Proposal of Standardization of a Blockchain-Based Interoperability Platform for Academic Credentials

Luis Anido-Rifón

INDEX

1. Introduction

This deliverable entitled “Proposal of Standardization of a Blockchain-Based Interoperability Platform for Academic Credentials”, focuses on the standardization of interoperability among distributed ledger technologies (DLT) systems to establish the necessary processes for the issuance and verification of academic credentials across different educational institutions, both within the European Union and worldwide.

The increasing internationalization of academic and professional mobility requires systems capable of operating seamlessly and without barriers across borders, ensuring that the credentials issued are reliable, verifiable and recognized in a variety of heterogeneous domains. Trust in academic information - essential for decision-making in areas such as higher education, employment and lifelong learning - depends primarily on the ability of systems to issue, share and validate such data in a secure and transparent manner.

Distributed legder technologies (DLT) and particularly blockchains, revolutionized academic credential management by providing a secure, verifiable and decentralized means of issuing and validating academic credentials. However, despite their potential, these solutions are not yet universally accepted because they are still in the technical development phase and have important limitations, such as the fact that they must comply with stringent data protection regulations such as EU’s General Data Protection Regulation (GDPR) and that there is still missing a comprehensive legal framework to support them.

The Blockstand report discussing a proposal for a blockchain-based academic certification system that complies with the GDPR [1] introduced an innovative system for issuing and validating academic certificates, designed to integrate EU’s electronic identification, authentication, and trust services (eIDAS) and to comply with the GDPR. The proposal emphasizes scalability and flexibility, allowing academic information of any type to be recorded efficiently. Additionally, it enables institutions to adopt the system from various technical perspectives without requiring modifications to their existing information systems. However, as indicated in the conclusions of that report, to build solutions based on that proposal it would be necessary to address a technically complex critical challenge, namely the lack of interoperability among different blockchain networks. Currently, most systems operate

in isolation, which prevents the efficient interconnection of academic credentials among institutions organized in the form of private and consortium blockchains. This difficulty is a significant obstacle to the widespread adoption of the proposal, so that international academic and professional mobility can be enhanced with data protection guarantees without increasing the difficulty of sharing and verifying academic certificates.

As indicated above, in a context where education is increasingly global and students seek opportunities in different countries, it is instrumental to develop a blockchain-based academic certification model that guarantees the trust, validity and compatibility of degrees and accreditations anywhere in the world. To this end, it is necessary to establish standards that enable the interoperability of blockchain systems, facilitating the issuance, verification and recognition of academic credentials without geographical or institutional restrictions.

Currently, there is no mature or universally adopted solution that effectively addresses this challenge. The lack of technical and regulatory standards prevents the creation of a unified blockchain ecosystem for education that follows the model proposed in [1]. This issue not only affects academia but also impacts other areas where blockchain interoperability is essential, such as digital identity, rights management and asset traceability.

The challenge of standardizing the interoperability of blockchain systems for academic certification to ensure their smooth operation among different educational institutions, both within the European Union and globally, inspired us to contribute the proposal discussed in this report. Here, technological challenges are addressed, including compatibility between blockchain architectures and the implementation of standardized communication protocols, to provide a general conceptual framework for inter-blockchain communication. The conclusions of this study will highlight the complexity of the solution to be developed, although necessary for the implementation of the proposal in [1], which must guarantee the security, privacy and validity of academic credentials in a decentralized digital environment.

In summary, this technical report has the vocation to become a pioneering initiative to establish common standards that promote the integration and coordinated operation of various blockchain-based solutions, responding to the need for scalable, reliable, secure and globally interoperable systems. Note that the proposed framework could be applied to other fields beyond academic certifications where there is a requirement for interconnection and information stored in different blockchains. Consequently, it can be deemed relevant in the context of the European Blockchain Service Infrastructure (EBSI), and it can be of relevance to design standardized mechanisms supporting information from other blockchains to be

transferred to EBSI's, or vice versa, by providing a standardized, secure, and scalable method for exchanging information (e.g., verifiable credentials or other verifiable data) across independent blockchain networks. EBSI aims to facilitate trusted cross-border digital services within the EU, and this framework enhances its interoperability by enabling seamless communication between blockchains. By leveraging smart contracts for structured data transfer, ensuring compliance with eIDAS and GDPR, and maintaining decentralization, the solution supports EBSI's vision of trusted, transparent, and tamper-proof digital interactions. Moreover, its adaptability allows for integration with existing and future blockchain networks within EBSI, extending its potential applications to areas such as identity verification, digital diplomas, and public administration services, fostering a more interconnected and efficient European digital ecosystem.

2. Context: Blockchain-based Verification of Academic Credentials

Blockstand report [1] introduced a comprehensive proposal for the issuance and validation of academic certifications based on blockchain technology, designed to meet the demanding requirements of the GDPR, which is reinforced by the integration of identification and trust mechanisms in accordance with the eIDAS framework. In this model, a hybrid architecture is proposed that combines private blockchains -leveraged by issuing institutions to record the verification information of their academic certificates- and a consortium blockchain that centralizes the information and enables data verification, allowing third parties to validate the authenticity of credentials without compromising the privacy of personal information.

This proposal has the key elements discussed below.

2.1. Data Management and Issuance of Academic Credentials

Educational institutions generate academic certifications and store them, including all relevant information, in their data-processing systems off-chain. In parallel, to ensure the integrity and authenticity of the information, the information strictly necessary for verification is extracted and recorded on the blockchain. More specifically, the root of a Merkle tree obtained by applying HMAC with unique keys for each piece of data involved in credential verification. This approach enables compliance with the GDPR data minimization principle, as sensitive personal data is not recorded on the blockchain.

2.2. Access Control and Accountability

The system incorporates a series of smart contracts (i.e., SCService/SCData, SCAccess and SCLog). SCService and SCData handle the academic certificate verification data, while SCAccess is responsible for access control and SCLog for logging all access or permission changes when consulting the academic information verification data. Thanks to this mechanism, the academic certification holder can dynamically and selectively authorize or revoke access to third parties, ensuring that only those actors duly identified and authenticated through eIDAS may consult the verified information. In addition, each interaction is immutably recorded on-chain, providing full traceability of access and modification operations.

2.3. Flexibility and Scalability

The proposed architecture is designed to adapt to different types of academic certifications (e.g., formal, non-formal, vocational, informal) and to integrate with the institutions' pre-

existing systems through APIs or other mechanisms, using already mature technologies that fit the needs of each issuing entity. The use of private blockchains for initial registration, together with a consortium blockchain for subsequent storage and validation, makes it possible to distribute the operational load and ensure adequate performance even during periods of high academic data issuance.

2.4. Reliable and Secure Verification Mechanisms

The introduction of Merkle trees and HMAC codes with robust keys on each piece of validation data guarantee that any modification or attempt to manipulate the recorded information can be detected immediately, while anonymizing the information recorded in the blockchain [2]. This enables automatic and secure verification of the integrity of an academic certificate, even when its content is partially shared, allowing the verifier to reconstruct the tree and compare the root stored in the blockchain with that obtained from the data provided.

Overall, this proposal addresses the challenges inherent to the issuance and validation of academic certifications in a regulated digital environment and lays the foundation for the creation of an interoperable and scalable ecosystem. The processes and mechanisms developed in [1] are designed to ensure security, transparency and legal compliance with the GDPR, which in turn are fundamental aspects in the validation of academic information.

