Modern technologies of decentralized databases, authentication, and authorization methods

Petro Petriv[1,†], Ivan Opirskyy[1,*,†] and Nataliia Mazur[2,†]

[1] Lviv Polytechnic National University, 79013 Lviv, Ukraine
[2] Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudryavska str., 04053 Kyiv, Ukraine

Abstract

With the development of decentralized technologies and the increasing volume of data generated and processed, there is a challenge to ensure effective and secure information management, especially in the context of distributed systems. Traditional centralized databases increasingly demonstrate limitations in terms of scalability and fault tolerance. The paper proposes a comprehensive analysis of modern blockchain-based decentralized database technologies and examines the authentication and authorization methods used in them. The study covers seven leading systems: BigchainDB, GUN, OrbitDB, Bluzelle, Fluree, and Ties.DB, and Hyperledger Fabric. The problem statement includes current challenges in the field of decentralized data storage, such as ensuring a high level of security, scalability, and compliance with regulatory requirements. An important component of the paper is the analysis of recent research and publications, focused on the development of consensus algorithms, improvement of cryptographic methods, and integration of smart contracts into decentralized databases. Each system is examined in terms of its architecture, consensus mechanisms, and approaches to data management. The main objective of the study is to systematize and comparatively analyze existing decentralized database technologies, assess their efficiency and security, and identify promising directions for further development. Special attention is paid to security methods, particularly the use of public key cryptography, smart contracts, and distributed access control.

Keywords

data protection, blockchain, government registries, transparency, data security, confidentiality, smart contracts, audit, personal data, mathematical model, trust

1. Introduction

The development of information technologies over the past decades has led to exponential growth in the volume of data generated, stored, and processed. Traditional centralized database management systems, which have long dominated the industry, are increasingly facing limitations in terms of scalability, security, and fault tolerance. In this context, decentralized databases (DDBs) based on blockchain technology have emerged as a promising solution that promises to overcome these limitations [1].

The concept of decentralized systems is not new. It dates back to the early days of computer networks and distributed systems development. However, the emergence of blockchain technology in 2008, presented in the work of Satoshi Nakamoto [2], gave impetus to the development of a new generation of decentralized data storage and processing systems. As Zheng et al. (2017) [3] point out, blockchain offers an innovative approach to ensuring data integrity and immutability in a distributed environment without the need for a trusted third party.

Blockchain-based decentralized databases offer several unique advantages compared to traditional systems. They provide enhanced security through cryptographic methods of data protection, transparency of operations through public access to transaction history, and resistance to censorship due to the distributed nature of the system [4]. In their comprehensive study, Dinh et al. (2018) [4] further analyze these systems from a data processing perspective, highlighting the unique challenges and opportunities that arise when implementing blockchain technology in database management. These characteristics make DDBs particularly attractive for a wide range of applications, from financial systems and electronic voting to supply chain management and medical data storage.

However, along with the advantages, decentralized databases also bring new challenges, especially in the area of user authentication and authorization. Traditional access control methods developed for centralized systems [4] often prove ineffective or impractical in the context of DDBs. The absence of a central governing body requires new approaches to user identification, data access management, and ensuring information confidentiality.

The importance of reliable authentication and authorization methods in decentralized systems cannot be overstated. They are fundamental to ensuring data security, access control, and maintaining user trust in the system. In

CPITS-II 2024: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, October 26, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
petro.p.petriv@lpnu.ua (P. Petriv);
iopirsky@gmail.com (I. Opirskyy);
n.mazur@kubg.edu.ua (N. Mazur)

0009-0000-7426-3696 (P. Petriv);
0000-0002-8461-8996 (I. Opirskyy);
0000-0001-7671-8287 (N. Mazur)

an environment where cyber attacks are becoming increasingly sophisticated and regulatory requirements for data protection are becoming more stringent (for example, GDPR in Europe) [5], the development of effective authentication and authorization mechanisms becomes a critical task for the widespread adoption of decentralized databases.

In recent years, several innovative approaches to solving these problems have emerged. They range from the use of complex cryptographic protocols and smart contracts to the implementation of decentralized identity management systems (DID) [6]. Each of these approaches has its advantages and limitations, and the choice of a specific solution often depends on the specific requirements of the particular application.

Problem formulation. Despite significant progress in the development of decentralized databases, several unresolved issues remain, especially in the context of authentication and authorization. The key challenges are:

- Ensuring a high level of security without excessively complicating the user experience.
- Developing scalable solutions capable of handling a large number of users and transactions.
- Addressing data privacy issues in the context of the transparent nature of blockchain systems.
- Ensuring compliance with regulatory requirements, especially in the field of personal data protection.
- Integration with existing systems and infrastructures.

These issues create an urgent need for a comprehensive analysis of existing decentralized database technologies and the authentication/authorization methods used in them.

Recent research and publications analysis. Research in the field of decentralized databases and authentication/authorization methods is actively developing. Dinh et al. conducted a comprehensive review of blockchain database systems [4], analyzing their architectures and consensus mechanisms. This work laid the foundation for understanding the basic principles of DDB functioning.

Wang et al. focused on the issues of scalability and performance of DDBs [4], proposing new algorithms for optimizing transaction processing. Their research emphasizes the importance of efficient data processing in distributed systems.

In the area of security and privacy, Zhang et al. proposed an innovative approach to ensuring data confidentiality [7] in blockchain systems using homomorphic encryption. This work opens up new possibilities for protecting sensitive data in a decentralized environment.

Li et al. developed a new smart contract-based identity management method [4] for blockchain systems, demonstrating the potential for integrating complex authorization logic directly into the blockchain.

Xu et al. proposed a distributed authentication scheme for the Internet of Things (IoT) based on blockchain, highlighting the importance of adapting authentication methods to the specific needs of different application domains.

Yevseiev et al. [8] presented a comprehensive analysis of security models for socio-cyber-physical systems, which is particularly relevant in the context of developing decentralized databases and their integration with IoT and other modern technologies. Balatska et al. [9] explored the concept of applying blockchain in the context of Single Sign-On (SSO) technology, opening new perspectives for improving the security and convenience of authentication in decentralized systems. Poberezhnyk et al. [10] proposed a concept for a learning management system based on blockchain technology, demonstrating the potential of decentralized databases in the educational sphere.

