multi-Key	document ID	CSP
	Linov et al. [59]	Ethereum	PIR	like SQL	block number	Hadoop
	Ge et al. [60]	Bitcoin	Bloom filter	Single-Key	block address	-
Data Leakage	Tahir et al. [61]	Hyperledger Fabric	SE	Single-Key	master key + document ID	MSP
	Cai et al. [62]	Ethereum	SE	Single-Key	document ID	CSP
	Yu et al. [63]	Ethereum	SE	Single-Key	document ID	CSP
	Hu et al. [64]	Hyperledger Fabric	SE	Single-Key	document ID	IPFS
	Jiang et al. [65]	Ethereum	SE	Multi-Key	document ID	CSP
	Yang et al. [66]	Ethereum	HSS	Multi-Key, Range, Semantic	document ID	CSP
	Ma et al. [67]	Ethereum	ABE, SE	Multi-Key	ciphertext Keywords	CSP
	Jiang et al. [68]	Ethereum	SE	Single-Key	document ID	IPFD
	Chen et al. [69]	Hyperledger Fabric	ECC	Multi-Key	ciphertext Keywords	IPFS
	Li et al. [70]	Ethereum	PIR, DP	Single-Key	block data	-
	Xie et al. [71]	Bitcoin	PIR	Single-Key	keywords of transactions	-
	Li et al. [72]	Bitcoin	SGX, d-DP	Single-Key	block data	CSP
Query Result	Yang et al. [73]	Hyperledger Fabric	DP	Aggregate	-	-
	Zhao et al. [74]	Ethereum	DP	Multi-Key	time ordered	CSP
	Xu et al. [75]	FISCO BCOS	DP	Multi-Key	-	CSP
	Kan et al. [76]	Ethereum	DP	Single-Key, Aggregate	-	CSP

Notations: Cypto. Method: Cryptographic Method; SE: Searchable Encryption; CSP: Cloud Service Provider; PIR: Private Information Retrieval; MSP: Membership Service Providers; IPFS: InterPlanetary File System; HSS: heuristic semantic searching; ABE: Attribute-Based Encryption; ECC: Elliptic Curve Cryptography; DP: Differential Privacy;

Yu et al. [63] designed an optimized dynamic symmetric searchable encryption scheme based on blockchain technology using cryptographic primitives and the immutable nature of blockchain. The scheme preserved the forward security of the pre-update operation and uses smart contracts to develop a verifiable scheme to ensure that the updated results are easy to verify. They constructed a inverted index to map different data associated with the same keywords to a single chain index and then located them once, which greatly reduces the computational cost compared to using each file to index different keywords as proposed by Cai et al. [62].

Hu et al. [64] presented two decentralized searchable symmetric encryption (SSE) methods for public and private chains, as well as access control for many users on private chains, allowing authorized users to view and search data in a shared manner. Their approach differs from previous verified SSE schemes in that the search results after a query are valid and immutable, and the data owner does not need to double-check the search results.

Existing [62, 77] approaches can expand single-keyword search to multi-keyword settings by conducting it numerous times and taking the intersection of the results, but such methods have privacy and efficiency issues. Jiang et al. [65] initially applied multi-keyword-identifier mappings to raw data and generate labels for these mappings. A Bloom filter is also built to capture all high-frequency keywords, locate low-frequency keywords, then filter the encrypted database using these low-frequency keywords to minimize the search space and increase multi-keyword search performance.

Yang et al. [66] and schemes (Cai et al. [62], Tahir et al. [61], Hu et al. [64]) both used searchable encryption method, but the first two schemes only support single-keyword matching on documents, while Yang et al. proposed a privacy-preserving nonlinear matching method based on retrieval heuristic semantics (PPWNM) to perform semantic search, and also propose a new blockchain verification method to verify the correctness of search results using the similarity proofs generated during PPWNM. It implemented multi-keyword query, top-k query, and semantic query.

Ma et al. [67] suggested a blockchain-based trusted data sharing architecture based on attribute-based encryption (ABE), searchable encryption, and a multi-keyword ciphertext retrieval encryption algorithm. They introduced the Random Oracle Model (ROM) to ensure that, even if the keywords are the same, different types of ciphertext indexes are used to ensure that no one can successfully guess the metrics, and that the correctness and integrity of the retrieval results are ensured by verifying the ciphertext labels. This technique, on the other hand, ignored the efficiency of data updates and access control permission updates, which is inconvenient in real-world situations.

Jiang et al. [68] established a blockchain-based framework for searching outsourced cryptographic data that is publicly verified. The system allows for the encrypted index to be stored on a decentralized blockchain while the outsourced data is sent to the cloud or the Interplanetary File System (IPFS). At the same time, they devised a stealth authorisation technique to allow for the distribution of access authorizations while maintaining privacy. Smart contracts allow authorized users to query data and validate the integrity of the results, however the search time for smart contracts needs to be improved.

Chen et al. [69] proposed a secure data transfer scheme based on HyperLedger Fabric blockchain using symmetric encryption algorithm (AES) to store the encrypted index in the blockchain and store the encrypted data in Interplanetary File System (IPFS), similar to the approach of Jiang et al. [68] for data storage. Chaincode is used to automate data calls during the query

process, and the elliptic curve signing algorithm (ECDSA) is used to sign the messages transmitted by all parties to ensure data integrity.

Li et al. [70] proposed a privacy-preserving scheme for the full-node design based on private information retrieval (PIR), which implements pointer-based PIR search via keywords (hashes) and introduces differential privacy to constrain the leakage problem of memory access patterns to provide quantifiable access privacy to the client. They constructed an optimized version of the differentially private information retrieval (DPIR) scheme, which aims to reduce communication overhead and provide query security by recognizing that the results of any two consecutive query sequences of the same length are indistinguishable.

Xie et al. [71] offered an improved PIR for efficient query transaction retrieval in order to improve query data privacy. They rearranged the data stored on the entire node and build two databases, one for the index database and the other for the transaction file repository, to facilitate light client retrieval of transaction data. In comparison to BIP37, the communication overhead is slightly lower (setting the false positive rate set to 0.05 percent).

To some extent, both the private information retrieval (PIR) [78, 79] and the oblivious RAM (ORAM) [80] protocols can address the full-node access mode privacy leakage problem. However, both protocols are not applicable in lightweight blockchain search scenarios due to excessive overhead or security limitations. Li et al. [72] presented a d-differential privacy solution based on Intel SGX trusted hardware to secure lightweight client privacy, making it impossible for an attacker to obtain the true access pattern during the data retrieval phase. However, the increased enclave launch time and authentication time caused by developing trusted hardware is the cost of security.

5.4. Query Result Protection

Yang et al. [73] described a blockchain-based data sharing generic system framework that employs checkpoint privacy approaches to store, validate, and adaptively allocated privacy budget consumption via smart contracts based on the privacy and data utility requirements of the data owner. To preserve dataset privacy, the system additionally provides anonymization services to federate sensitive data of data owners to the cloud.