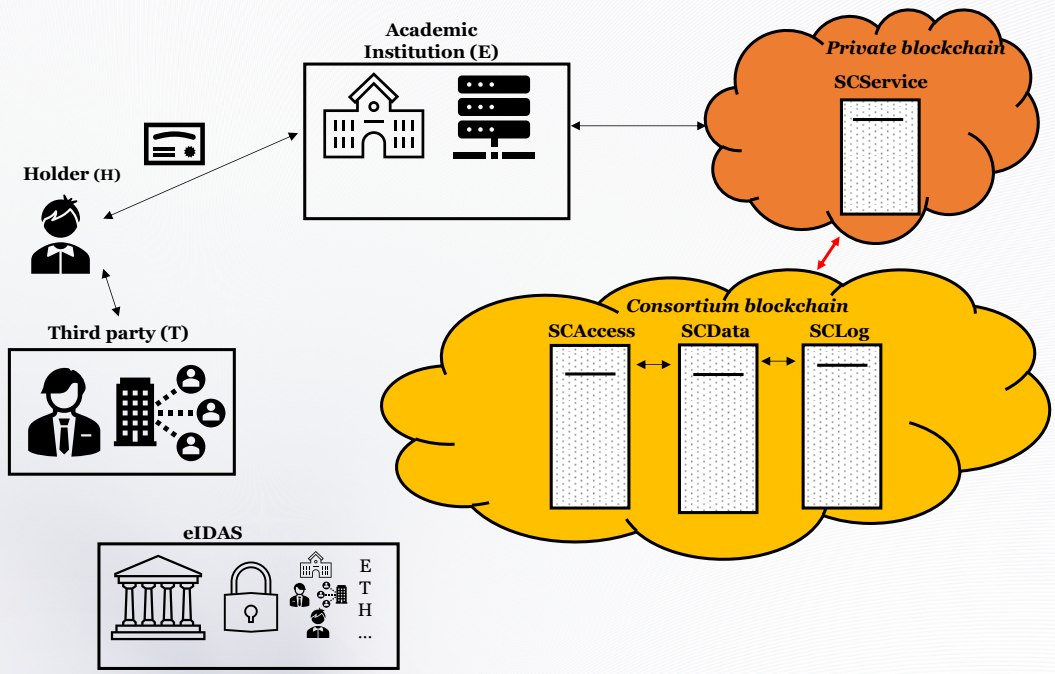

Figure 1 Elements involved in blockchain interconnection.

Figure 1 illustrates a simplified version of the model proposed in [1] in which the data to be recorded are abstracted, since the proposal is agnostic with respect to the size and structure of the information exchanged. Both the institution issuing the academic certificates (E), the certificate holder (H) and the third party that wants to verify them (T) are securely identified thanks to eIDAS. The elements to be covered in this report are highlighted in orange and red.

This deliverable builds upon this background to focus on the standardization of blockchain interoperability, defining protocols that enable the connection between different blockchain platforms. This standardized framework is intended to address the interconnection challenges identified in the field of academic certification in a way that can be extended to other use cases in sectors as diverse as health, public administration or international trade, contributing to the creation of globally recognized and reliable systems.

3. Related Work

Blockchain interoperability became a critical challenge as distributed ledger technologies (DLTs) expand beyond isolated ecosystems into interconnected networks that require seamless data exchange. Enabling secure, efficient, and scalable communication between independent blockchains is essential for advancing decentralized applications, financial systems, and digital identity management. The next paragraphs explore the state of the art in blockchain interoperability, focusing on both academic research initiatives and practical implementations that address this issue. By analysing different approaches to inter-chain communication, the current technological limitations are highlighted to serve as the foundation for an informed proposal of a standardized framework capable of facilitating complex data exchanges across diverse blockchain platforms.

3.1. Scientific Proposals

Interoperability among blockchains became one of the most important challenges in this field of application [2][4][5]. There are multiple research proposals that address this challenge from different prisms, seeking methods that enable secure, reliable and efficient communication among heterogeneous platforms. The main lines of work that have contributed to this field are discussed below.

For example, Cross-Chain Smart Contract Invocations (CCSCIs) are distributed transactions that involve the invocation of smart contracts hosted on two or more blockchain systems [6]. Tam Vo et al. [7] proposed the Internet of Blockchains, where different platforms communicate to facilitate inter-blockchain transactions, and data and state exchanges. The paper also identifies some technologies to enable enterprise environments with scalable blockchain interconnections. In a more practical approach, Westerkamp and Küpper [8] propose and prototype an EVM-based solution [9] to support the interaction between smart contracts running in separate blockchains by creating contract clients on the same execution environment, so that these clients are able to contain the logic and state of the original instance and thus allowing trans-chain function execution. These contracts provide instant read-only function calls to other applications hosted on the target blockchain, facilitating cross-chain communications.

In turn, Siris et al. [10] propose decentralized interledger gateway architectures for IoT authorization environments that support the interconnection of two ledgers, one of them

being the authorization ledger and the other a payment ledger. The proposed architectures vary in complexity, transaction cost and ability to handle transactions among multiple ledgers. Also, in the IoT domain, the same research team [11]proposed different models utilizing smart contracts and interledger mechanisms to provide decentralized authorization for constrained IoT devices, in a way that balances cost, latency, complexity and privacy, taking advantage of the features of smart contracts and communication among multiple blockchains through interledger mechanisms. These models are further evaluated in EVM test environments.

Kan et al. [12] propose a framework for information exchange between arbitrary blockchain systems. This architecture creates a dynamic multi-chain network for inter-blockchain communication, implementing a connection model that manages routing and message transfer. Additionally, they propose protocols to guarantee the atomicity and consistency of transactions taking place between chains. In tests conducted on a network of private blockchains, the results indicated a performance increase when compared to the parallel execution of multiple chains. Additionally, Want et al. [13] propose a cross-chain transaction processing flow that utilizes version control to manage transactions. Unlike traditional approaches based on locking mechanisms, the model proposed adopts an optimistic approach, allowing updated data to be used immediately. This model integrates additional mechanisms to guarantee atomicity in case of failures, ensuring that transactions are consistent and secure.

The General Purpose Atomic Crosschain Transactions protocol proposed by Robinson and Ramesh [14] allows the programming and execution of applications among multiple EVM-based blockchains without the need to modify anything of this technology, both in the case of public and private or consortium blockchains. The protocol supports calls both within the smart contracts themselves and among blockchains, and the calls are synchronous and atomic, so that if something fails, the whole process is reversed. Also, for EVM-based platforms and from the same authors, the Atomic Crosschain Transaction for Ethereum Private Sidechains protocol was introduced to support programming between permissioned blockchains [15], where calls between contracts and between synchronous and atomic blockchains are also allowed.

However, a recent literature review on blockchain interconnection [6], concluded that all initiatives analysed suffer from drawbacks that complicate real-world adoption, such as the low support for handling heterogeneity and the need for trusted third parties, which confirms that this challenge is still in an early stage of development. Besides, Zamyatin et al. [16] argue that cross-chain communication cannot be provided without a trusted third party. With this

perspective, they proposed a framework to evaluate existing cross-chain communication protocols and facilitate the design of new ones, focusing on the inherent trust assumptions, and derived a classification covering the field of cross-chain communication. Open challenges and the implications of interoperability on the security and privacy of blockchains are also discussed in that research.

3.2. Off-the-Shelf Implementations

In this context, several practical solutions beyond research projects and academic proposals are available to interconnect isolated blockchain networks. Each approach presents specific benefits and challenges in their quest for seamless blockchain interoperability, but they can address just the simpler use cases, such as fungible token transfers, and they are not mature enough to allow sending arbitrary data or making complex function calls between blockchains. To have a clear depiction of the state of the art in this field, some relevant developments are introduced in the next paragraphs.

Chainlink [16] is a decentralised blockchain oracle network built on top of Ethereum originally created to facilitate transfers of cryptocurrency-related information from on-chain ecosystems to off-chain ones and vice versa. By means of oracles, a blockchain could make its cryptocurrency information available to be read by an oracle, which in turn could feed that information to other blockchains or off-chain systems. To leverage this cryptocurrency functionality, Chainlink is implementing a Cross-Chain Interoperability protocol (CCIP), which aims to support the exchange of arbitrary data between blockchains, enabling decentralised applications on separated ecosystems to interact.

As a different approach, Cosmos [18] aims to create an Internet of Blockchains through its Inter-Blockchain Communication (IBC) protocol. The Cosmos ecosystem comprises multiple independent blockchains, each known as a zone, connected to a central hub. The IBC protocol facilitates secure and reliable communication between these zones, enabling the transfer of tokens and data across different blockchains within the Cosmos network. This architecture promotes interoperability and scalability, allowing each blockchain to maintain its sovereignty while participating in a larger, interconnected ecosystem.

Several projects are being developed to leverage the Cosmos SDK. Among the most relevant we can find EVMOS [19], the EVM on Cosmos. EVMOS is EVM-compatible blockchain platform designed to integrate the functionality of the EVM within the Cosmos ecosystem. EVMOS allows developers to deploy Ethereum-compatible smart contracts on a Cosmos-based blockchain, bridging the gap between these two blockchain technologies. By combining

the robust smart contract capabilities of Ethereum with the scalability and interoperability features of Cosmos, EVMOS enables interaction between Ethereum-based dApps and the broader Cosmos network. This integration enables the transfer of assets and data across these ecosystems, leveraging the Cosmos IBC protocol to enhance cross-chain communication. EVMOS also supports the use of familiar Ethereum development tools and languages, making it easier for developers to build and deploy dApps while benefiting from the enhanced performance and interoperability provided by the Cosmos infrastructure.