The purpose of the paper. The purpose of this paper is to conduct a comprehensive analysis of modern decentralized database technologies and the authentication/authorization methods used in them. The research is focused on:

- Systematization and comparative analysis of architectures and functionalities of leading DDB systems, such as BigchainDB, GUN, OrbitDB, Bluzelle, Fluree, and Ties.DB, and Hyperledger Fabric.
- Evaluation of the effectiveness and security of various authentication and authorization methods in a decentralized environment.
- Identification of key problems and limitations of existing approaches to ensuring security in DDBs.
- Determination of promising directions for further research and development in the field of DDB security.

The results of this study aim to provide developers, researchers, and organizations with valuable information for decision-making regarding the selection and implementation of decentralized data management systems, as well as to outline ways to improve security methods in these systems.

This work is particularly relevant in the context of growing interest in decentralized technologies across various sectors, from finance and healthcare to public administration and the Internet of Things. Understanding the strengths and weaknesses of different approaches to authentication and authorization in decentralized systems is critical for developing secure, efficient, and scalable solutions capable of meeting the needs of the modern digital world.

2. Overview of decentralized database technologies

Decentralized databases (DDBs) represent a new generation of data storage and processing systems that combine the principles of distributed systems with blockchain technology. Unlike traditional centralized databases, DDBs distribute data across multiple nodes, ensuring high fault tolerance, transparency, and protection against unauthorized changes.

In this section, we will conduct a detailed analysis of seven leading decentralized database technologies: BigchainDB, GUN, OrbitDB, Bluzelle, Fluree, and Ties.DB, and Hyperledger Fabric. Each of these systems offers a

unique approach to solving key problems of decentralized data storage, in particular:

1. Architecture and data model: We will examine how each system structures and organizes data, including the use of blockchain, graph models, or other approaches.
2. Consensus mechanisms: We will analyze the methods used to achieve agreement between network nodes regarding the state of data.
3. Scalability and performance: We will assess each system's ability to handle large volumes of data and transactions.
4. Identification and authorization methods: Special attention will be paid to mechanisms that ensure secure user identification and control of data access. This includes:

- Cryptographic methods used for identity creation and verification.
- Key and certificate management systems.
- Access control mechanisms at the data and transaction levels.
- Implementation of smart contracts for automating access rules.

5. Integration and compatibility: We will consider how easily each system can be integrated with existing technologies and standards.
6. Privacy and confidentiality: We will analyze the methods used to protect sensitive data in a distributed environment.

This comprehensive review will allow us not only to understand the technical features of each system but also to assess their suitability for various use cases, from financial applications to supply chain management systems and the IoT.

Furthermore, we will pay attention to the challenges and limitations faced by each technology, which will help identify directions for further research and development in the field of decentralized databases.

2.1. BigchainDB

BigchainDB is a decentralized database that combines the properties of traditional databases with blockchain characteristics, providing high throughput and low latency [11].

Architecture and data model. BigchainDB uses a transaction-based data model, where each transaction contains metadata, digital assets, and ownership transfer information. The system organizes data into "blocks" that are linked in a chain, forming a blockchain. This hybrid architecture allows BigchainDB to retain the advantages of both traditional databases and blockchain systems.

Consensus mechanism. BigchainDB uses the Tendermint consensus algorithm [11], which ensures rapid agreement between network nodes. This mechanism allows the system to achieve transaction finality within seconds, significantly faster than traditional blockchain systems. Tendermint also provides resistance to Byzantine failures, enhancing system reliability.

Scalability and performance. Performance evaluation of BigchainDB showed that the system is capable of processing thousands of transactions per second, bringing it close to the performance of traditional databases. Scalability is achieved through horizontal scaling of network nodes. However, as the number of nodes increases, the complexity of achieving consensus may grow.

According to research by McConaghy et al. (2016), BigchainDB demonstrates the ability to process up to 1 million records per second using a cluster of 32 nodes. This significantly exceeds the performance of traditional blockchain systems such as Bitcoin (7 transactions per second) or Ethereum (15 transactions per second).

Identification and authorization method. BigchainDB uses public key cryptography for user identification. Each user has a pair of keys: public (for identification) and private (for signing transactions). Authorization is based on the concept of "Proof of Asset Ownership". Transactions are signed with the owner's private key, ensuring action authorization. This approach provides a high level of security but may create challenges in managing a large number of keys in corporate environments.

Integration and compatibility. BigchainDB provides an API for integration with other systems, facilitating its implementation into existing infrastructures. However, full compatibility with traditional SQL databases is limited due to its specific data model.

Privacy and confidentiality. BigchainDB ensures transaction transparency, which can be an advantage for some use cases but creates challenges for maintaining the confidentiality of sensitive data. The system offers limited built-in data encryption mechanisms at the transaction level.

In summary, BigchainDB offers a unique combination of high performance of traditional databases with the security and immutability of blockchain. However, the balance between transparency and confidentiality remains a challenge for widespread implementation in scenarios requiring a high level of data privacy.

2.2. GUN

GUN is an open-source decentralized graph database that provides real-time data replication and supports an offline-first architecture. According to Nadal (2018) [12], the creator of GUN, this system was designed to be a decentralized alternative to traditional databases, offering features such as real-time synchronization, offline-first capabilities, and graph-based data modeling.

Architecture and data model. GUN uses a graph data model where each node can have connections with other nodes. This model provides flexibility in representing complex relationships between data. GUN's architecture is based on the peer-to-peer principle, where each node can act as both client and server simultaneously. This allows the system to operate even with partial network connection loss.

Consensus mechanism. GUN uses a Conflict-free Replicated Data Type (CRDT) mechanism [12] to achieve consensus. This approach allows the system to effectively

resolve conflicts during simultaneous data updates by different nodes, ensuring eventual consistency. CRDT enables GUN to maintain high data availability even under unstable network conditions.

Scalability and performance. Performance evaluation of GUN has shown that the system is capable of processing a large number of read and write operations in real time. Scalability is achieved through a decentralized architecture where each node can independently process requests. However, as the number of connections between data increases, there may be delays in processing complex queries.