Zhao et al. [74] used differential privacy techniques to protect the end result of querying data on the blockchain. By minimizing privacy loss through reusing noise algorithms, the algorithm can partially or completely reuse the previous noisy answers when accessing the same query type multiple times, effectively protecting personal data privacy. Unlike Yang et al. [73] the former uses Gaussian noise for easier privacy analysis of combinations of different privacy-preserving algorithms, while the disadvantage of the Laplacian mechanism used by the latter is that the sum of independent Laplacian random variables does not obey the Laplacian distribution.

Xu et al. [75] proposed a blockchain-based differential privacy data publishing framework and two separate differential privacy data publishing protocols for two different types of published data (histogram publishing and anonymous data publishing). The suggested protocols can guarantee that the noise results provided by the collector satisfy both the requestor's demand for data usefulness and the data owner's requirement for privacy protection by involving the nodes of the blockchain network in the Laplace noise selection process. However, both protocols' time consumption in the verification interaction process should be reduced.

Kan et al. [76] developed a customized privacy management platform that complies with the General Data Protection Regulation (GDPR) standards by conducting differentiated privacy queries via smart contracts and storing the query results on the blockchain. The architecture allows users to send raw data that is integrated with differential privacy by selecting the level of data perturbation via a DApp; it also allows users to perform aggregated queries, where the differential privacy algorithm adds Laplacian noise to the queried results and delivers the noise-added results to the user.

5.5. Summary

The privacy protection provided by blockchain in the query process can be separated into three categories: identity privacy, data leakage during transmission, and query result privacy. Anonymity technology, server proxy, or encryption are commonly employed to mask the real identity of the client querier in identity privacy protection. To ensure that no data leakage happens during transmission, searchable encryption and private information retrieval are commonly used to encrypt and index the data during transmission. This ensures that a semi-honest (curious) server cannot know the precise query statement and query result. In the data return phase, the use of differential privacy mechanisms prohibits semi-honest (curious) clients from inferring private information about an individual person from certain group characteristics. Table 5 analyzes several techniques to preserve privacy in different articles, types of queries, indexed content, and certain sorts of third-party servers by applying multiple cryptographic approaches to help make our blockchain queries more secure.

6. Blockchain Application scenarios

6.1. Healthcare Data Management

6.1.1. Problem Describe

Electronic medical record (EMR) [81] captures what happens to a patient (dizziness, shock, vomiting, etc.) where (hospital, operating room, private home, nursing home, etc.) and who (local doctor, hospital doctor, nurse, etc.) treats the patient (prescribes medication, injections, surgery, etc.), or through wearable e-health devices driven in part by the Internet of Things, EMR automatically collects real-time body metrics of the user, for example through continuous blood glucose testing [82], heart rate testing [83], etc. Applications of EMR include issues such as access control, efficient query, data sharing, and security privacy.

Table 6: Blockchain based health care systems

Approach	System Name	Implement
Zaabar et al. [92]	HealthBlock	access control, privacy
Zou et al. [93]	SPChain	high throughput, privacy
Azbeg et al. [94]	BlockMedCare	access control, security
Kim et al. [95]	Dynamichain	dynamic consent, security
Madine et al. [96]	appXchain	cross-chain interoperability

Table 7: Blockchain based supply chain systems

Approach	Platform	Fields
Uddin et al. [104]	Hyperledger Fabric	pharmaceutical industry
Agrawal et al. [105]	Ethereum	textile and clothing industry
Casino et al. [106]	Ethereum	food industry
Ho et al. [107]	Hyperledger Fabric	airline industry
Song et al. [108]	Ethereum	agriculture

6.1.2. Blockchain-based Solutions

Blockchain technology is appropriate in the above case since it is a non-tamperable, decentralized data sharing platform. The main principle behind blockchain is that the user is solely responsible for the data in his electronic medical record, as well as the only one who can provide others access to it [84]. The blockchain supports fine-grained access authorization that supports flexible queries, protects private data without user authorization [85], and provides better interoperability as a data broker in a standardized format enabling reliable interoperability across enterprises [86]. Combine features of Merkle trees and prefix dictionary trees in blocks with associated identification for each block header attribute. Implement sharable data for doctors, hospitals, or other data requesters with different identities to perform multiple combinations of queries with conditions according to their needs [87].

To improve the storage efficiency, Zhu et al. [88] used the convolution to improve the structure of Merkle Tree to achieve compressed tree height, which makes the storage more compact and efficiency. Smart contracts can verify the validity of query results [89], which is a key reference base for doctors in the diagnosis process, to secure the integrity and non-repudiation of the retrieved data. Peng et al. [90] presented a zero-knowledge proof approach for generating verifiable proofs while maintaining user privacy. Medical records contain a wide range of sensitive private information that cannot be maintained in plaintext in the blockchain, therefore data privacy protection is unavoidable. To safeguard privacy, Chen et al. [91] advocated using the K-anonymity technique to preproces the data and searchable encryption to protect data that is available but not visible on the chain.

Existing electronic medical record management systems such as Zaabar et al. [92], Zou et al. [93], Azbeg et al. [94], Kim et al. [95], Madine et al. [96]. Table 6 summarizes the examples mentioned above.

6.2. Supply Chain Data Management

6.2.1. Problem Describe

The supply chain refers to the cross-regional processing of raw materials from the agricultural or manufacturing sector to the finished product, which is packaged and delivered to the consumer. Supply chain management is a complex operation that necessitates the proper recording, sharing, and tracing of all information along the supply chain, including data regarding the manufacturing, processing, storage, shipping, and retailing of food goods [97]. Using inverter air conditioners as an example, each air conditioner has about 2,000 parts, and each supplier must retain records for the manufacture process of these parts, which is likely to include several nations and regions. In such a case, the standard and quality of each item are difficult to guarantee, and the source is difficult to identify when machine failure happens, and there are issues with data tampering, low query efficiency, auditing difficulty, and high management and verification expenses in the typical traceability system [98, 99].

6.2.2. Blockchain-based Solutions

The usage of blockchain can prevent information tampering during the exchange of goods and construct an irreversible decentralized database by uploading goods information onto the chain in real time, allowing each participant to verify the quality status of items through a single application [100]. The on-chain off-chain blockchain storage approach [101] can be used to store just the hash value of the original data onto the blockchain and construct the index to tackle the problem of large data volume in supply chain data management. Malik et al. [102] featured a generalized hierarchical network supply chain architecture that uses the sharding technique to effectively safeguard sensitive information of consumers and stakeholders while also improving query efficiency. Tsang et al. [103] proffered the PoSCS consensus algorithm, which considers transportation time, stakeholder weight analysis, and shipment volume in the supply chain to select the verifier in a probabilistic manner, making the consensus method more consistent with the scenario. The consensus algorithm is more suited to scenario-based needs and ensures the blockchain's tamper-proofness.

Existing supply chain blockchain management technologies are applied in many industries, e.g., Uddin et al. [104], Agrawal et al. [105], Casino et al. [106], Ho et al. [107], Song et al. [108]. Table 7 summarizes the examples mentioned above.