Polkadot [20] introduces a different approach to blockchain interoperability with its unique architecture consisting of a relay chain and parachains. The relay chain is the central blockchain responsible for network security, consensus, and cross-chain interoperability. Parachains are individual blockchains that connect to the relay chain, benefiting from the shared security and the ability to communicate with each other through the relay chain. This design enables seamless interoperability between parachains within the Polkadot ecosystem. Thus, Polkadot parachains offer another innovative solution for blockchain interconnection by providing an architecture allowing multiple blockchains to connect to a central relay chain, which facilitates cross-chain communication and interoperability while maintaining the security and consensus of the entire network. Polkadot's approach is particularly notable for its ability to support a wide variety of blockchains with different functionalities, making it a versatile solution for diverse use cases. However, Polkadot's interoperability is largely confined to its ecosystem of parachains. Connecting Polkadot to other external blockchains involves additional complexity and is not natively supported by the relay chain-parachain model.

LayerZero [21] is an omnichain interoperability protocol designed to connect blockchain networks in a decentralized manner. It aims to enable seamless communication and data transfer across different blockchains without relying on a centralized intermediary. LayerZero uses ultra-light nodes (ULNs) and decentralized relayers to facilitate secure cross-chain messaging and transactions. This protocol ensures that messages are securely delivered with minimal latency, making it a promising solution for developers seeking to build cross-chain applications that require efficient and secure interoperability. By focusing on the underlying infrastructure, LayerZero seeks to enable seamless communication between different blockchain networks without requiring significant changes to their existing architectures. This approach has the potential to simplify the integration process and enhance the overall efficiency of blockchain interconnections.

However, one relevant limitation of LayerZero is its reliance on external validators for cross-chain communication, which can introduce potential security vulnerabilities. While LayerZero aims to provide a universal interoperability layer, the use of external validators means that the security of the entire system depends on the trustworthiness and reliability of these validators. For this reason, it is highly recommended to have separated entities controlling the chosen oracle and relayer. If both were under the same validator and it is compromised or behaves maliciously, it could potentially disrupt the communication between blockchains or lead to unauthorised transactions. This reliance on external entities contrasts with fully decentralised solutions, where security is inherently distributed across the network. Besides, LayerZero owns the interoperability contracts in each chain integrated in its ecosystem. As a consequence, integrating a new chain in LayerZero requires transferring their control to LayerZero to be integrated in LayerZero's contract library. Therefore, while LayerZero offers significant advantages in terms of ease of integration and interoperability, it cannot be considered a fully decentralised solution, as none of the available IBC solutions are due to the isolated nature of blockchain[16]. Besides, LayerZero also centralizes the hosting of the protocol on its own infrastructure, not allowing the deployment of alternate endpoints by users and external developers. Therefore, implementing LayerZero not only requires a careful consideration and robust mechanisms to ensure validators' integrity and trustworthiness, but also a thorough auditing of data exchanged, due to LayerZero having centralized access to all transactions and their payloads through its endpoints.

Polyhedra [22] offers an innovative approach to blockchain interoperability by integrating advanced cryptographic proofs, particularly zero-knowledge proofs (ZKP [23]), to enhance the functionality of oracles. A ZKP is a cryptographic method that allows one party (i.e., the tester) to prove to another party (i.e., the verifier) that a claim is true without revealing additional information beyond the veracity of the claim itself, which ensures privacy and security during data exchange. Unlike traditional oracles that primarily focus on fetching and verifying off-chain data, Polyhedra's zero-knowledge oracles enable secure and private data transfer between different blockchains. By leveraging ZKPs, Polyhedra ensures that the data exchanged among chains remains confidential and tamper-proof, providing a higher level of security and privacy. This makes Polyhedra a compelling alternative for scenarios where sensitive information needs to be transferred across blockchain networks without compromising on privacy or security.

However, Polyhedra requires LayerZero to operate, which has control on endpoint contracts and libraries. Consequently, LayerZero drawbacks are also relevant here. Besides, the

complexity and the computational overhead associated with Polyhedra's geometric representation of blockchain data can be seen as an important drawback. While this approach offers enhanced security and efficiency in data transfers, it requires significant computational resources to process and verify the geometric structures. This can lead to increased latency and reduced performance, particularly in high-throughput environments where rapid transaction processing is critical. Additionally, the specialised nature of Polyhedra's cryptographic techniques may necessitate more advanced technical expertise for implementation and maintenance, potentially limiting its adoption among organisations with limited resources or technical capabilities. As a result, while Polyhedra presents a novel and secure method for blockchain interoperability, its practical application may be constrained by these performance and complexity challenges.

IBC YUI [24] is a cross-chain framework developed to facilitate interoperability between multiple blockchain networks and based on Cosmos IBC [25]. It is designed to support a wide range of blockchain protocols and aims to provide a standardized method for cross-chain communication. YUI focuses on enabling the transfer of not just tokens, but also more complex data and smart contract interactions across different blockchains. By providing a versatile and adaptable framework, YUI aims to simplify the development of cross-chain applications and enhance the interoperability of blockchain ecosystems. This makes it a valuable tool for developers looking to create more integrated and interactive blockchain solutions.

The IBC Transport Layer is agnostic to the kind of data or content transferred, providing a high level of flexibility. Complex data elements are transformed into bytes prior being sent to the destination chain by the corresponding packing and unpacking contracts, which hide the specificities of data elements. Being a transport-level protocol, supports authentication, packetization, retransmissions and guarantees data ordering. Each blockchain is represented by a Light Client that keeps the state of connected blockchains in a simplified way so every blockchain can know the state of the others before interacting with them. Channels keep the different modules on different blockchains connected to each other. Finally, Handlers and Relayers ensure that inter-blockchain communication is efficient, accurate, and secure. The Handler, implemented as a smart contract on each participating blockchain, acts as the primary interface for receiving and sending data. It ensures that data, whether simple or complex, is correctly encoded into a format suitable for transmission and properly decoded upon receipt. Additional contracts are responsible for managing the packing and unpacking of information into byte arrays, ensuring that the communication remains agnostic to the specific

data types being exchanged, as pointed out above. On the other hand, the Relayer functions as an off-chain component that monitors events on one blockchain and transmits the corresponding data to the other. It establishes a secure, bidirectional communication channel between blockchains, ensuring that data transfers occur in real-time and with high reliability. The Relayer listens for specific triggers from the Handler smart contract, such as transaction completions or status updates, and then relays this information to the handler contract on the target blockchain.

3.3. Analysis of related work

The solutions available off-the-shelf discussed above represent the state of the art in blockchain interoperability and also significant milestones in the development of this concept. At the same time, they highlight the current limitations and challenges of these approaches. More specifically, this fragmented landscape evidences the need for standardized comprehensive and flexible interoperability solutions that can facilitate complex data exchanges and interactions across different blockchain platforms. Exchanges like Chainlink compromise decentralization and offer unidirectional data feeds. Ecosystems like Cosmos and Polkadot provide robust interoperability within their respective networks but face difficulties extending this functionality to external blockchains. While LayerZero facilitates secure cross-chain communication, it still faces scalability challenges due to its reliance on decentralised relayers and ultra-light nodes and the requirement that new integrations must be approved and deployed by LayerZero. Polyhedra, while enhancing security and privacy, can introduce significant computational overhead and complexity and inherits LayerZero limitations. Despite its versatility, YUI's broad support for multiple blockchain protocols could lead to interoperability issues and performance inconsistencies across different networks and, insofar functionality implementation is concerned, it lags behind Cosmos, which relayer is being adapted by YUI to work within the Ethereum ecosystem.

Table 1 summarizes to what extent these solutions support blockchain interoperability in their current state. *Simple data* indicates whether the solution support the transfer of simple data values; *Token Exchange* identifies the proposals enabling the transfer of cryptocurrency tokens (i.e., all solutions discussed support crypto exchanges); *Arbitrary data* refers to the variable-length, multiple-field data elements required in academic credential verification; *Cross-chain Smart Contracts* refer to smart contracts able to interact with data from different chains and *Redundant Data* identifies the solutions that require the information being exchanged to be duplicated in both the origin and destination chains. Finally, *Open Design*

indicates whether all the components of the corresponding architecture are accessible to external developers. This analysis reveals a diverse landscape of technologies aimed at enhancing interoperability across different blockchain networks.

Note that Polkadot transfers are limited to the Polkadot parachain ecosystem and data is duplicated both in the parachain and relay chain.

Table 1. Summary of commercial blockchain interconnection solutions.