Identification and authorization method. GUN uses a key pair-based identification system known as SEA (Security, Encryption, Authorization). It supports decentralized authentication without the need for a centralized server. Users create and manage their keys locally. The concept of a "trust graph" is implemented for access management between nodes. This approach provides a high level of privacy and control for users but may create difficulties in implementing centralized security policies in corporate environments.

Integration and compatibility. GUN provides an API for JavaScript, which facilitates integration with web applications and Node.js projects. However, support for other programming languages is limited, which may complicate integration into some existing systems.

Privacy and confidentiality. GUN ensures a high level of privacy through local key storage and the ability to encrypt data on the client side. However, full decentralization may create challenges for implementing complex access control and audit schemes in corporate environments.

GUN stands out for its ability to provide high data availability and offline operation, making it attractive for distributed and mobile applications. However, limited support for programming languages and the complexity of implementing centralized security policies may limit its application in some corporate scenarios.

2.3. OrbitDB

OrbitDB is a distributed database built on the InterPlanetary File System (IPFS), providing decentralized data storage and synchronization. Haad and Nævdal (2019) [13], the creators of OrbitDB, describe it as a peer-to-peer database specifically designed for the decentralized web. They emphasize its ability to operate without centralized servers, making it particularly suitable for decentralized applications (dApps) and distributed systems that require robust data management capabilities.

Architecture and data model. OrbitDB uses IPFS for data storage, ensuring high scalability and resistance to censorship. The system supports various types of data stores, including key-value stores, event logs, and document databases. This flexible architecture allows OrbitDB to adapt to diverse usage scenarios.

Consensus mechanism. OrbitDB uses a Conflict-free Replicated Data Type (CRDT) based consensus mechanism [13], which effectively resolves conflicts during simultaneous data updates by different nodes. This approach ensures eventual data consistency without the need for complex consensus algorithms.

Scalability and performance. Evaluation has shown that OrbitDB can scale effectively thanks to its use of IPFS. However, performance may vary depending on the size of the IPFS network and the type of operations. The system is particularly effective for applications requiring high data availability and resilience to network failures.

Identification and authorization method. OrbitDB uses IPFS identifiers for unique user identification. The system supports distributed access control, where each database has its own set of access rights. Elliptic curve cryptography-based signatures are used to verify user actions. This approach provides flexible access control but may complicate management in large organizations.

Integration and compatibility. OrbitDB provides a JavaScript API, facilitating integration with web applications. However, support for other programming languages is limited, which may create challenges when integrating with diverse systems.

Privacy and confidentiality. OrbitDB provides a basic level of privacy through access control but lacks built-in data encryption mechanisms. This may require additional measures to ensure the confidentiality of sensitive information.

OrbitDB stands out for its integration with IPFS, making it attractive for decentralized web applications. However, limited built-in encryption mechanisms and dependence on the JavaScript ecosystem may restrict its application in some scenarios.

2.4. Bluzelle

Bluzelle is a decentralized database that uses a 'swarm' model for data storage and management, providing high scalability and reliability. According to the Bluzelle Networks whitepaper (2017) [14], Bluzelle was specifically designed as a decentralized database service for decentralized applications (dApps). The whitepaper emphasizes Bluzelle's unique 'swarm' architecture, which enables the network to dynamically scale and self-heal, providing robust data storage solutions for blockchain-based applications and other decentralized systems.

Architecture and data model. Bluzelle uses a distributed architecture where data is distributed among many nodes in a 'swarm'. This ensures high availability and fault tolerance. The system implements a NoSQL data model, allowing flexible storage and retrieval of data with various structures.

Consensus mechanism. The system uses its consensus algorithm based on the concept of 'Proof of Stake' [14], which enables rapid agreement between nodes. This mechanism allows Bluzelle to achieve high throughput while maintaining the decentralized nature of the system.

Scalability and performance. Evaluation has shown that Bluzelle's architecture allows for efficient scaling, and processing of a large number of parallel queries. The system uses dynamic sharding for load distribution, which maintains high performance as data volume increases.

Identification and authorization method. Bluzelle uses cryptographic tokens for access control and employs smart contracts to manage access rights. The system supports multi-level authorization for different types of operations. This approach provides flexible access control but may

require additional effort to integrate with existing identification systems.

Integration and compatibility. Bluzelle provides APIs for various programming languages, facilitating integration with different types of applications. The system also supports standard data exchange protocols, simplifying interaction with existing infrastructures.

Privacy and confidentiality. Bluzelle offers basic data encryption mechanisms, but full confidentiality can be challenging in a distributed environment. The system allows for configuring privacy levels for different types of data.

Bluzelle stands out for its ability to provide high scalability and reliability thanks to its 'swarm' architecture. However, implementing complex access control schemes and ensuring full data confidentiality may require additional efforts when deploying in corporate environments.

2.5. Fluree

Fluree is a semantic graph database on blockchain that supports smart contracts and provides high query performance. Platz and Hilger (2019) [15], the creators of Fluree, describe it as a practical decentralized database that combines the benefits of blockchain technology with the flexibility of semantic graph databases. They emphasize Fluree's unique approach to data management, which includes time-travel queries, blockchain-grade security, and the ability to run complex analytical queries directly on blockchain data. This design, according to the authors, makes Fluree particularly suitable for enterprise applications that require both the immutability of blockchain and the advanced querying capabilities of traditional databases.

Architecture and data model. Fluree uses a semantic graph data model, allowing the creation of complex relationships between data. The system integrates blockchain to ensure the immutability and transparency of transactions. This hybrid architecture enables Fluree to combine the advantages of graph databases and blockchain.

Consensus mechanism. Fluree uses its consensus mechanism [15], which combines elements of Proof of Stake and Byzantine fault tolerance. This allows the system to achieve rapid consensus while maintaining a high level of security and decentralization.

Scalability and performance. Evaluation has shown that Fluree provides high query performance thanks to its optimized graph data structure. Scalability is achieved through the ability to create private subnets. The system also supports parallel query processing, which increases overall performance.