6.3. Certificate Data Management

6.3.1. Problem Describe

Certificate authorities (CAs) keep the public keys of the certificates that users need to query to authenticate users and establish TLS [109] connections with servers using existing electronic certificates. However, the CA-authorized certificate may be subjected to a variety of attacks that result in failure [110], such as the server's private key being exposed or the attacker falsifying user identification information to tamper with the data maintained by CA. As a result, managing certificate data necessitates both the storage of large volumes of data as well as the requirement to assure data reliability and user privacy.

Table 8: Blockchain based certificate management systems

Approach	Platform	Implement
Nguyen et al. [116]	Hyperledger Fabric	anti-forgery, verification
Dilshan et al. [116]	Hyperledger Fabric	transparency, security
Ali et al. [117]	Hyperledger Fabric	efficiency, security
Lodagala et al. [118]	Hyperledger Fabric/Ethereum	verification, security
Shen et al. [119]	Hyperledger Fabric	identity privacy protection

Table 9: Blockchain based registration management systems

Approach	Platform	Implement
Nandi et al. [127]	Ethereum	secure, tamper-proof
Veeramani et al. [128]	Hyperledger Fabric	secure, transparency
Mendi et al. [129]	Hyperledger Fabric	secure, transparency
Thakur et al. [130]	Hyperledger Fabric	privacy, tamper-proof
Biswas et al. [131]	Ethereum	transparency, tamper-proof

6.3.2. Blockchain-based Solutions

To tackle the above challenge, the solution must provide both distributed and anonymous storage, and blockchain technology appears to be a good fit for this demand situation. When storing large amounts of data, we typically utilize highly compact data structures such as CertLedger [111], which stores the certificate encoding in the Merkle Hash Tree (MHT) in the block, PROCESS [112], which uses the Counting Garbled Bloom Filter (CFBF), and CRchain [113] , which uses cuckoo filters. LightLedger [114] provided a novel domain authentication technique that allows remote clients to securely and efficiently query blockchain data, CAs can no longer approach any certificate autonomously without the owner's approval, and Liu et al. [115] designed a private data authentication protocol that secures participants' identities without requiring third-party participants' signatures.

There are many examples in the literature where blockchains are used to manage various certificates, e.g., Nguyen et al. [116], Dilshan et al. [116], Ali et al. [117], Lodagala et al. [118] and Shen et al. [119]. Table 8 summarizes the above instances.

6.4. Property Registration Data Management

6.4.1. Problem Describe

The land property registration, for example, indicates who has the right to use the land, how long the land is available, and the restrictions of land usage. Registry data is normally maintained and controlled by government departments. Paper documents are commonly used as proof and basis for business transactions (buying, selling, leasing, etc.) in traditional land registration, however paper receipts are vulnerable to damage, loss, manipulation, or even falsification, resulting in significant land ownership disputes [120]. Furthermore, because there are several parties engaged in the land acquisition process [121], land registration takes time (lawyers, notaries, brokers, agents, appraisers, government agencies, etc.) Several documents must be processed and verified, as well as site inspections, during the procedure.

6.4.2. Blockchain-based Solutions

As blockchain itself is tamper-proof and multiple copy storage can solve the problems of paper document corruption, loss, and interfering to some extent for us. To reduce the possibility of document forgery and fraud Yadav et al. [122, 123] suggested two new consensus algorithms to reduce the communication overhead and an an efficient hash table search approach. They also adopted a new sidechain architecture [124] that stores metadata in the publicly accessible main chain and non-transactional data in the sidechain to reduce the storage consumption and authentication time of the main chain and improve the query efficiency of land registration data. Ameyaw et al. [125] used smart contracts to eliminate dishonest middlemen to shorten and simplify the land registration process. Soner et al. [126] employed a framework for autonomous land registry record maintenance that uses smart contracts to check the regularity of the loan procedure and make the buying and selling process easier for both parties.

The following are the relevant literature on land registration systems, e.g., Nandi et al. [127], Veeramani et al. [128], Mendi et al. [129], Thakur et al. [130], Biswas et al. Table 9 summarizes the above instances. [131].

6.5. Summary

It is not difficult to find that the core requirements of all the above application scenarios are to require data immutability, ensure data correctness, and improve the efficiency of data query and verification. Blockchain technology can be a good fit for the above requirements, as blockchain provides immutability, tamper-proof, and persistence to the data stored in it. The most obvious advantage of blockchain is that it enables the process of building trust between mutually distrustful entities through mathematical methods, enabling interoperability among multiple entities. Usually, blockchain storage costs are more expensive than traditional centralized systems, so many scholars will propose to put important resources or indexes into the blockchain, and other complicated data can be kept in encrypted form to prevent data leakage or theft by others. Finally, smart contracts can run on the blockchain, meaning that we can hand over the tedious and complex third-party intermediated things to smart contracts for completion, a feature that is particularly advantageous in the supply chain as well as land registration system applications.

7. Future Challenge

Based on the existing query strategies for blockchain data provided in the previous sections, we elaborate on prospective prospects and problems for future study in this section. We categorize the challenges as follows: data structure, block structure, storage scalability, privacy protection during querying, and legal liability.

7.1. Block Structures

The general query step in a standard blockchain query system starts with the latest block and iterates over each block to

see if the exact data required by the query exists, and if not, the next block is queried. Although using the block-by-block query method can ensure data integrity, it is often inefficient. So it is necessary to identify or index the blocks in a certain way. For example, we can use B+tree indexing, skip list, or even construct the blockchain as a double-linked list to allow backward and forward traversal. So the first challenge is deciding what kind of structure to use to index the blocks in order to increase query efficiency.

7.2. Data Structures

The most direct way to efficiently handle blockchain queries is to combine distributed databases with blockchain queries. However, this method does not guarantee data tamperability on these databases, and the most recent data queried by users may not be available because the synchronization module has not yet entered the next synchronization cycle. To solve these issues and accomplish efficient querying, new data structures must be designed, such as integrating MHT with B+ Tree and putting a hash pointer on the leaf nodes Merkle Tree to construct a transaction history data chain for each transaction data to query historical transactions.

Furthermore, the blockchain's data structure should not only provide efficient access but also maintain the consistency of data. If we execute data queries outside of the blockchain, we has to provide an additional validation phase to check the blockchain's results. To simplify this process, we can use MRT, BMT and ADS-based Merkle optimization trees, for example, to integrate the properties of the search data structure with the validation data structure. All of these ways necessarily raise the computational overhead as well as the query time, thus the second problem is to build the data structure in such a way that it provides both fast access to stored data while also ensuring data integrity and dependability in an efficient verification manner.

7.3. Storage Scalability

With the increasing volume of data, we must take into account storage scalability. There are three trends to consider. The first is to improve the data structure, for example, using convolutional operations to optimize the Merkle Tree [88], which not only expands the storage capacity of a single tree while also compressing the tree's height. Second, the introduction of sharding technology can generally solve the problem of blockchain scalability [132]. Through sharding, we can construct specific divisions of various data so that different subchains manage different data to increase the performance of data searches. The third trend is the cross-chain exchange, which is similar to sharding in that different category of data are put into different chains, and when users make queries, cross-chain technology allows them to access the whole data returns. Both sharding and cross-chaining operations require a large number of connection operations to interact, and cross-chaining operations involve consistency, data structure configuration, and trust model selection issues, so the challenge is how to ensure high efficiency and high-security issues are waiting for further research by scholars.