	Simple Data	Token Exchange	Arbitrary Data	Cross-chain Smart Contracts	Redundant Data	Open Design
Chainlink	Yes	Yes	Yes	Yes	Yes	No
LayerZero	Yes	Yes	Yes	Yes	Pending	No
Polyhedra	Yes	Yes	Yes	Yes	Yes	No
Cosmos EVMOS	Pending	Yes	Yes with proper implementation	Pending	Yes	Yes
IBC YUI	Yes	Yes	Yes	Yes	Yes	Yes
Polkadot	Yes	Yes	Yes	Yes	Yes	Yes

Considering the different initiatives discussed, it can be argued that to support full blockchain interoperability, robust mechanisms for securely transmitting arbitrary data across different platforms are needed, as it requires more than just facilitating token transfers or simple data exchanges; it necessitates robust mechanisms for securely transmitting arbitrary data across different blockchain platforms. Existing frameworks struggle to implement efficient mechanisms to handle complex, structured data, which are essential not only for academic credential verification, but also for applications in decentralized finance (DeFi), supply chain management or healthcare, among others.

To enable true cross-chain communication, interoperability protocols must be flexible enough to encode, transmit, and verify complex data structures while ensuring security, consistency, and integrity. This requires advanced mechanisms, such as packing and unpacking smart contracts, to convert structured data into standardized formats (e.g., byte arrays) that can be transmitted across heterogeneous blockchain architectures. Additionally, the integration of cryptographic proofs and decentralized relayers is essential to prevent data manipulation and ensure that cross-chain transactions remain verifiable without introducing trust dependencies on centralized actors.

Furthermore, scalability is a key concern in inter-blockchain communication. As blockchain adoption grows, interoperability mechanisms must handle high transaction volumes without creating bottlenecks or excessive fees. Efficient data compression, batching strategies, and consensus mechanisms optimized for cross-chain transactions can help address these

performance challenges. Moreover, interoperability frameworks must comply with legal and regulatory requirements, such as GDPR and eIDAS, ensuring that personal or sensitive data is securely managed while enabling trusted identity verification across chains.

In summary, full blockchain interoperability requires a comprehensive and decentralized approach that supports arbitrary data transfers beyond simple transactions. By developing standardized communication protocols, cryptographically secure message transmission, and decentralized relayers, blockchain ecosystems can move toward a trustless, scalable, and truly interoperable future where diverse blockchain platforms seamlessly exchange structured information while preserving security, privacy, and efficiency.

4. Proposal

Our proposal leverages smart contracts and is inspired in existing inter-blockchain communication protocols to ensure reliable and efficient data exchange. The next sections introduce the different elements that configure this proposal for inter-blockchain communication, including a basic characterization of the information exchanged, the reference infrastructure (i.e., two intercommunicating but otherwise independent blockchains) and the system's software elements. Then, the transfer of complex data structures between blockchains is discussed by means of an inter-blockchain messaging protocol (IBMP, a key part of the solution proposed). Finally, an initial specification to be satisfied by an IBMP compliant with this proposal is provided.

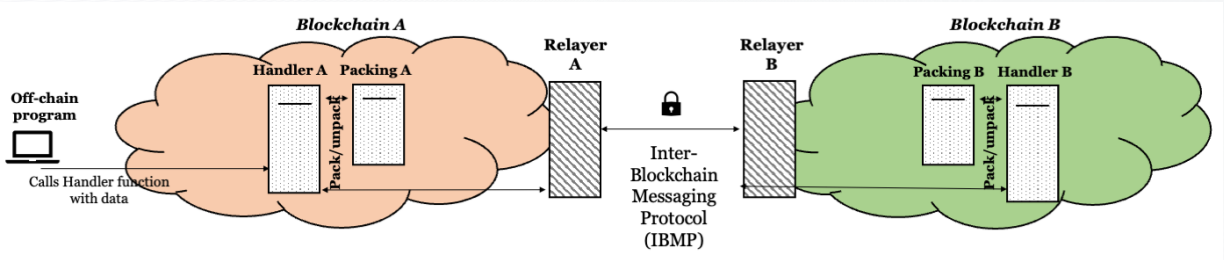

Figure 2 Elements involved in blockchain interconnection.

4.1. Components of the Proposal

The solution is composed of the elements below (cf. Figure 2):

a) Information exchanged. Arbitrary data with undetermined size, such as variable-length messages or complex data structures. It is assumed that data can be encoded (i.e., packed) as sequences of bytes (8-bit data elements), in a way that it is possible to decode (i.e., unpack) the original data structures. Byte arrays are the optimal choice for transferring information between blockchains because they provide a compact, standardized, and universally interpretable format that ensures efficient and secure data transmission. Unlike structured data types, which can vary across blockchain implementations, byte arrays are blockchain-agnostic, enabling seamless interoperability between heterogeneous systems. They minimize storage and computational costs by reducing the data footprint, which is critical in blockchain environments where transaction fees and processing efficiency are major concerns. Additionally, encoding arbitrary data structures into byte arrays allows for flexibility in handling diverse data types, including numerical values, text, and complex objects,

while ensuring that the original structure can be accurately reconstructed upon reception. This approach also enhances security by allowing cryptographic verification and integrity checks, preventing data corruption or unauthorized modifications during transmission. This encoding can be achieved with any existing encoding mechanism that supports the serialization of data structures as byte arrays, such as JavaScript Object Notation (JSON).

b) Infrastructure. Two blockchains (Blockchain A and Blockchain B) deployed on a blockchain technology that supports the smart contract concept, that is, self-executing Turing-complete[1] programs stored on a blockchain node(s) that automatically enforce and execute predefined rules and agreements when specified conditions are met. By encoding logic directly into the blockchain, smart contracts enable trustless interactions between communicating blockchains, reducing the risk of fraud, disputes, and inefficiencies in inter-blockchain communication. Examples of blockchain technologies implementing the smart contract concept are Ethereum, Polkadot, Cardano with the Plutus framework, Solana with on-chain programs, Hyperledger Fabric with chaincode, NEO with NeoContract, Algorand with ASC1 or Cosmos with CosmWasm.

c) System's logic. Smart contracts are deployed on both Blockchains A and B to address specific tasks in this proposal:

    a. Packing and Unpacking smart contracts that dynamically encode and decode the data based on its length and structure. The Packing smart contract on Blockchain A estimates the data size and encodes it into a byte array by packing all fields and appending their lengths to serve as offsets when decoding the message in Blockchain B.

[1] The virtual machines supporting the smart contracts in this proposal—Relayer, Handler, Packing and Unpacking—should be Turing-complete to ensure maximum flexibility, programmability, and adaptability in inter-blockchain communication. A Turing-complete virtual machine allows for the execution of arbitrary logic, enabling smart contracts to handle complex data structures, dynamic message encoding and decoding, and conditional execution of inter-chain transactions. Since the proposal involves securely transferring arbitrary-sized data between blockchains, the smart contracts must be capable of processing diverse formats, iterating over structured datasets, and managing state-dependent operations, which require loops, recursion, and conditional branching—all features exclusive to Turing-complete systems. Additionally, interoperability protocols often demand error handling, cryptographic verification, and adaptive mechanisms to process data across heterogeneous blockchain environments, further necessitating Turing-completeness. Without this capability, smart contracts would be severely constrained in their ability to encode, validate, and reconstruct transferred data, limiting their effectiveness in a generalized, cross-chain communication framework.

b. Handler smart contract. The Handlers, one on each blockchain, act as the primary interface for receiving and sending data. They ensure that data is correctly encoded into a format suitable for transmission and that it is properly decoded upon receipt.

c. Relayer smart contract. One Relayer on each blockchain, function as an off-chain component from the perspective of the other blockchain that monitors events on one blockchain and transmits the corresponding data to the other. It establishes a secure, bidirectional communication channel between blockchains, ensuring that data transfers occur in real-time and with high reliability. Relayers listen for specific triggers from Handlers, such as transaction completions or status updates, and then relay this information to the Handler contract on the target blockchain.

d) Inter-blockchain messaging protocol (IBMP). The IBMP is a framework original to this proposal that enables secure and reliable communication between independent blockchain networks A and B. Its specification is discussed below.