Identification and authorization method. Fluree uses digital signatures based on elliptic curve cryptography for identification. The system supports complex authorization rules at the data level through smart functions, allowing access rules to be defined at the level of individual predicates. This provides high flexibility in configuring access rights but may require careful planning during implementation.

Integration and compatibility. Fluree provides a RESTful API and GraphQL interface, facilitating integration with various types of applications. The system also supports standard data formats, simplifying information exchange with other systems.

Privacy and confidentiality. Fluree offers flexible access control mechanisms, but full data confidentiality can be challenging due to the transparency of the blockchain. The system allows configuring different levels of data visibility for different users.

Fluree stands out for its ability to combine semantic queries with blockchain security, making it attractive for applications that require complex data processing and high levels of auditing. However, balancing blockchain transparency with confidentiality requirements can be challenging in some use cases.

2.6. Ties.DB

Ties.DB is an open-source decentralized SQL-like database that provides flexibility in querying and data indexing. According to the Ties.Network whitepaper (2017) [16], Ties.DB was designed as a distributed database solution that combines the familiarity of SQL with the benefits of decentralization. The whitepaper emphasizes Ties.DB's unique approach to decentralized data management, includes support for complex SQL-like queries, a tokenized economic model for incentivizing network participants, and a flexible architecture that allows for custom implementation of consensus mechanisms. These features, as described by Ties.Network, make Ties.DB particularly suitable for decentralized applications that require sophisticated data querying capabilities while maintaining the benefits of blockchain-based data integrity and distribution.

Architecture and data model. Ties.DB uses a distributed architecture with support for SQL-like queries. The system provides a relational data model in a decentralized environment. This architecture allows combining a familiar SQL interface with the advantages of decentralized systems.

Consensus mechanism. Ties.DB uses a Proof of Stake-based consensus mechanism [16] for validating transactions and data changes. This ensures efficient agreement between network nodes while maintaining the decentralized nature of the system.

Scalability and performance. Evaluation has shown that Ties.DB provides good scalability thanks to its distributed architecture. The system is optimized for fast execution of complex queries. The use of indexing and caching allows maintaining high performance when working with large volumes of data.

Identification and authorization method. Ties.DB uses cryptographic keys for user identification. The system supports a tokenized model for access management and service payments. Data owners can set flexible access rules for their tables and records. This approach provides high flexibility but may require additional efforts to integrate with existing identification systems.

Integration and compatibility. Ties.DB provides an SQL-like interface, facilitating integration with existing systems and applications. This allows developers to use familiar tools and methods for working with data in a decentralized environment.

Privacy and confidentiality. Ties.DB offers basic mechanisms for ensuring data privacy, but full confidentiality can be challenging in a decentralized environment. The system allows configuring access rights at the level of individual tables and records.

Ties.DB stands out for its ability to provide an SQL-like interface in a decentralized environment, making it attractive to organizations looking to transition to decentralized systems while maintaining familiar data-handling tools. However, ensuring full confidentiality and compliance with regulatory requirements may require additional measures.

2.7. Hyperledger Fabric

Hyperledger Fabric is a platform for creating private blockchain networks with the ability to store data and execute smart contracts, designed for enterprise use [17].

Architecture and data model. Hyperledger Fabric uses a modular architecture that allows customization of various system components. The platform supports different data models through the concept of 'world state'. This flexible architecture allows adapting the system to diverse business requirements.

Consensus mechanism. Hyperledger Fabric offers a flexible approach to consensus [17], allowing the selection of different algorithms depending on the needs of a specific network. This can include algorithms based on Practical Byzantine Fault Tolerance (PBFT) or Raft. Such flexibility allows for optimizing network performance and security according to specific requirements.

Scalability and performance. Evaluation has shown that Hyperledger Fabric provides high-performance thanks to an architecture that separates tasks between different types of nodes. Scalability is achieved through the ability to create separate channels for different groups of participants. The system also supports parallel execution of transactions, which increases overall throughput.

Research by Androulaki et al. (2018) showed that Hyperledger Fabric can achieve a throughput of over 3500 transactions per second with a latency of less than a second in a network of 100 nodes. The system demonstrates linear scaling as the number of nodes increases.

Identification and authorization method. Hyperledger Fabric uses X.509 certificates to identify network participants. The system supports a role-based membership model (Membership Service Provider, MSP) and allows configuring complex authorization rules through the endorsement policies system. This approach provides a high level of control and flexibility in access management, which is especially important for enterprise applications.

Integration and compatibility. Hyperledger Fabric provides SDKs for various programming languages, facilitating integration with enterprise systems. The platform also supports standard data exchange protocols and can be integrated with existing identity and access management systems. This makes Fabric particularly attractive to organizations looking to implement blockchain technologies into their existing IT infrastructure.

Privacy and confidentiality. Hyperledger Fabric offers advanced privacy features, including private channels and private data. This allows the creation of subnets with limited access and the storing of sensitive information visible only to authorized participants. Additionally, the platform supports the use of Zero-Knowledge Proofs for additional privacy protection.

Hyperledger Fabric stands out for its focus on enterprise needs, offering a high level of customization, performance, and privacy. The platform is particularly suitable for creating consortium blockchains where control over network participants and their rights is required. However, the complexity of setting up and managing such a system may require significant resources and expertise.

Overall, Hyperledger Fabric offers a powerful solution for organizations seeking ways to implement blockchain technologies while meeting corporate requirements for security, performance, and confidentiality. Its modular architecture and flexibility in configuration allow adapting the platform to a wide range of uses, from supply chain management to financial services and healthcare.

3. Comparative analysis of authentication and authorization methods

The analysis of seven leading decentralized database technologies revealed significant differences in approaches to authentication and authorization. These differences reflect the diversity of requirements and use cases for which these systems were developed.

3.1. Cryptographic methods

All the systems examined are based on public key cryptography but implement it differently. BigchainDB and Hyperledger Fabric use traditional approaches with digital signatures, providing a high level of security and compatibility with existing standards. In contrast, GUN and OrbitDB introduce innovative approaches such as SEA (Security, Encryption, Authorization) and IPFS identifiers respectively, allowing them to better adapt to the specific requirements of decentralized systems.