7.4. Privacy Protection

There are a plethora of cryptographic algorithms that enhance query privacy, and the privacy-preserving solutions that have been developed thus far are woefully inadequate for the above-mentioned practical objectives. Complex cryptographic primitives normally necessitate a substantial amount of processing when it comes to indexing efficiency. When generating index structures or storing files with searchable encryption, for example, the average running time is proportional to the number of files, suggesting that the computational cost increases as the number of files increases. As a result, addressing the problem or introducing efficient cryptographic primitives may be viable options. When it comes to establishing efficient search capabilities, we can see that single-term searches in an inverted index are used by the majority of privacy-preserving blockchain-based systems. Advanced query capabilities such as aggregate and range searches, on the other hand, should be investigated further. Existing differential privacy approaches, for example, have extremely strong assumption premises that demand a substantial degree of randomization in the query, resulting in a dramatic drop in data availability. Improving the usefulness of data under weak assumption premises is one of the prospective study paths.

8. Conclusion

Blockchain has received great attention from scholars and enterprises because of its decentralized and tamper-proof features. This survey reviews the existing query schemes related to blockchain and provides a comprehensive analysis of query efficiency, query correctness, and query privacy, respectively. With the growing demand for blockchain query methods, we summarize four popular scenarios of such demand so that scholars can conduct research along with these scenarios. Also in order to explore the full potential of blockchain technology in the query direction, such as in query efficiency, reliability, and security, we propose four future directions for this purpose: block structure, data structure, storage scalability, and privacy protection. Through this comprehensive research study, we hope that our analysis and proposed challenges can provide further research ideas for blockchain query systems.

References

[1] S. Madakam, V. Lake, V. Lake, V. Lake, et al., Internet of things (iot): A literature review, Journal of Computer and Communications 3 (05) [2015] 164. doi:10.4236/jcc.2015.35021.

[2] S. Malik, V. Dedeoglu, S. S. Kanhere, R. Jurdak, Trustchain: Trust management in blockchain and iot supported supply chains, in: 2019 IEEE International Conference on Blockchain (Blockchain), 2019, pp. 184–193. doi:10.1109/Blockchain.2019.00032.

[3] S.-O. LEE, H. JUNG, B. Han, Security assured vehicle data collection platform by blockchain: Service provider's perspective, in: 2019 21st International Conference on Advanced Communication Technology (ICACT), 2019, pp. 265–268. doi:10.23919/ICACT.2019.8701965.

[4] R. Kumar, R. Sharma, Leveraging blockchain for ensuring trust in IoT: A survey, Journal of King Saud University - Computer and Information Sciences [2021]. doi:10.1016/j.jksuci.2021.09.004.

[5] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, Decentralized Business Review [2008]. doi:10.2139/ssrn.3440802.

[6] S. King, S. Nadal, Ppcoin: Peer-to-peer crypto-currency with proof-of-stake, self-published paper, August 19 (1) [2012].

[7] M. Castro, B. Liskov, et al., Practical byzantine fault tolerance, in: OsDI, Vol. 99, 1999, pp. 173–186. doi:10.1145/571637.571640.

[8] K.-B. Yue, K. Chandrasekar, H. Gullapalli, Storing and querying blockchain using SQL databases, Information Systems Education Journal 17 (4) [2019] 24.

[9] K.-B. Yue, K. Chandrasekar, H. Gullapalli, Querying Bitcoin Blockchain Using SQL, in: Proceedings of the EDSIG Conference ISSN, Vol. 2473, 2018, p. 3857.

[10] J. Tobey, Abe: block browser for bitcoin and similar currencies [2011].

[11] E. Team, Etherscan: The ethereum block explorer, https://etherscan.io [2017].

[12] L. Mearian, Q&A: Walmart's Frank Yiannas on the Use of Blockchain for Food Safety, Computerworld 1 [2018].

[13] K. Tulkinbekov, M. Pirahandeh, D.-H. Kim, CLeveldb: Coalesced leveldb for small data, in: 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN), IEEE, 2019, pp. 567–569. doi:10.1109/ICUFN.2019.8806187.

[14] R. Tamassia, Authenticated data structures, in: European symposium on algorithms, Springer, 2003, pp. 2–5. doi:10.1007/978-3-540-39658-1_2.

[15] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, S. Linkman, Systematic literature reviews in software engineering – A systematic literature review, Information and Software Technology 51 (1) [2009] 7–15, special Section - Most Cited Articles in 2002 and Regular Research Papers. doi:10.1016/j.infsof.2008.09.009.

[16] P. O'Neil, E. Cheng, D. Gawlick, E. O'Neil, The log-structured merge-tree (lsm-tree), Acta Informatica 33 (4) [1996] 351–385. doi:10.1007/s002360050048.

[17] A. Abuhashim, C. C. Tan, Smart contract designs on blockchain applications, in: 2020 IEEE Symposium on Computers and Communications (ISCC), 2020, pp. 1–4. doi:10.1109/ISCC50000.2020.9219622.

[18] N. A. Thabet, N. Abdelbaki, Efficient quering blockchain applications, in: 2021 3rd Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2021, pp. 365–369. doi:10.1109/NILES53778.2021.9600533.

[19] G. Gürsoy, C. M. Brannon, M. Gerstein, Using ethereum blockchain to store and query pharmacogenomics data via smart contracts, BMC medical genomics 13 (1) [2020] 1–11. doi:10.1186/s12920-020-00732-x.

[20] M. S. Chishti, F. Sufyan, A. Banerjee, Decentralized on-chain data access via smart contracts in ethereum blockchain, IEEE Transactions on Network and Service Management 19 (1) [2022] 174–187. doi:10.1109/TNSM.2021.3120912.

[21] Y. Li, K. Zheng, Y. Yan, Q. Liu, X. Zhou, Etherql: a query layer for blockchain system, in: International Conference on Database Systems for Advanced Applications, Springer, 2017, pp. 556–567. doi:10.1007/978-3-319-55699-4_34.

[22] F. A. Pratama, K. Mutijarsa, Query support for data processing and analysis on ethereum blockchain, in: 2018 International Symposium on Electronics and Smart Devices (ISESD), 2018, pp. 1–5. doi:10.1109/ISESD.2018.8605476.

[23] S. Wang, T. T. A. Dinh, Q. Lin, Z. Xie, M. Zhang, Q. Cai, G. Chen, B. C. Ooi, P. Ruan, Forkbase: An efficient storage engine for blockchain and forkable applications, Proc. VLDB Endow. 11 (10) [2018] 1137–1150. doi:10.14778/3231751.3231762.

[24] S. Bragagnolo, M. Marra, G. Polito, E. Gonzalez Boix, Towards scalable blockchain analysis, in: 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), 2019, pp. 1–7. doi:10.1109/WETSEB.2019.00007.

[25] Z. Zhang, Y. Zhong, X. Yu, Blockchain storage middleware based on external database, in: 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), 2021, pp. 1301–1304. doi:10.1109/ICSP51882.2021.9408752.