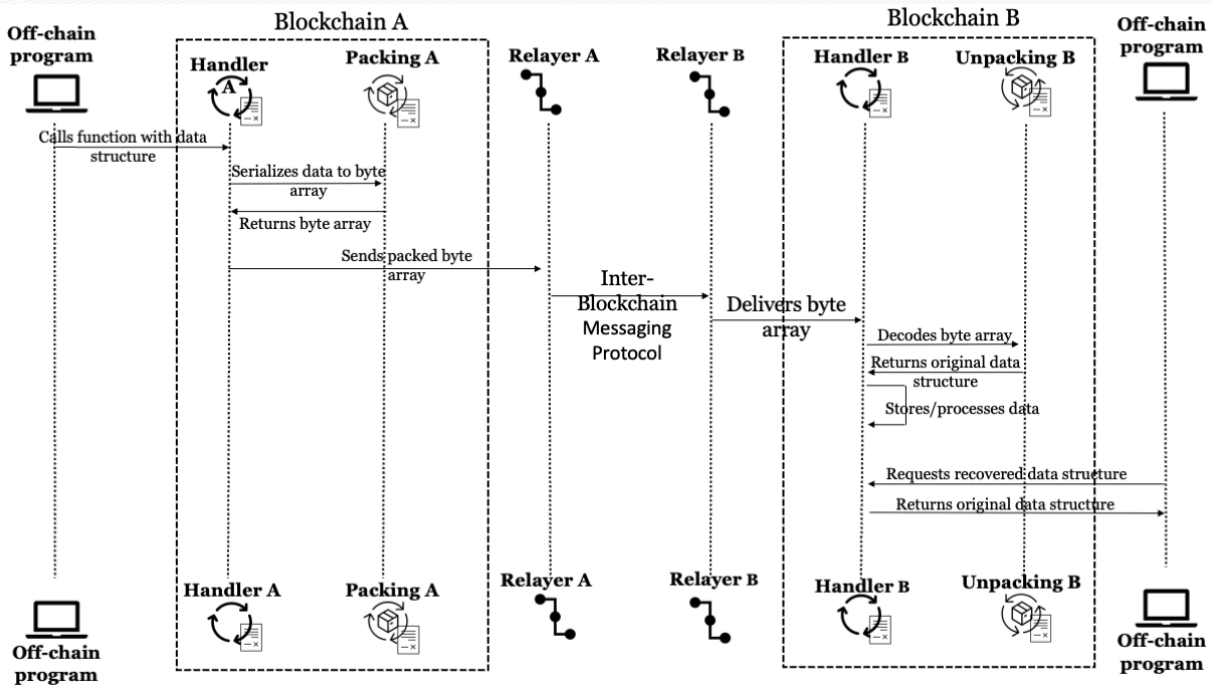

Figure 3 Inter-blockchain data transfer: sequence of events.

4.2. Inter-Blockchain Data Transfer

Using the elements above, the transfer of complex data structures from Blockchain A to Blockchain B proceeds as follows (cf. Figure 3):

1. **Invocation of the Handler in Blockchain A**
    a. An off-chain program calls a function of the **Handler** contract deployed on Blockchain A.
    b. The function receives an **arbitrary data structure** as input.
2. **Packing the Data**
    a. The **Handler** contract in Blockchain A invokes the **Packing** contract.
    b. The **Packing** serializes the arbitrary data structure into a **byte array**, ensuring it can be efficiently transmitted between blockchains.
3. **Relaying the Packed Data**
    a. The **Handler** contract in Blockchain A passes the packed byte array to the **Relayer** contract.
    b. The **Relayer** in Blockchain A sends the byte array to its counterpart **Relayer** in Blockchain B.
    c. The transfer occurs over an inter-blockchain messaging protocol, ensuring secure and reliable data transmission.
4. **Receiving and Unpacking in Blockchain B**
    a. The **Relayer** in Blockchain B delivers the byte array to the **Handler** contract on Blockchain B.
    b. The **Handler** contract in Blockchain B calls the **Unpacking** contract to decode the byte array back into its original **data structure**.
5. **Making the Data Available**
    a. The **Handler** contract in Blockchain B stores or processes the recovered data structure as needed.
    b. The data is now available for retrieval by other smart contracts or external applications.
6. **Retrieving the Data from Blockchain B**
    a. An off-chain program calls the **Handler** contract in Blockchain B to fetch the recovered data structure.

b. The **Handler** contract returns the original data structure, completing the end-to-end inter-blockchain transfer.

As it can be inferred from the events introduced above, the process of transferring complex data structures from Blockchain A to Blockchain B follows a structured approach to ensure secure, reliable, and efficient cross-chain communication. First, an off-chain program invokes the Handler contract in Blockchain A, submitting an arbitrary data structure. The Handler then calls the Packing contract, which serializes the data into a byte array, making it suitable for transmission. The Relayer contract facilitates the transfer by securely sending the packed data to its counterpart Relayer in Blockchain B through an inter-blockchain messaging protocol. Upon arrival, the Handler contract in Blockchain B receives the byte array and invokes the Unpacking contract, which reconstructs the original data structure. Once unpacked, the Handler stores or processes the data, making it available for smart contracts or external applications on Blockchain B. Finally, an off-chain program can retrieve the transferred data by calling the Handler in Blockchain B, which returns the recovered data structure, completing the end-to-end inter-blockchain transfer. This modular approach ensures seamless interoperability, preserving the integrity and accessibility of data across blockchain ecosystems.

4.3. Specification of the Inter-Blockchain Messaging Protocol

As pointed out above, the inter-blockchain messaging protocol (IBMP) is a framework that enables secure and reliable communication between independent blockchain networks. Any IBMP can be integrated in this proposal as far as it complies with the following requirements:

1. Interoperability.
2. Standardized data encoding.
3. Security and integrity.
4. Reliability and finality.
5. Atomicity and consistency.
6. Decentralization and trust.
7. Scalability and performance.
8. Flexibility and extensibility.
9. Cross-chain event handling.
10. Auditability and transparency.

These characteristics collectively define a robust inter-blockchain messaging protocol, enabling decentralized networks to communicate efficiently while maintaining security, reliability, and flexibility. These requirements are further elaborated below.

4.3.1 Interoperability

The protocol must facilitate data exchange between blockchains with different architectures, consensus mechanisms, and smart contract capabilities, ensuring compatibility across heterogeneous networks.

For an inter-blockchain messaging protocol to be effective, it must be capable of facilitating data exchange between blockchains that operate under different architectures, consensus mechanisms, and smart contract environments. Blockchains vary significantly in their design (e.g., some use Proof of Work PoW- or Proof of Stake -PoS- consensus, while others implement federated or hybrid models). Additionally, blockchains may support different virtual machines, such as EVM, WebAssembly (WASM) based smart contracts (e.g., Polkadot or Cosmos via CosmWasm), or specialized execution environments like Plutus in Cardano. This diversity introduces challenges in ensuring seamless communication and data consistency across networks.

To achieve true interoperability, the protocol must define a common data exchange standard that is agnostic to the underlying blockchain technology. This requires using a universal data format, such as byte arrays, which can be efficiently encoded and decoded regardless of the blockchain's data structure. Additionally, smart contracts involved in cross-chain communication must be designed to translate complex data formats into a universally interpretable form, ensuring that blockchain-specific representations do not hinder message processing.

Another key challenge is ensuring that transaction finality—the point at which a transaction is considered irreversible—aligns across different blockchains. Some blockchains provide instant finality, while others rely on probabilistic finality, where transactions become more secure as additional blocks are added. The messaging protocol must include mechanisms to verify and validate the state of transactions before acting upon them, preventing inconsistencies and potential vulnerabilities.

Moreover, smart contract execution models vary, with some blockchains enforcing gas limits and execution constraints, while others provide more computational flexibility. The protocol must account for these differences, ensuring that cross-chain transactions remain executable

within the computational constraints of all participating networks. This may involve optimizing contract logic to reduce execution costs or designing adaptive message-handling mechanisms that adjust based on the receiving blockchain's constraints.

Finally, achieving interoperability also requires a decentralized and trust-minimized approach to data transmission. This means that no single entity should control the relaying process, and cryptographic techniques, such as Merkle proofs, zero-knowledge proofs (ZKPs), or threshold signatures, should be used to verify the authenticity of messages across chains. By addressing these challenges, an inter-blockchain messaging protocol can enable seamless, secure, and efficient cross-chain communication, unlocking new possibilities for decentralized applications that span multiple blockchain ecosystems.

4.3.2 Standardized Data Encoding

Messages must be encoded as byte arrays, a blockchain-agnostic format that ensures that data remains interpretable regardless of the underlying blockchain technology.

As introduced in Sect. 4.3.1, for an inter-blockchain messaging protocol to function across heterogeneous blockchain ecosystems, it must adopt a universal and blockchain-agnostic data encoding format. Byte arrays provide the ideal solution because they offer a compact, efficient, and standardized method of representing structured and unstructured data. Unlike high-level data structures, which can vary significantly across blockchains due to differences in virtual machines, data storage models, and execution environments, byte arrays are a low-level, universally interpretable format that can be easily processed by any blockchain that supports smart contracts.