Particular attention should be paid to Fluree's approach, which uses smart functions to implement complex authorization rules at the data level. This gives the system unique flexibility in configuring access rights but may complicate the security management process for less experienced users.

The analysis shows that the choice of cryptographic method significantly affects the balance between security, flexibility, and ease of use of the system. Systems with more traditional approaches tend to integrate more easily with existing infrastructures, while innovative solutions offer new possibilities but may require additional staff training.

3.2. Key management

Key management approaches differ significantly between systems, reflecting various philosophies regarding the balance between security and usability. BigchainDB and Ties.DB places the responsibility for key management on users, which enhances security but can be challenging for ordinary users. This approach may be optimal for systems where users have a high level of technical literacy.

GUN offers decentralized key management, which improves privacy but may complicate access recovery. This solution is particularly interesting for applications where user privacy is a top priority.

Hyperledger Fabric uses centralized certification services (CA), which facilitates management in corporate environments but creates a single point of failure. This approach reflects Fabric's orientation towards enterprise applications, where decentralized identity management is the norm.

The analysis shows that the choice of key management approach should take into account the specifics of the target audience and use cases. Systems aimed at mass users may require simpler solutions, while enterprise applications may prefer more controlled approaches.

3.3. Granularity of access control

The level of access control granularity varies from system to system, affecting their suitability for different use cases. Fluree and Hyperledger Fabric offer the most flexible mechanisms, allowing access rules to be defined at the level of individual data fields. This makes them particularly attractive for scenarios requiring fine-grained control over data access, such as in the financial sector or healthcare.

BigchainDB and Bluzelle provide access control at the transaction and asset level, which may be sufficient for many business applications but less flexible compared to the approach of Fluree and Fabric.

GUN and OrbitDB have more limited capabilities, focusing on access to nodes or databases as a whole. This may be acceptable for simple applications or systems where speed and simplicity are priorities, but it may limit their use in complex corporate environments.

The analysis shows that choosing a system with an appropriate level of access control granularity is critical to balancing security and data management efficiency. Systems with more detailed access control typically require more resources for setup and management but provide more opportunities for regulatory compliance and protection of sensitive data.

3.4. Integration with existing authentication systems

The integration of decentralized databases with existing authentication systems is a critical aspect of their implementation in organizational structures. Analysis of the technologies examined revealed significant differences in their integration capabilities, which substantially affect their suitability for various environments.

For effective integration, an identity and data transformation model is proposed, which ensures a smooth transition from traditional systems to decentralized solutions. This model includes stages of input data normalization, generation and validation of decentralized identifiers (DIDs), processing in a distributed ledger, and generation of output tokens for existing systems.

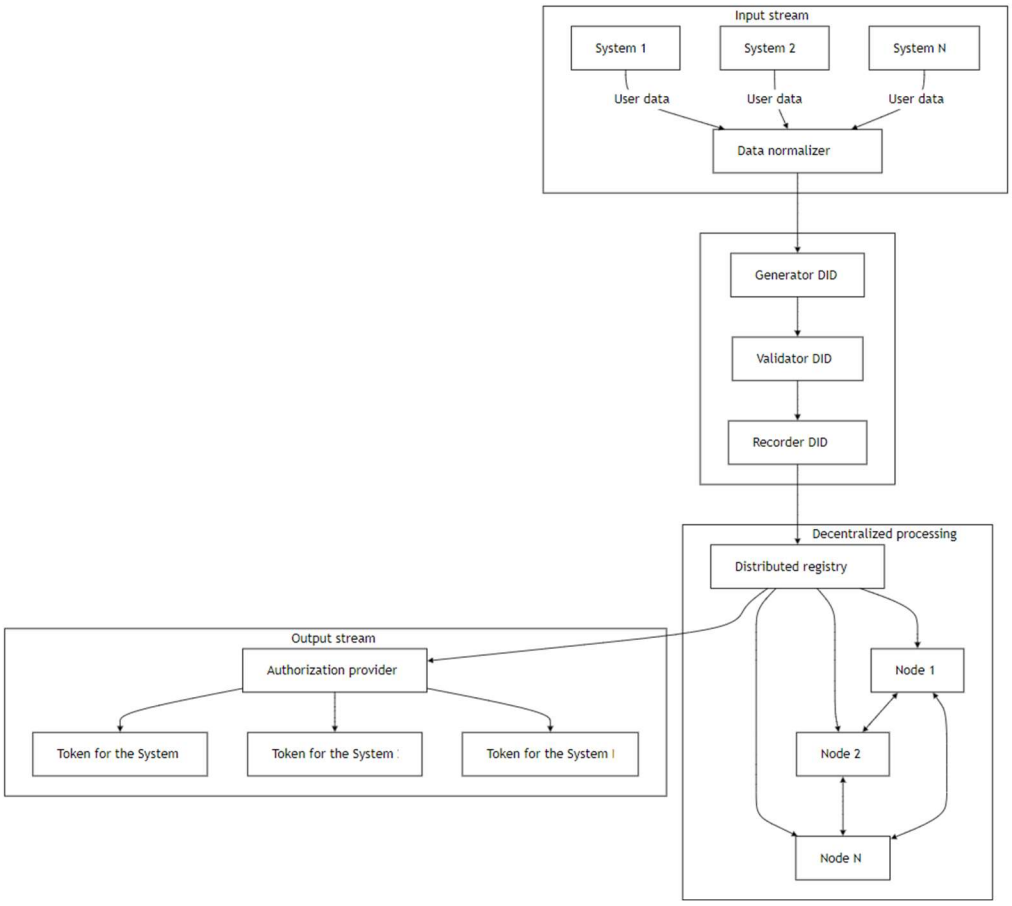

Figure 1: A model of identity and data transformation in decentralized databases

Hyperledger Fabric demonstrates the highest level of integration capabilities due to its support for standard protocols such as LDAP, OAuth 2.0, and Active Directory. This allows effective interaction with existing corporate identity management systems, simplifying the process of input data normalization and identity transformation.