[26] A. Dinh, J. Wang, S. Wang, G. Chen, W.-N. Chin, Q. Lin, B. C. Ooi, P. Ruan, K.-L. Tan, Z. Xie, et al., Ustore: a distributed storage with rich semantics [2017]. arXiv:1702.02799.

[27] L. Zeng, W. Qiu, X. Wang, H. Wang, Y. Yao, Z. Yu, Transaction-based static indexing method to improve the efficiency of query on the blockchain, in: 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), 2021, pp. 780–784. doi:10.1109/ICAICA52286.2021.9497966.

[28] T. Jiao, D.-R. Shen, T.-Z. NIE, et al., Blockchaindb: querable and immutable database, Journal of Software 30 (09) [2019] 2671–2685. doi:10.13328/j.cnki.jos.005776.

[29] S. Bragagnolo, H. Rocha, M. Denker, S. Ducasse, Ethereum query language, in: 2018 IEEE/ACM 1st International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), 2018, pp. 1–8. doi:10.1145/3194113.3194114.

[30] T.-L. Huang, J. Huang, An efficient data structure for distributed ledger in blockchain systems, in: 2020 International Computer Symposium (ICS), IEEE, 2020, pp. 175–178. doi:10.1109/ics51289.2020.00043.

[31] P. Du, Y. Liu, Y. Li, H. Yin, L. Zhang, EtherH: A Hybrid Index to Support Blockchain Data Query, Association for Computing Machinery, New York, NY, USA, 2021, p. 72–76. doi:10.1145/3472634.3472653.

[32] L. Wan, A query optimization method of blockchain electronic transaction based on group account, in: International Conference on Big Data Analytics for Cyber-Physical-Systems, Springer, 2020, pp. 1358–1364. doi:10.1007/978-981-33-4572-0_196.

[33] L. Wan, An optimization method for blockchain electronic transaction queries based on indexing technology, in: International Conference on Big Data Analytics for Cyber-Physical-Systems, Springer, 2020, pp. 1273–1281. doi:10.1007/978-981-33-4572-0_183.

[34] M. Liu, H. Wang, F. Yang, An efficient data query method of blockchain based on index, in: 2021 7th International Conference on Computer and Communications (ICCC), IEEE, 2021, pp. 1539–1544. doi:10.1109/ICCC54389.2021.9674708.

[35] D.-Y. Jia, J.-C. Xin, Z.-Q. Wang, H. Lei, G.-R. Wang, Se-chain: A scalable storage and efficient retrieval model for blockchain, Journal of Computer Science and Technology 36 (3) [2021] 693–706. doi:10.1007/s11390-020-0158-2.

[36] H. XiaoJu, G. XueQing, H. ZhiGang, Z. LiMei, G. Kun, Ebtree: A b-plus tree based index for ethereum blockchain data, in: Proceedings of the 2020 Asia Service Sciences and Software Engineering Conference, 2020, pp. 83–90. doi:10.1145/3399871.3399892.

[37] Y. Zhu, Z. Zhang, C. Jin, A. Zhou, Y. Yan, Sebdb: semantics empowered blockchain database, in: 2019 IEEE 35th international conference on data engineering (ICDE), IEEE, 2019, pp. 1820–1831. doi:10.1109/icde.2019.00198.

[38] P. Ruan, T. T. Anh Dinh, Q. Lin, M. Zhang, G. Chen, B. Chin Ooi, Revealing every story of data in blockchain systems, ACM Sigmod Record 49 (1) [2020] 70–77. doi:10.1145/3422648.3422665.

[39] Q. Pei, E. Zhou, Y. Xiao, D. Zhang, D. Zhao, An efficient query scheme for hybrid storage blockchains based on merkle semantic trie, in: 2020 International Symposium on Reliable Distributed Systems (SRDS), 2020, pp. 51–60. doi:10.1109/SRDS51746.2020.00013.

[40] Y. Xu, S. Zhao, L. Kong, Y. Zheng, S. Zhang, Q. Li, Ecbc: A high performance educational certificate blockchain with efficient query, in: International Colloquium on Theoretical Aspects of Computing, Springer, 2017, pp. 288–304. doi:10.1007/978-3-319-67729-3_17.

[41] Y. You, L. Kong, Z. Xiao, Y. Zheng, Q. Li, Hybrid indexing scheme supporting blockchain transaction tracing, Comput Intergrated Manuf Syst 25 (04) [2019] 978–984. doi:10.13196/j.cims.2019.04.021.

[42] X. Xing, Y. Chen, T. Li, Y. Xin, H. Sun, A blockchain index structure based on subchain query, Journal of Cloud Computing 10 (1) [2021] 1–11. doi:10.1186/s13677-021-00268-0.

[43] C. Xu, C. Zhang, J. Xu, VChain: Enabling Verifiable Boolean Range Queries over Blockchain Databases, in: Proceedings of the 2019 International Conference on Management of Data, SIGMOD '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 141–158. doi:10.1145/3299869.3300083.

[44] Y. Zhu, Z. Zhang, C. Jin, A. Zhou, Enabling generic verifiable aggregate query on blockchain systems, in: 2020 IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS), 2020, pp. 456–465. doi:10.1109/ICPADS51040.2020.00066.

[45] X. Dai, J. Xiao, W. Yang, C. Wang, J. Chang, R. Han, H. Jin, Lvq: A lightweight verifiable query approach for transaction history in bitcoin, in: 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS), 2020, pp. 1020–1030. doi:10.1109/ICDCS47774.