Encoding messages as byte arrays ensures compatibility and consistency, preventing format mismatches when transmitting data between chains with different architectures. For example, an EVM-based blockchain may store data using ABI (Application Binary Interface) encoding, while a WASM-based blockchain might utilize a completely different serialization mechanism. By standardizing messages in byte array format, the protocol eliminates the need for each blockchain to interpret complex native structures, reducing the risk of data corruption or misinterpretation during transmission.

Additionally, the use of byte arrays allows for efficient storage and transmission, which is critical in blockchain environments where transaction fees and gas costs are directly influenced by data size. Encoding data in a compact form minimizes on-chain storage

requirements and network bandwidth consumption, making cross-chain interactions more scalable and cost-effective. Furthermore, using generally accepted serializable formats like JSON or packing structured data into a byte array with offsets and length indicators enables dynamic and flexible message construction, allowing smart contracts to reconstruct the original data structure on the receiving blockchain with precision.

To ensure robust data integrity, cryptographic hashing and signature verification mechanisms can be applied to encoded byte arrays before transmission. This guarantees that messages remain untampered and verifiable, even when relayed across multiple networks. Additionally, using compression techniques before encoding can further optimize performance, particularly for large or frequently transmitted data structures.

Finally, implementing encoding and decoding libraries as standardized smart contract modules on each participating blockchain ensures seamless integration into diverse blockchain environments. These libraries can provide serialization and deserialization functions, ensuring that any blockchain can pack, transmit, and unpack messages correctly. By adopting standardized byte array encoding, the inter-blockchain messaging protocol establishes a universal foundation for secure, efficient, and interoperable data exchange, enabling diverse blockchain ecosystems to communicate without format conflicts or computational inefficiencies.

4.3.3 Security and Integrity

> *The protocol should support cryptographic techniques to ensure data authenticity, prevent tampering, and protect against replay or man-in-the-middle attacks. Digital signatures, hash functions, and encryption mechanisms are some examples of technologies that can be used to satisfy this requirement.*

Security is a fundamental requirement for any inter-blockchain messaging protocol, as cross-chain communication introduces new attack vectors that can compromise the authenticity, integrity, and confidentiality of transmitted data. To ensure that messages exchanged between blockchains remain tamper-proof and verifiable, the protocol must incorporate cryptographic security mechanisms that protect against unauthorized modifications, replay attacks, and man-in-the-middle (MITM) attacks. Without robust security measures, an attacker could intercept, alter, or duplicate messages, leading to fraudulent transactions, inconsistent state updates, or unauthorized access to sensitive data.

One of the primary methods for ensuring message authenticity is the use of digital signatures. When a message is generated on the sending blockchain, it should be cryptographically signed by the smart contract or entity responsible for its creation. This allows the receiving blockchain to verify the sender's identity and confirm that the message has not been altered during transmission. Public-key cryptography (e.g., ECDSA, EdDSA) enables this verification process, ensuring that only valid, signed messages are accepted and processed by the receiving blockchain.

To protect against data tampering, the protocol should implement cryptographic hash functions (e.g., SHA-256, Keccak-256). By hashing each message before transmission and including the hash within the message payload, the receiving blockchain can recompute and compare the hash to detect any alterations. If the computed hash does not match the original, the message is considered invalid and rejected. Additionally, Merkle trees and proofs can be used to efficiently verify the integrity of batched messages, reducing the computational overhead for validation.

Preventing replay attacks—where an adversary intercepts and retransmits a valid message to execute unauthorized actions—requires the use of nonces, timestamps, or sequence numbers. A unique identifier should be attached to each message to ensure that it is only processed once. The receiving blockchain should check for duplicate message IDs and discard any attempts to replay previous messages. Time-sensitive operations can also incorporate cryptographic time-locks to ensure that messages expire after a predefined period, further reducing the risk of replay attacks.

To defend against MITM attacks, where an attacker intercepts communication between blockchains, the protocol should support end-to-end encryption using symmetric or asymmetric encryption techniques. Encrypting message payloads ensures that only the intended recipient blockchain can decrypt and process the data, preventing unauthorized third parties from gaining access. Secure key exchange mechanisms, such as Diffie-Hellman key exchange or threshold cryptography, can be employed to establish secure communication channels between blockchains.

Finally, decentralized validation mechanisms can enhance security by reducing reliance on any single intermediary. If relayers or validators are used to transport messages between chains, they should operate in a trust-minimized environment, where they cannot alter messages without detection. Multi-signature schemes, threshold cryptography, and zero-

knowledge proofs (ZKPs) can further reinforce the security of message validation, ensuring that cross-chain communication remains trustless and resistant to manipulation.

By integrating these cryptographic security measures, the inter-blockchain messaging protocol can provide strong guarantees of authenticity, integrity, and confidentiality, protecting against adversarial threats while enabling secure and reliable cross-chain interactions.

4.3.4 Reliability and Finality

Messages must be reliably delivered with mechanisms to handle failures, retries, and confirmations. The protocol should also ensure that once a message is confirmed on the source blockchain, it is finalized on the destination blockchain without inconsistencies.

For an inter-blockchain messaging protocol to be effective, it must guarantee reliable message delivery and transaction finality, ensuring that data is consistently and accurately transferred between blockchains without the risk of loss, duplication, or inconsistencies. Since blockchain networks operate asynchronously and independently, various factors such as network congestion, transaction fees, consensus delays, or unexpected failures can impact the successful transmission of cross-chain messages. To address these challenges, the protocol must implement robust failure-handling mechanisms, message retries, and confirmation processes to ensure that every transmitted message reaches its destination and is processed correctly.

One key requirement for reliability is message persistence and retry mechanisms. If a message transmission fails due to network issues, transaction rejections, or relayer downtime, the protocol should automatically reattempt delivery until confirmation is received from the destination blockchain. Implementing acknowledgment receipts ensures that a sender blockchain does not consider a message successfully transmitted until an explicit confirmation is received. Additionally, timeout mechanisms should be in place to detect long delays and trigger alternative actions, such as re-routing the message through a different relayer or reverting the transaction.

To prevent message duplication, the protocol should implement idempotency controls, ensuring that the same message is not processed multiple times on the receiving blockchain. This can be achieved by assigning unique transaction identifiers or nonces to each message, allowing the receiving blockchain to track processed messages and discard duplicates.

Ensuring transaction finality is equally critical in inter-blockchain communication. Finality refers to the irreversibility of a confirmed transaction, meaning that once a message is successfully processed on the destination blockchain, it cannot be rolled back, altered, or invalidated due to a reorganization or chain fork. However, different blockchains have varying finality guarantees—for example, Ethereum uses probabilistic finality, where transactions become more secure as more blocks are added, whereas other networks like Tendermint (Cosmos) or Polkadot offer instant finality through their consensus mechanisms. The protocol must account for these differences and implement safeguards to confirm transactions only after they reach an appropriate level of finality on the source blockchain before proceeding with execution on the destination blockchain.

As discussed below (cf. Sect. 4.3.5), another important aspect of finality is atomicity, ensuring that either the entire cross-chain transaction succeeds or none of it is executed. Without atomicity, there is a risk that a message could be confirmed on the source blockchain but fail on the destination blockchain, leading to incomplete state updates and inconsistencies. Solutions such as Hashed Time-Locked Contracts (HTLCs), commit-reveal schemes, or cross-chain state proofs can be used to enforce atomic execution, ensuring that both chains reach a consistent state.

Additionally, verifiable state proofs—such as Merkle proofs, fraud proofs, or zero-knowledge proofs—can be integrated into the protocol to provide cryptographic assurance that a message was successfully committed on the source blockchain before it is finalized on the destination blockchain. This reduces reliance on external validators and enhances trust in the finality of transactions.

By implementing fault tolerance mechanisms, retries, acknowledgments, and finality verification, the inter-blockchain messaging protocol ensures that messages are consistently delivered and executed without inconsistencies, enabling trustworthy and seamless cross-chain interactions in a decentralized ecosystem.

4.3.5 Atomicity and Consistency

> *It must support atomic transactions, meaning that either the entire message exchange succeeds or none of it is processed, preventing partial execution that could lead to data inconsistencies.*

As pointed out above, for an inter-blockchain messaging protocol to maintain the integrity of cross-chain transactions, it must ensure atomicity and consistency, meaning that either the

entire message exchange succeeds or fails as a whole, preventing scenarios where only part of the transaction is executed. Without atomicity, a situation could arise where a transaction is committed on the source blockchain but fails on the destination blockchain, leading to incomplete state updates, data mismatches, or financial discrepancies. This is particularly critical in scenarios involving token transfers, cross-chain smart contract execution, or multi-step workflows, where any failure in execution could result in funds loss, smart contract desynchronization, or inconsistent records between blockchains.