BigchainDB and Ties.DB, offering APIs for integration, occupies an intermediate position. While they provide some flexibility, additional development may be needed to achieve full compatibility. In the context of the proposed model, this means creating specialized adapters for efficient data processing and DID generation.

GUN and OrbitDB have the most limited integration capabilities, creating significant challenges when implementing them in existing infrastructures. These systems require the development of complex gateways or intermediate services, which can negatively affect overall efficiency and complicate scaling.

Bluzelle and Fluree occupy an intermediate position, offering a certain level of integration through APIs and support for external services. This allows adapting them to the proposed model with moderate effort.

The effectiveness of integration significantly affects the overall performance and scalability of the system. Using the proposed mathematical model, integration efficiency (E) can be expressed as a function of throughput (T), DID validation speed (V), level of consensus between nodes (C), data transformation delay (D), and network load (L):

$$E = \frac{T \cdot V \cdot C}{D \cdot L}.$$

Additionally, the scalability coefficient (S) can be represented as:

$$S = \frac{N \cdot P}{I \cdot R}$$

where N is the number of nodes in the network, P is the query processing performance per node, I is the complexity of integrating a new node, and R is the resource requirements per node.

Systems with better integration capabilities, such as Hyperledger Fabric, allow achieving higher E and S indicators by reducing parameters D and I.

Thus, choosing a system with appropriate integration capabilities is a critical factor for the successful implementation of decentralized databases. Systems with developed integration capabilities provide a smoother transition and reduce risks, especially in the context of large organizations with complex existing infrastructures.

3.5. Support for anonymity and pseudonymity

Approaches to ensuring anonymity and pseudonymity differ significantly among the systems examined, reflecting different priorities regarding privacy and transparency.

GUN and OrbitDB provide a high level of anonymity due to their decentralized nature and the use of pseudonyms. This makes them attractive for applications where user privacy is a top priority, such as in social networks or voting systems.

BigchainDB and Bluzelle allow pseudonymous use but store all transactions, which may allow behavior analysis. This approach provides a balance between privacy and auditability, which can be useful for financial applications or supply chain management systems.

Hyperledger Fabric, oriented towards enterprise use, has limited possibilities for anonymity but offers private channel features for confidentiality. This reflects the priority of regulatory compliance and the need for auditing in corporate environments.

The analysis shows that the choice of a system with an appropriate level of anonymity and pseudonymity support depends on the specific requirements for privacy and transparency within a particular application. Systems with a high level of anonymity may be better for applications focused on protecting user privacy, while systems with greater transparency may be more suitable for corporate and regulated environments.

3.6. General conclusions of the comparative analysis

The comparative analysis of authentication and authorization methods in the examined decentralized databases reveals a significant diversity of approaches, each with its advantages and limitations.

Systems oriented towards enterprise use, such as Hyperledger Fabric, offer more traditional and integrated approaches to authentication and authorization, facilitating their implementation into existing business processes. However, these systems may be less flexible in the context of decentralization and anonymity.

On the other hand, systems like GUN and OrbitDB offer a high level of decentralization and anonymity but may create challenges when integrating with traditional corporate systems.

BigchainDB, Bluzelle, Fluree, and Ties.DB occupy intermediate positions, offering various combinations of features that allow them to adapt to different usage scenarios.

The choice of an optimal system depends on the specific requirements of the project, including the necessary level of security, privacy, scalability, and integration with existing systems. Organizations should carefully evaluate their needs and constraints before choosing a specific decentralized database technology.

Authentication and authorization in decentralized systems present a particular challenge due to the absence of a central governing body. Traditional methods that rely on centralized authentication servers cannot be directly applied in such an environment. Instead, decentralized databases must develop innovative approaches that ensure reliable user identification and access control while maintaining the advantages of a distributed architecture.

These tables demonstrate the diversity of approaches to authentication and authorization in decentralized databases, highlighting the strengths and limitations of each system.

3.7. Comparative tables

Table 1
Comparison of authentication methods

System	Authentication method	Key management	Anonymity support
BigchainDB	Public key cryptography	User-managed	Medium
GUN	SEA (Security, Encryption, Authorization)	Decentralized	High
OrbitDB	IPFS identifiers	Decentralized	High
Bluzelle	Cryptographic tokens	User-managed	Medium
Fluree	Digital signatures + smart functions	User-managed	Low
Ties.DB	Cryptographic keys	User-managed	Medium
Hyperledger Fabric	X.509 certificates	Centralized (CA)	Low

Table 2
Comparison of authorization methods

System	Control Granularity	Authorization Mechanism	Integration with Existing Systems
BigchainDB	Transaction level	Proof of Asset Ownership	Medium
GUN	Node level	Trust graph	Low
OrbitDB	Database level	Distributed access control	Low
Bluzelle	Transaction level	Smart contracts	Medium
Fluree	Predicate level	Smart functions	High
Ties.DB	Table/record level	Tokenized system	Medium
Hyperledger Fabric	Channel/chain code level	Endorsement policies	High

4. Advancing decentralized database technologies

Current research in the field of decentralized databases (DDBs) reveals several key areas that require further improvement and development. Analysis of these areas not only outlines the current limitations of the technology but also identifies promising ways to overcome them.

One of the most critical aspects of DDB development is improving their scalability. Research by Bano et al. (2019) [18] demonstrates that existing consensus algorithms, particularly Proof of Work, have significant limitations in terms of throughput as the number of nodes in the network increases. This leads to a decrease in transaction processing speed and an increase in latency, which is especially critical for applications in the financial sector and real-time systems.