2020.00096.
[46] M. Loporchio, A. Bernasconi, D. D. F. Maesa, L. Ricci, Authenticating spatial queries on blockchain systems, IEEE Access 9 [2021] 163363–163378. doi:10.1109/ACCESS.2021.3132990.
[47] C. Zhang, C. Xu, J. Xu, Y. Tang, B. Choi, Gem^2-tree: A gas-efficient structure for authenticated range queries in blockchain, in: 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019, pp. 842–853. doi:10.1109/ICDE.2019.00080.
[48] Z. Peng, J. Xu, X. Chu, S. Gao, Y. Yao, R. Gu, Y. Tang, Vfchain: Enabling verifiable and auditable federated learning via blockchain systems, IEEE Transactions on Network Science and Engineering 9 (1) [2022] 173–186. doi:10.1109/TNSE.2021.3050781.
[49] Q. Shao, S. Pang, Z. Zhang, C. Jing, Authenticated range query using sgx for blockchain light clients, Vol. 12114 LNCS, Jeju, Korea, Republic of, 2020, pp. 306 – 321, authenticated data structures;Empirical studies;Execution environments;Large-scale applications;Query results;Security analysis;State of the art;Storage resources;. doi:10.1007/978-3-030-59419-0_19.
[50] S. Pang, Q. Shao, Z. Zhang, C. Jin, Authqx: Enabling authenticated query over blockchain via intel sgx, Vol. 12114 LNCS, Jeju, Korea, Republic of, 2020, pp. 727 – 731, authenticated data structures;Execution environments;Large-scale applications;Performance issues;Query authentications;Secure memory;State of the art;Traditional industry;. doi:10.1007/978-3-030-59419-0_45.
[51] Y. Niu, C. Zhang, L. Wei, Y. Xie, X. Zhang, Y. Fang, An efficient query scheme for privacy-preserving lightweight bitcoin client with intel sgx, in: 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9013131.
[52] W. Zhou, Y. Cai, Y. Peng, S. Wang, K. Ma, F. Li, VeriDB: An SGX-Based Verifiable Database, Association for Computing Machinery, New York, NY, USA, 2021, p. 2182–2194. doi:10.1145/3448016.3457308.
[53] H. Wu, Z. Peng, S. Guo, Y. Yang, B. Xiao, Vql: Efficient and verifiable cloud query services for blockchain systems, IEEE Transactions on Parallel and Distributed Systems 33 (6) [2022] 1393–1406. doi:10.1109/TPDS.2021.3113873.
[54] M. S. Rahman, I. Khalil, N. Moustafa, A. P. Kalapaaking, A. Bouras, A blockchain-enabled privacy-preserving verifiable query framework for securing cloud-assisted industrial internet of things systems, IEEE Transactions on Industrial Informatics [2021] 1–1doi:10.1109/TII.2021.3105527.
[55] R. Han, J. Xiao, X. Dai, S. Zhang, Y. Sun, B. Li, H. Jin, Vassago: Efficient and authenticated provenance query on multiple blockchains, in: 2021 40th International Symposium on Reliable Distributed Systems (SRDS), 2021, pp. 132–142. doi:10.1109/SRDS53918.2021.00022.
[56] X. S. Wang, K. Nayak, C. Liu, T. H. Chan, E. Shi, E. Stefanov, Y. Huang, Oblivious data structures, in: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, 2014, pp. 215–226. doi:10.1145/2660267.2660314.
[57] V. Atul, Get into the zone: Building secure systems with arm trustzone technology, White Paper, Texas Instruments [2013].
[58] Y. Chen, J. Bai, Y. Hao, S. Liao, Z. Yi, H. Zhang, Blockchain-based dynamic group management for multiple keywords searchable encryption technology, in: 2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2020, pp. 1–6. doi:10.1109/CyberC49757.2020.00011.
[59] S. Linoy, H. Mahdikhani, S. Ray, R. Lu, N. Stakhanova, A. Ghorbani, Scalable privacy-preserving query processing over ethereum blockchain, in: 2019 IEEE International Conference on Blockchain (Blockchain), 2019, pp. 398–404. doi:10.1109/Blockchain.2019.00061.
[60] L. Ge, T. Jiang, A privacy protection method of lightweight nodes in blockchain, Security and Communication Networks 2021 [2021]. doi:10.1155/2021/2067137.
[61] S. Tahir, M. Rajarajan, Privacy-preserving searchable encryption framework for permissioned blockchain networks, in: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), IEEE, 2018, pp. 1628–1633. doi:10.1109/Cybermatics_2018.2018.00272.
[62] C. Cai, J. Weng, X. Yuan, C. Wang, Enabling reliable keyword search in encrypted decentralized storage with fairness, IEEE Transactions on Dependable and Secure Computing 18 (1) [2021] 131–144. doi:10.1109/TDSC.2018.2877332.
[63] Y. Guo, C. Zhang, X. Jia, Verifiable and forward-secure encrypted search using blockchain techniques, in: ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020, pp. 1–7. doi:10.1109/ICC40277.2020.9148612.
[64] S. Hu, C. Cai, Q. Wang, C. Wang, Z. Wang, D. Ye, Augmenting encrypted search: A decentralized service realization with enforced execution, IEEE Transactions on Dependable and Secure Computing 18 (6) [2021] 2569–2581. doi:10.1109/TDSC.2019.2957091.
[65] S. Jiang, J. Cao, J. A. McCann, Y. Yang, Y. Liu, X. Wang, Y. Deng, Privacy-preserving and efficient multi-keyword search over encrypted data on blockchain, in: 2019 IEEE International Conference on Blockchain (Blockchain), IEEE, 2019, pp. 405–410. doi:10.1109/blockchain.2019.00062.
[66] W. Yang, B. Sun, Y. Zhu, D. Wu, A secure heuristic semantic searching scheme with blockchain-based verification, Information Processing Management 58 (4) [2021] 102548. doi:10.1016/j.ipm.2021.102548.
[67] X. Ma, C. Wang, X. Chen, Trusted data sharing with flexible access control based on blockchain, Computer Standards & Interfaces 78 [2021] 103543. doi:10.1016/j.csi.2021.103543.
[68] S. Jiang, J. Liu, L. Wang, S.-M. Yoo, Verifiable search meets blockchain: A privacy-preserving framework for outsourced encrypted data, in: ICC 2019-2019 IEEE International Conference on Communications (ICC), IEEE, 2019, pp. 1–6. doi:10.1109/icc.2019.8761146.
[69] C.-L. Chen, J. Yang, W.-J. Tsaur, W. Weng, C.-M. Wu, X. Wei, Enterprise data sharing with privacy-preserved based on hyperledger fabric blockchain in iiot's application, Sensors 22 (3) [2022] 1146. doi:10.3390/s22031146.
[70] X. Li, F. Ahmed, L. Wei, C. Zhang, Y. Fang, Protecting access privacy in ethereum using differentially private information retrieval, in: GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1–6. doi:10.1109/GLOBECOM42002.2020.9348108.
[71] Y. Xie, C. Zhang, L. Wei, Y. Niu, F. Wang, Private transaction retrieval for lightweight bitcoin client, in: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), 2019, pp. 440–446. doi:10.1109/BLOC.2019.8751352.
[72] X. Li, Z. Yang, L. Wei, C. Zhang, Protecting access privacy for bitcoin lightweight client using trusted hardware, in: 2019 IEEE/CIC International Conference on Communications in China (ICCC), IEEE, 2019, pp. 706–711. doi:10.1109/iccchina.2019.8855891.
[73] M. Yang, A. Margheri, R. Hu, V. Sassone, Differentially private data sharing in a cloud federation with blockchain, IEEE Cloud Computing 5 (6) [2018] 69–79. doi:10.1109/MCC.2018.064181122.
[74] Y. Zhao, J. Zhao, J. Kang, Z. Zhang, D. Niyato, S. Shi, K.-Y. Lam, A blockchain-based approach for saving and tracking differential-privacy cost, IEEE Internet of Things Journal 8 (11) [2021] 8865–8882. doi:10.1109/JIOT.2021.3058209.
[75] L. Xu, T. Bao, L. Zhu, Blockchain empowered differentially private and auditable data publishing in industrial iot, IEEE Transactions on Industrial Informatics 17 (11) [2020] 7659–7668.
URL 10.1109/tii.2020.3045038
[76] N. Khan, A. Lahmadi, Z. Kräussl, R. State, Management plane for differential privacy preservation through smart contracts, in: 2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA), 2020, pp. 1–8. doi:10.1109/AICCSA50499.2020.9316507.
[77] Y. Zhang, R. H. Deng, X. Liu, D. Zheng, Outsourcing service fair payment based on blockchain and its applications in cloud computing, IEEE Transactions on Services Computing 14 (4) [2018] 1152–1166. doi:10.1109/tsc.2018.2864191.
[78] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, Private information retrieval, in: Proceedings of IEEE 36th Annual Foundations of Computer Science, IEEE, 1995, pp. 41–50. doi:10.1006/jcss.1999.1689.
[79] D. Jiachen, Y. Nenghai, L. Xianzheng, Z. Weiming, Bitcoin-based payment protocol for private information retrieval, Journal of Cyber Security 4 (6) [2019] 9. doi:10.19363/J.cnki.cn10-1380/tn.2019.11.01.