Atomicity can be enforced using mechanisms such as Hashed Time-Locked Contracts (HTLCs), commit-reveal schemes, or two-phase commit protocols. HTLCs ensure that a transaction is only completed if cryptographic proofs validate the exchange on both blockchains, otherwise, the transaction is automatically reverted. Similarly, commit-reveal mechanisms allow the destination blockchain to confirm receipt before the source blockchain finalizes the transaction, ensuring synchronous execution across networks.

Ensuring consistency between blockchains also requires state validation mechanisms. Since different blockchains operate independently, they do not have native knowledge of each other's states. To solve this, the protocol should integrate state proofs, such as Merkle proofs, fraud proofs, or zero-knowledge proofs (ZKPs), allowing one blockchain to cryptographically verify the status of a transaction on another blockchain before proceeding with execution. This prevents cases where a blockchain executes a transaction based on outdated or incorrect information, reducing the risk of inconsistencies.

Another approach to enforcing atomicity is using cross-chain smart contracts that act as intermediaries to lock assets or data until all conditions are met. If any condition fails, the contracts ensure that the transaction is automatically reversed on all involved blockchains, preserving a consistent global state.

Additionally, the protocol should include rollback mechanisms to handle failures. If a cross-chain transaction fails at any point, it must be possible to revert all related operations, ensuring that no blockchain is left in an incomplete or incorrect state. This can be achieved through transaction dependency tracking, where execution steps are monitored, and if any part of the process fails, previous steps are undone automatically.

By enforcing atomic execution, state validation, and rollback mechanisms, the inter-blockchain messaging protocol ensures that data remains synchronized and trustworthy across multiple chains, preventing fragmentation, inconsistencies, and unintended partial

executions. This guarantees that cross-chain transactions are as reliable and secure as single-chain transactions, allowing decentralized applications to operate seamlessly across different blockchain ecosystems.

4.3.6 Decentralization and Trust Model

The protocol should avoid reliance on centralized intermediaries by means of trust-minimized mechanisms such as zero-knowledge proofs if necessary.

A truly effective inter-blockchain messaging protocol must minimize reliance on centralized intermediaries, ensuring that cross-chain communication remains trustless, censorship-resistant, and secure. Centralized relayers, validators, or third-party bridges introduce single points of failure, security vulnerabilities, and potential manipulation risks that contradict the core principles of blockchain technology. Instead, the protocol should be designed with trust-minimized mechanisms, allowing blockchains to verify messages and transactions autonomously without requiring external entities to guarantee correctness.

One approach to achieving decentralization is the use of zero-knowledge proofs (ZKPs), such as zk-SNARKs or zk-STARKs, which allow a blockchain to cryptographically verify the validity of a cross-chain message without relying on a centralized relayer. By generating cryptographic proofs that confirm the correctness of a transaction without revealing sensitive details, ZKPs ensure both data privacy and trustless verification, reducing dependency on external validation services.

Another decentralized approach is the use of light clients, which act as on-chain verifiers of external blockchain states. A light client embedded in one blockchain can efficiently verify the state and transactions of another blockchain without requiring full node synchronization. This enables direct, trustless verification of cross-chain transactions, removing the need for intermediary relayers or validators.

In cases where relayers are necessary, the protocol should ensure that they operate in a decentralized manner, such as through a network of competing relayers incentivized to act honestly. Mechanisms like staking, slashing penalties, and multi-signature (multi-sig) validation can prevent malicious activity by requiring multiple independent entities to confirm a transaction before it is accepted. Additionally, a threshold cryptography approach, where no single entity has full control over message transmission, further strengthens decentralization.

Decentralized governance models can also enhance trust by allowing protocol upgrades, security fixes, and policy changes to be determined collectively rather than by a single controlling entity. A DAO-based governance mechanism could enable participating blockchains to vote on protocol changes, ensuring that decision-making remains transparent, decentralized, and resistant to unilateral control.

By adopting zero-knowledge proofs, light clients, multi-party validation, and decentralized governance, the protocol ensures that inter-blockchain communication remains secure, verifiable, and free from centralized control. This trust-minimized design enhances resilience, prevents manipulation risks, and upholds the core principles of decentralization, making the protocol more scalable, censorship-resistant, and adaptable to evolving blockchain ecosystems.

4.3.7 Scalability and Performance

The protocol should optimize bandwidth usage, reduce computational overhead, and support batching or compression techniques to enhance scalability.

For an inter-blockchain messaging protocol to be practical and widely adopted, it must be designed to handle high transaction volumes efficiently, minimizing network congestion, computational overhead, and excessive costs. As blockchain ecosystems expand and more chains seek to interoperate, the protocol must ensure that cross-chain communication remains fast, cost-effective, and scalable, preventing bottlenecks that could degrade network performance.

One key factor in scalability is bandwidth optimization, ensuring that cross-chain messages do not overload the network with unnecessary data. The protocol should incorporate efficient data encoding techniques, such as byte arrays with structured offsets, to reduce the size of transmitted messages. Additionally, data compression algorithms can be applied before transmission, significantly reducing the payload size, which in turn lowers transaction fees and speeds up message propagation across networks.

To further enhance performance, the protocol should support message batching, allowing multiple transactions to be aggregated into a single transmission. Instead of sending multiple small messages individually, batching reduces on-chain storage requirements and minimizes the number of cryptographic verifications needed for message authentication. This technique

is particularly beneficial in high-frequency cross-chain applications, such as multi-asset transfers, decentralized finance (DeFi) interactions, or large-scale data synchronization.

Reducing computational overhead is also critical, especially in blockchain environments where gas fees and execution limits constrain smart contract operations. The protocol should ensure that packing, unpacking, and verification processes are optimized, avoiding redundant computations and leveraging precompiled cryptographic functions where possible. Implementing off-chain computation for complex tasks, such as zero-knowledge proof generation or advanced cryptographic validations, can further reduce the burden on the blockchain, improving transaction efficiency without compromising security.

Another aspect of scalability is parallel processing. Instead of executing cross-chain messages sequentially, the protocol should allow asynchronous execution where possible, enabling transactions to be processed in parallel without waiting for the completion of prior messages. This is particularly important in networks that support sharding, rollups, or sidechains, where decentralized applications (dApps) require real-time interoperability without introducing delays.

Lastly, the protocol should be adaptive to blockchain-specific constraints, ensuring that it remains efficient across diverse environments. Some blockchains impose strict gas limits, others have variable block times, and some require high-latency finality mechanisms. The protocol should dynamically adjust message sizes, processing methods, and validation steps based on the capabilities of each participating blockchain, ensuring optimal performance regardless of network conditions.

By integrating data compression, message batching, computational optimizations, off-chain computation, parallel processing, and adaptive execution strategies, the inter-blockchain messaging protocol can achieve high scalability and performance. This ensures that cross-chain communication remains fast, cost-efficient, and capable of supporting large-scale blockchain interoperability, even as adoption grows, and transaction volumes increase.

4.3.8 Flexibility and Extensibility

The protocol should be adaptable to different use cases, supporting any message payload serializable as a byte array (e.g., JSON data).

An inter-blockchain messaging protocol must be designed with flexibility and extensibility in mind to accommodate a wide range of use cases and evolving blockchain applications. Since

different blockchain ecosystems serve diverse purposes—including financial transactions, identity verification, supply chain tracking, academic credential verification, and decentralized governance—the protocol must support the transmission of any type of data without being constrained to specific formats or structures. This requires a generic and adaptable message payload format that can efficiently handle various types of information.

To achieve this, the protocol should allow for serialization of arbitrary data structures into a byte array format, ensuring that messages can be transmitted across heterogeneous blockchains without loss of information. This flexibility enables the encoding of structured data formats, such as JSON, Protobuf, or CBOR, allowing complex payloads to be packed and unpacked seamlessly by smart contracts on both the sending and receiving blockchains. By standardizing this serialization process, the protocol ensures that all participating blockchains can correctly interpret transmitted messages, regardless of their internal data handling mechanisms.

Extensibility is also crucial for ensuring long-term adaptability. As new blockchain technologies, consensus mechanisms, and execution environments emerge, the protocol should remain modular and upgradeable, allowing for future enhancements without requiring major architectural overhauls. This can be achieved by defining flexible message schemas that support versioning, enabling new data formats to be introduced while maintaining backward compatibility with existing implementations.