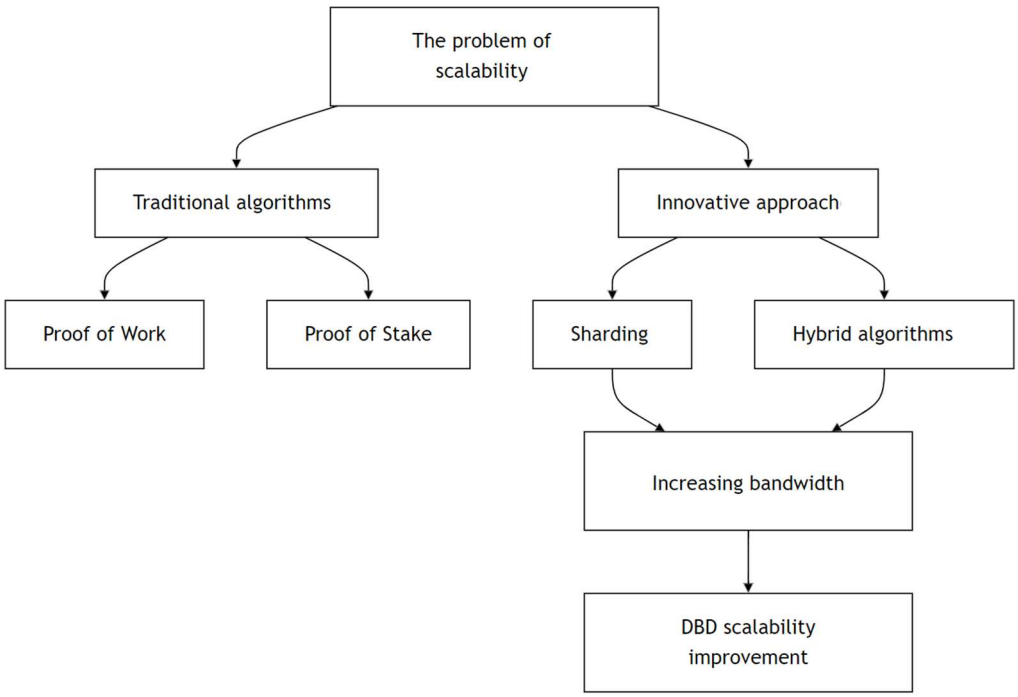

Figure 2: Approaches to solving the problem of scalability in decentralized databases

To address the scalability problem, new approaches are being developed, among which the concept of sharding is particularly noteworthy. Zamani et al. (2018) [19] propose a method of dividing the network into subnets for parallel transaction processing, which significantly increases the system's throughput without compromising security. This approach opens up possibilities for creating high-performance DDBs capable of competing with centralized systems in terms of transaction processing speed.

Another important aspect of DDB development is improving data storage and processing methods. Sharma et al. (2019) [20] point to the problem of significant database size increase when using traditional approaches to data storage in blockchain. This complicates maintenance and synchronization between nodes, especially for full nodes that store the entire transaction history. This can result in a decrease in the network's decentralization level due to a reduction in the number of participants capable of maintaining full nodes.

Ensuring data confidentiality in a distributed environment remains one of the key challenges for DDBs. Reid and Harrigan (2013) [21] demonstrated the possibility of analyzing links between transactions even in systems considered anonymous, which can lead to user de-anonymization. This problem is particularly relevant for applications requiring a high level of privacy, such as in healthcare or financial services.

A promising direction for solving the confidentiality problem is the application of Zero-Knowledge Proofs (ZKP) technology. Kosba et al. (2016) [7] demonstrate the potential of this technology for creating private smart contracts, allowing transaction verification without disclosing their content. This opens up new possibilities for ensuring privacy in decentralized systems while maintaining their main advantages.

An important aspect of DDB development is also ensuring compliance with regulatory requirements, particularly the General Data Protection Regulation (GDPR) in the European Union. Finck (2019) [22] analyzes the potential conflict between the right to be forgotten provided by GDPR and the immutability of data in blockchain. This problem requires the development of innovative technical solutions that will allow modifying or deleting data without compromising blockchain integrity.

Given the development of quantum computing, the development and implementation of quantum-resistant cryptography algorithms become particularly relevant. Bernstein and Lange (2017) [23] propose some post-quantum cryptographic primitives that can ensure DDB security even in the era of quantum computers. This is critical for ensuring the long-term viability and reliability of decentralized systems.

The development of quantum-resistant cryptography is crucial for the long-term security of decentralized databases. Horpenyuk et al. [24] argue that the implementation of post-quantum cryptographic algorithms is not just a future concern, but a present necessity. They emphasize that the transition to post-quantum cryptography should be gradual and well-planned, involving the coexistence of classical and post-quantum algorithms during the transition period. This approach ensures the continuity of security measures while adapting to emerging quantum threats. The authors also highlight the importance of standardizing post-quantum algorithms, which is crucial for their widespread adoption in decentralized systems [24]. This research provides valuable insights for developing robust security strategies for decentralized databases in the face of advancing quantum computing technologies.

Research by Deineka et al. [25] on designing data classification and secure store policy according to SOC 2 Type II provides valuable insights into ensuring regulatory compliance and data security in decentralized systems. This work is particularly relevant for DDBs that need to meet stringent security and privacy standards.

The development of decentralized identification systems (DID) and the concept of self-sovereign identity, described by Allen (2016) [6], opens new perspectives for improving identity management in DDBs. These approaches allow users to have full control over their identification data, which is an important step towards enhancing privacy and security.

An important direction of development is ensuring cross-blockchain interaction. Projects such as Polkadot, proposed by Wood (2016) [26], aim to create an infrastructure for effective communication between different blockchain systems. This can significantly expand the capabilities and application areas of decentralized systems, creating a single global ecosystem.

Recent research has also explored the application of decentralized database technologies in specific domains, demonstrating their versatility and potential for innovation. Balatska et al. [9] propose a concept for applying blockchain technology in the context of Single Sign-On (SSO) systems. Their work suggests that integrating blockchain with SSO can enhance security and user authentication processes, potentially revolutionizing access management in decentralized environments. This approach could be particularly beneficial for DDBs that require robust and secure authentication mechanisms.

Furthermore, Poberezhnyk et al. [10] have developed a concept for a learning management system based on blockchain technology. Their research illustrates how DDBs can be effectively utilized in educational settings, offering improved data integrity, transparent record-keeping, and enhanced security for student information. This application of blockchain in education demonstrates the potential of decentralized databases to transform traditional systems across various sectors, providing new solutions to longstanding challenges in data management and security.

Martseniuk et al. [27] propose an automated conformity verification concept for cloud security, which can be adapted for use in decentralized database environments to enhance security measures and ensure compliance with various standards.