[80] E. Stefanov, M. V. Dijk, E. Shi, T.-H. H. Chan, C. Fletcher, L. Ren, X. Yu, S. Devadas, Path oram: an extremely simple oblivious ram protocol, Journal of the ACM (JACM) 65 (4) [2018] 1–26. doi:10.1145/3177872.

[81] X. Zhang, S. Poslad, Blockchain support for flexible queries with granular access control to electronic medical records (emr), in: 2018 IEEE International Conference on Communications (ICC), 2018, pp. 1–6. doi:10.1109/ICC.2018.8422883.

[82] T. M. Fernández-Caramés, I. Froiz-Míguez, O. Blanco-Novoa, P. Fraga-Lamas, Enabling the Internet of Mobile Crowdsourcing Health Things: A Mobile Fog Computing, blockchain and iot based continuous glucose monitoring system for diabetes mellitus research and care, Sensors 19 (15) [2019]. doi:10.3390/s19153319.

[83] B. Pradhan, S. Bhattacharyya, K. Pal, Iot-based applications in healthcare devices, Journal of healthcare engineering 2021 [2021]. doi:10.1155/2021/6632599.

[84] D. D. F. Maesa, P. Mori, Blockchain 3.0 applications survey, Journal of Parallel and Distributed Computing 138 [2020] 99–114. doi:10.1016/j.jpdc.2019.12.019.

[85] H. Huang, X. Sun, F. Xiao, P. Zhu, W. Wang, Blockchain-based ehealth system for auditable ehrs manipulation in cloud environments, Journal of Parallel and Distributed Computing 148 [2021] 46–57. doi:10.1016/j.jpdc.2020.10.002.

[86] A. R. Lee, M. G. Kim, I. K. Kim, Sharechain: Healthcare data sharing framework using blockchain-registry and fhir, in: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2019, pp. 1087–1090. doi:10.1109/BIBM47256.2019.8983415.

[87] J. Li, W. Dun, Range query in blockchain-based data sharing model for electronic medical records, in: Journal of Physics: Conference Series, Vol. 1634, IOP Publishing, 2020, p. 012035. doi:10.1088/1742-6596/1634/1/012035.

[88] H. Zhu, Y. Guo, L. Zhang, An improved convolution merkle tree-based blockchain electronic medical record secure storage scheme, Journal of Information Security and Applications 61 [2021] 102952. doi:10.1016/j.jisa.2021.102952.

[89] A.-S. Kleinaki, P. Mytis-Gkometh, G. Drosatos, P. S. Efraimidis, E. Kaldoudi, A blockchain-based notarization service for biomedical knowledge retrieval, Computational and structural biotechnology journal 16 [2018] 288–297. doi:10.1016/j.csbj.2018.08.002.

[90] Z. Peng, C. Xu, H. Wang, J. Huang, J. Xu, X. Chu, P2b-trace: Privacy-preserving blockchain-based contact tracing to combat pandemics, in: Proceedings of the 2021 international conference on management of data, 2021, pp. 2389–2393. doi:10.1145/3448016.3459237.

[91] Y. Chen, L. Meng, H. Zhou, G. Xue, A blockchain-based medical data sharing mechanism with attribute-based access control and privacy protection, Wireless Communications and Mobile Computing 2021 [2021]. doi:10.1155/2021/6685762.

[92] B. Zaabar, O. Cheikhrouhou, F. Jamil, M. Ammi, M. Abid, Healthblock: A secure blockchain-based healthcare data management system, Computer Networks 200 [2021] 108500. doi:10.1016/j.comnet.2021.108500.

[93] R. Zou, X. Lv, J. Zhao, Spchain: blockchain-based medical data sharing and privacy-preserving ehealth system, Information Processing & Management 58 (4) [2021] 102604. doi:10.1016/j.ipm.2021.102604.

[94] K. Azbeg, O. Ouchetto, S. J. Andaloussi, Blockmedcare: A healthcare system based on iot, blockchain and ipfs for data management security, Egyptian Informatics Journal [2022]. doi:10.1016/j.eij.2022.02.004.

[95] T. M. Kim, S.-J. Lee, D.-J. Chang, J. Koo, T. Kim, K.-H. Yoon, I.-Y. Choi, Dynamichain: Development of medical blockchain ecosystem based on dynamic consent system, Applied Sciences 11 (4) [2021] 1612. doi:10.3390/app11041612.

[96] M. Madine, K. Salah, R. Jayaraman, Y. Al-Hammadi, J. Arshad, I. Yaqoob, appxchain: Application-level interoperability for blockchain networks, IEEE Access 9 [2021] 87777–87791. doi:10.1109/access.2021.3089603.

[97] Z. Li, G. Liu, L. Liu, X. Lai, G. Xu, Iot-based tracking and tracing platform for prepackaged food supply chain, Industrial Management & Data Systems [2017]. doi:10.1108/imds-11-2016-0489.

[98] J. F. Galvez, J. C. Mejuto, J. Simal-Gandara, Future challenges on the use of blockchain for food traceability analysis, TrAC Trends in Analytical Chemistry 107 [2018] 222–232. doi:10.1016/j.trac.2018.08.011.

[99] D. Di Francesco Maesa, P. Mori, Blockchain 3.0 applications survey, Journal of Parallel and Distributed Computing 138 [2020] 99–114. doi:10.1016/j.jpdc.2019.12.019.

[100] K. N. Menon, K. Thomas, J. Thomas, D. J. Titus, D. James, Coldblocks: Quality assurance in cold chain networks using blockchain and iot, in: Emerging Technologies in Data Mining and Information Security, Springer, 2021, pp. 781–789. doi:10.1007/978-981-15-9927-9_76.

[101] J. Xie, S. Zhu, B. Li, Research on data storage model of household electrical appliances supply chain traceability system based on blockchain, in: 2021 13th International Conference on Advanced Computational Intelligence (ICACI), 2021, pp. 179–185. doi:10.1109/ICACI52617.2021.9435913.

[102] S. Malik, S. S. Kanhere, R. Jurdak, Productchain: Scalable blockchain framework to support provenance in supply chains, in: 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA), 2018, pp. 1–10. doi:10.1109/NCA.2018.8548322.

[103] Y. P. Tsang, K. L. Choy, C. H. Wu, G. T. S. Ho, H. Y. Lam, Blockchain-driven iot for food traceability with an integrated consensus mechanism, IEEE Access 7 [2019] 129000–129017. doi:10.1109/ACCESS.2019.2940227.

[104] M. Uddin, Blockchain medledger: Hyperledger fabric enabled drug traceability system for counterfeit drugs in pharmaceutical industry, International Journal of Pharmaceutics 597 [2021] 120235. doi:10.1016/j.ijpharm.2021.120235.

[105] T. K. Agrawal, V. Kumar, R. Pal, L. Wang, Y. Chen, Blockchain-based framework for supply chain traceability: A case example of textile and clothing industry, Computers Industrial Engineering 154 [2021] 107130. doi:10.1016/j.cie.2021.107130.

[106] F. Casino, V. Kanakaris, T. K. Dasaklis, S. Moschuris, S. Stachtiaris, M. Pagoni, N. P. Rachaniotis, Blockchain-based food supply chain traceability: a case study in the dairy sector, International Journal of Production Research 59 (19) [2021] 5758–5770. doi:10.1080/00207543.2020.1789238.

[107] G. Ho, Y. M. Tang, K. Y. Tsang, V. Tang, K. Y. Chau, A blockchain-based system to enhance aircraft parts traceability and trackability for inventory management, Expert Systems with Applications 179 [2021] 115101. doi:10.1016/j.eswa.2021.115101.

[108] H. Song, A. Vajdi, Y. Wang, J. Zhou, et al., Blockchain for consortium: A practical paradigm in agricultural supply chain system, Expert Systems with Applications 184 [2021] 115425. doi:10.1016/j.eswa.2021.115101.

[109] E. Rescorla, The transport layer security (TLS) protocol version 1.3, Tech. rep. [2018].

[110] J. A. Berkowsky, T. Hayajneh, Security issues with certificate authorities, in: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 2017, pp. 449–455. doi:10.1109/UEMCON.2017.8249081.

[111] M. Y. Kubilay, M. S. Kiraz, H. A. Mantar, Certledger: A new pki model with certificate transparency based on blockchain, Computers & Security 85 [2019] 333–352. doi:10.1016/j.cose.2019.05.013.

[112] M. Jia, K. He, J. Chen, R. Du, W. Chen, Z. Tian, S. Ji, Process: Privacy-preserving on-chain certificate status service, in: IEEE INFOCOM 2021 - IEEE Conference on Computer Communications, 2021, pp. 1–10. doi:10.1109/INFOCOM42981.2021.9488858.

[113] X. Ge, L. Wang, W. An, X. Zhou, B. Li, Crchain: An efficient certificate revocation scheme based on blockchain, in: International Conference on Algorithms and Architectures for Parallel Processing, Springer, 2021, pp. 453–472. doi:10.6028/NIST.IR.8202.

[114] A. Garba, Z. Chen, Z. Guan, G. Srivastava, Lightledger: A novel blockchain-based domain certificate authentication and validation scheme, IEEE Transactions on Network Science and Engineering 8 (2) [2021] 1698–1710. doi:10.1109/TNSE.2021.3069128.

[115] B. Liu, L. Xiao, J. Long, M. Tang, O. Hosam, Secure digital certificate-based data access control scheme in blockchain, IEEE Access 8 [2020] 91751–91760. doi:10.1109/ACCESS.2020.2993921.

[116] B. M. Nguyen, T.-C. Dao, B.-L. Do, Towards a blockchain-based certificate authentication system in vietnam, PeerJ Computer Science 6 [2020] e266. doi:10.7717/peerj-cs.266.

[117] A. Ali, H. A. Rahim, J. Ali, M. F. Pasha, M. Masud, A. U. Rehman, C. Chen, M. Baz, A novel secure blockchain framework for accessing electronic health records using multiple certificate authority, Applied Sciences 11 (21) [2021] 9999. doi:10.3390/app11219999.
[118] V. S. V. D. P. Lodagala, P. K. Baruah, Sharecert: Sharing and authenticating certificates and credentials on blockchain, in: Blockchain and Deep Learning, Springer, 2022, pp. 3–29. doi:10.1007/978-3-030-95419-2_1.
[119] H. Shen, J. Zhou, Z. Cao, X. Dong, K.-K. R. Choo, Blockchain-based lightweight certificate authority for efficient privacy-preserving location-based service in vehicular social networks, IEEE Internet of Things Journal 7 (7) [2020] 6610–6622. doi:10.1109/JIOT.2020.2974874.
[120] M. Barbieri, D. Gassen, Blockchain-can this new technology really revolutionize the land registry system, in: Responsible Land Governance: Towards an Evidence Based Approach: Proceedings of the Annual World Bank Conference on Land and Poverty, 2017, pp. 1–13.
[121] M. Themistocleous, et al., Blockchain technology and land registry, Cyprus Review 30 (2) [2018] 195–202.
[122] A. S. Yadav, D. S. Kushwaha, Query optimization in a blockchain-based land registry management system., Ingénierie des Systèmes d Inf. 26 (1) [2021] 13–21. doi:10.18280/isi.260102.
[123] A. S. Yadav, D. S. Kushwaha, Blockchain-based digitization of land record through trust value-based consensus algorithm, Peer-to-Peer networking and applications 14 (6) [2021] 3540–3558. doi:10.1007/s12083-021-01207-1.
[124] A. S. Yadav, N. Singh, D. S. Kushwaha, Sidechain: storage land registry data using blockchain improve performance of search records, Cluster Computing [2022] 1–21doi:10.1007/s10586-022-03535-0.
[125] P. D. Ameyaw, W. T. de Vries, Toward smart land management: Land acquisition and the associated challenges in ghana. a look into a blockchain digital land registry for prospects, Land 10 (3) [2021] 239. doi:10.3390/land10030239.
[126] S. Soner, R. Litoriya, P. Pandey, Exploring blockchain and smart contract technology for reliable and secure land registration and record management, Wireless Personal Communications 121 (4) [2021] 2495–2509. doi:10.1007/s11277-021-08833-1.
[127] M. Nandi, R. K. Bhattacharjee, A. Jha, F. A. Barbhuiya, A secured land registration framework on blockchain, in: 2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP), IEEE, 2020, pp. 130–138. doi:10.1109/isea-isap49340.2020.235011.
[128] K. Veeramani, S. Jaganathan, Land registration: Use-case of e-governance using blockchain technology, KSII Transactions on Internet and Information Systems (TIIS) 14 (9) [2020] 3693–3711. doi:10.5195/ledger.2016.62.
[129] A. F. Mendi, K. K. Sakaklı, A. Çabuk, A blockchain based land registration system proposal for turkey, in: 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), IEEE, 2020, pp. 1–6. doi:10.1109/ismsit50672.2020.9255078.
[130] V. Thakur, M. Doja, Y. K. Dwivedi, T. Ahmad, G. Khadanga, Land records on blockchain for implementation of land titling in india, International Journal of Information Management 52 [2020] 101940. doi:10.1016/j.ijinfomgt.2019.04.013.
[131] M. Biswas, J. Al Faysal, K. A. Ahmed, Landchain: A blockchain based secured land registration system, in: 2021 International Conference on Science & Contemporary Technologies (ICSCT), IEEE, 2021, pp. 1–6. doi:10.1109/ICSCT53883.2021.9642505.
[132] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, B. C. Ooi, Towards scaling blockchain systems via sharding, in: Proceedings of the 2019 international conference on management of data, 2019, pp. 123–140. doi:10.1145/3299869.3319889.