Additionally, research by Yevseiev et al. (2023) [8] on security models of socio-cyber-physical systems emphasizes the importance of integrating DDBs with other modern technologies. Balatska et al. (2024) [9] consider the concept of blockchain application in the context of Single Sign-On (SSO) technology, opening new perspectives for improving the security and convenience of authentication in

decentralized systems. Poberezhnyk et al. (2023) [10] demonstrate the potential of DDBs in the educational sphere, proposing a concept of a learning management system based on blockchain technology.

In summary, it can be stated that decentralized database technologies have significant potential for further development and improvement. Addressing current challenges in the areas of scalability, confidentiality, regulatory compliance, and security paves the way for creating a new generation of distributed systems capable of meeting the growing needs of the modern digital world. Further research and innovation in this field are critical for realizing the full potential of decentralized technologies and their widespread implementation in various spheres of human activity.

5. Conclusions

The research emphasizes that blockchain-based decentralized databases, due to their distributed nature, can solve problems associated with centralized data storage and management systems. This allows avoiding a single point of failure and contributes to a higher level of user information protection.

The main aspect of the study lies in the careful examination and comparison of the advantages of various DDB technologies, such as BigchainDB, GUN, OrbitDB, Bluzelle, Fluree, and Ties.DB, and Hyperledger Fabric. The results show that these systems not only provide a high level of security but also contribute to solving problems of scalability, confidentiality, and access management.

The technical aspects of implementing authentication and authorization methods in DDBs are examined in detail, including the use of public key cryptography, smart contracts, and distributed access control. This can significantly increase the reliability of user identification processes and access rights management.

The results of the DDB technology analysis show that, despite their advantages in ensuring data transparency and immutability, there are problems related to scalability and compliance with regulatory requirements. The use of innovative approaches, such as sharding and Zero-Knowledge Proofs, can help solve these issues, providing an efficient and confidential data processing mechanism.

Additionally, it is important to note that DDBs can become a fundamental element in solving interoperability problems that often arise in traditional database systems. Their ability to provide a unified and reliable record of information can contribute to creating global data ecosystems without the risk of security breaches.

In the context of DDB development, it is important to consider collaboration between developers of different systems to ensure standardization and interaction between various platforms and protocols, especially in the field of cross-blockchain interaction.

References

[1] V. Zhebka, et al., Methodology for Choosing a Consensus Algorithm for Blockchain Technology, in: Digital Economy Concepts and Technologies, vol. 3665 (2024) 106–113.

[2] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System (2008).

[3] Z. Zheng, et al., An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends, IEEE International Congress on Big Data (2017). doi: 10.1109/BigDataCongress.2017.85.

[4] T. T. A. Dinh, et al., Untangling Blockchain: A Data Processing View of Blockchain Systems, IEEE Transactions on Knowledge and Data Engineering, 30(7) 1366–1385 (2018). doi: 10.1109/TKDE.2017.2781227.

[5] Regulation (EU) 2016/679 (General Data Protection Regulation) (2016).

[6] C. Allen, The Path to Self-Sovereign Identity (2016).

[7] A. Kosba, et al., Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts, IEEE Symposium on Security and Privacy (SP) (2016). doi: 10.1109/SP.2016.55.

[8] S. Yevseiev, et al., Models of socio-cyber-physical systems security: monograph, Technology Center (2023). doi: 10.15587/978-617-7319-72-5.

[9] V. Balatska, et al., Blockchain Application Concept in SSO Technology Context, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3654, (2024) 38–49.

[10] V. Poberezhnyk, V. Balatska, I. Opirskyy, Development of the Learning Management System Concept based on Blockchain Technology, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3550 (2023) 143–156.

[11] T. McConaghy, et al., BigchainDB: A Scalable Blockchain Database (2016).

[12] M. Nadal, GUN Documentation (2018).

[13] H. Haad, J. Nævdal, OrbitDB—Peer-to-Peer Databases for the Decentralized Web (2019).

[14] 7 Bluzelle Networks, Bluzelle: A Decentralized Database Service for dApps (2017).

[15] 8 B. Platz, A. Hilger, Fluree: A Practical Decentralized Database (2019).

[16] Ties.Network, Ties.DB: Distributed Database (2017).

[17] E. Androulaki, et al., Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains, in: Proceedings of the 13th EuroSys Conference, 30 (2018) 1–15. doi: 10.1145/3190508.3190538.

[18] S. Bano, et al., SoK: Consensus in the Age of Blockchains, in: Proceedings of the 1st ACM Conference on Advances in Financial Technologies (AFT) (2019) 183–198. doi: 10.1145/3318041.3355458.

[19] M. Zamani, et al., RapidChain: Scaling Blockchain via Full Sharding, in: ACM SIGSAC Conference on Computer and Communications Security (2018) 931–948. doi: 10.1145/3243734.3243853.

[20] Sharma, et al., Blurring the Lines between Blockchains and Database Systems: The Case of Hyperledger Fabric, in: Proceedings of the 2019 International Conference on Management of Data (SIGMOD) (2019) 105–122 doi: 10.1145/3299869.3319883.

[21] F. Reid, M. Harrigan, An Analysis of Anonymity in the Bitcoin System, Security and Privacy in Social

Networks (2013) 197–223. doi: 10.1007/978-1-4614-4139-7_10.
[22] M. Finck, Blockchain and the General Data Protection Regulation: Can Distributed Ledgers be Squared with European Data Protection Law? (2019).
[23] D. J. Bernstein, T. Lange, Post-Quantum Cryptography, NIST (2017).
[24] A. Horpenyuk, I. Opirskyy, P. Vorobets, Analysis of Problems and Prospects of Implementation of Post-Quantum Cryptographic Algorithms, in: Classic, Quantum, and Post-Quantum Cryptography, vol. 3504 (2023) 39–49.
[25] O. Deineka, et al., Designing Data Classification and Secure Store Policy According to SOC 2 Type II, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3654 (2024) 398–409.
[26] G. Wood, Polkadot: Vision for a Heterogeneous Multi-Chain Framework (2016).
[27] Y. Martseniuk, et al., Automated Conformity Verification Concept for Cloud Security, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3654 (2024) 25–37.