Additionally, the protocol should support custom message types that allow developers to define application-specific payloads for unique use cases. For instance, an academic institution might encode student credentials, while a DeFi application might include multi-asset transaction details. By providing a generic but extensible messaging framework, the protocol enables cross-chain communication to be tailored to different industry needs while maintaining interoperability between diverse blockchain platforms.

Another aspect of extensibility is integration with existing and future interoperability solutions. The protocol should be compatible with various blockchain infrastructures, including public, private, and consortium blockchains, as well as emerging technologies like rollups, sharding, and Layer 2 scaling solutions. By ensuring compatibility with multiple blockchain paradigms, the protocol remains relevant even as blockchain ecosystems evolve.

Lastly, smart contract modularity is essential for extensibility. Instead of implementing rigid, monolithic smart contracts, the protocol should be structured as independent functional

components, such as Packers, Unpackers, Handlers, and Relayers, allowing different implementations to extend, replace, or enhance individual components based on specific use cases. This modularity ensures that new cryptographic techniques, validation methods, or efficiency improvements can be seamlessly integrated without disrupting the core protocol.

By supporting arbitrary data serialization, flexible message schemas, backward compatibility, application-specific extensions, and modular smart contract architecture, the inter-blockchain messaging protocol remains highly adaptable, future-proof, and capable of supporting an ever-expanding range of decentralized applications.

4.3.9 Cross-chain Event Handling

> *Cross-Chain Event Handling – Mechanisms should be in place to allow smart contracts or off-chain programs to react to events triggered on a remote blockchain, enabling automated workflows and decentralized applications to function seamlessly across multiple chains.*

To enable seamless automation and real-time interoperability between blockchains, an inter-blockchain messaging protocol must include cross-chain event handling mechanisms that allow smart contracts or off-chain applications to react to events triggered on a remote blockchain. This capability is essential for supporting decentralized applications (dApps), automated workflows, and cross-chain financial instruments, where actions on one blockchain must dynamically influence processes on another. Without effective event handling, cross-chain interactions would require constant manual intervention or inefficient polling mechanisms, limiting scalability and usability.

An ideal event-handling mechanism should support asynchronous communication, allowing a blockchain to emit an event that is captured, relayed, and processed by another blockchain or off-chain application without requiring direct synchronization. For example, a liquid staking protocol could trigger an event on Blockchain A when a user deposits assets, causing Blockchain B to automatically mint a corresponding wrapped asset. Similarly, in a cross-chain decentralized exchange (DEX), an order fulfillment event on one chain should instantly notify another chain to settle the corresponding transaction.

To achieve secure and reliable event propagation, the protocol should use event listeners and relayers that continuously monitor blockchain states for predefined triggers. When an event occurs, the Handler smart contract on the emitting blockchain records the event and passes it

to the Relayer, which securely transmits it to the receiving blockchain's Handler. The Handler on the destination chain then invokes the appropriate smart contract or dApp logic to process the event, ensuring that the response is executed with minimal latency and maximum reliability.

To prevent event manipulation or replay attacks, cryptographic proofs of event authenticity should be included in message transmission. This can be achieved using Merkle proofs, digital signatures, or zero-knowledge proofs (ZKPs) to ensure that only valid and untampered events are recognized by the receiving blockchain. Additionally, events should include sequence numbers, timestamps, or unique IDs to prevent duplicate processing and ensure event ordering is maintained.

For off-chain applications that need to react to cross-chain events, the protocol should expose event subscription mechanisms via APIs or WebSocket connections, allowing developers to build applications that listen for and respond to blockchain-triggered events in real time. This would enable integrations such as oracles reacting to on-chain changes, enterprise systems updating records based on blockchain events, or automated compliance checks triggered by on-chain transactions.

Furthermore, the protocol should support event filtering, enabling smart contracts or off-chain applications to subscribe to specific types of events rather than processing all emitted events indiscriminately. This improves efficiency by reducing unnecessary computations and allowing dApps to focus only on relevant cross-chain interactions.

By implementing reliable event propagation, cryptographic security, real-time processing, and event filtering, the inter-blockchain messaging protocol ensures that blockchains and applications remain seamlessly synchronized, enabling scalable, decentralized, and automated cross-chain workflows without requiring trust in centralized intermediaries.

4.3.10 Auditability and Transparency

> *Auditability and Transparency – Message transfers should be verifiable on-chain, providing an immutable record of inter-chain transactions that enhances trust and accountability.*

For an inter-blockchain messaging protocol to be trusted and widely adopted, it must ensure that all cross-chain message transfers are verifiable on-chain, creating an immutable and transparent record of inter-chain interactions. This guarantees that all transactions can be

independently audited, enhancing trust, security, and accountability across decentralized ecosystems. Without proper auditability, malicious actors could manipulate cross-chain transactions, exploit relayers, or introduce inconsistencies between blockchains without detection.

To achieve full transparency, each cross-chain message must be recorded on-chain in a way that allows anyone to trace, verify, and reconstruct the message's history. This can be implemented by logging key transaction details—such as message sender, recipient, timestamp, unique transaction ID, cryptographic proof, and message payload hash—within the smart contract responsible for processing the cross-chain communication. By doing so, the protocol ensures that all messages can be independently verified at any point in time, preventing disputes and unauthorized modifications.

Cryptographic techniques such as Merkle proofs and zero-knowledge proofs (ZKPs) can be leveraged to provide lightweight and efficient verification of cross-chain messages. For instance, a blockchain receiving a message can request a proof of inclusion from the sender blockchain, ensuring that the message was genuinely recorded and finalized before acting on it. Additionally, commit-reveal schemes can be employed to prevent relayers from tampering with messages by ensuring that their contents are publicly verifiable but not alterable once committed.

Another critical aspect of auditability is ensuring cross-chain message finality tracking. Since different blockchains follow different consensus models and finality mechanisms, the protocol must maintain a state registry that logs whether a cross-chain transaction has been successfully completed, pending, or failed. This prevents issues where a transaction appears finalized on one blockchain but fails to be delivered on the destination blockchain, avoiding inconsistencies or unintentional state divergence.

Transparency is also crucial for governance and security monitoring. Open-source dashboards, analytics tools, and blockchain explorers should be able to query and display cross-chain message history, allowing researchers, regulators, and developers to monitor trends, identify anomalies, and detect suspicious activity such as double spending, frontrunning, or relayer manipulation. By ensuring that all message transfers are publicly accessible and independently verifiable, the protocol strengthens the decentralized security model, reducing the need for trusted third parties in inter-chain communications.

Additionally, role-based access control can be incorporated for private or permissioned blockchain use cases, where certain entities (e.g., regulators or consortium members) may require audit access to specific cross-chain transactions while maintaining confidentiality for other participants.

By implementing on-chain logging, cryptographic verification, finality tracking, and transparent governance, the inter-blockchain messaging protocol ensures a high level of auditability and accountability, enabling decentralized networks to operate with greater security, integrity, and trustworthiness while maintaining the core principles of blockchain technology.

4.4. Use Case: Verifying Student Credentials Across Blockchains

A university operates its own private blockchain to facilitate the verification of student records, including degrees, certifications, and academic achievements, according to the provisions in [1]. To ensure broader verification and transparency, the university needs to transfer a student's credential verification data from its private blockchain to the consortium blockchain shared by multiple educational institutions and accreditation bodies. This enables third parties, such as employers or other universities, to verify credentials in a trustless manner.

The transfer of verification information will be completed as follows:

1. **Off-Chain Invocation of Handler in the Private Blockchain**
    - The university's system (an off-chain program) invokes the **Handler** smart contract on the **private blockchain**, providing the credential verification data as input.
    - The data structure includes the verification information discussed in [1].
2. **Packing and Encoding the Data**
    - The **Handler** contract calls the **Packing** contract to convert the structured data into a **byte array**, ensuring a standardized format for cross-chain transmission.
3. **Relaying the Packed Data to the Consortium Blockchain**
    - The **Handler** contract sends the encoded byte array to the **Relayer** contract.
    - The **Relayer** on the private blockchain securely transmits the byte array to its corresponding **Relayer** on the consortium blockchain.

4. **Receiving and Unpacking the Data in the Consortium Blockchain**
    - The **Relayer** contract on the **consortium blockchain** delivers the byte array to the **Handler** contract.
    - The **Handler** contract invokes the **Unpacking** contract, which decodes the byte array back into the original credential verification structure.
5. **Making the Data Available for Verification**
    - The **Handler** contract on the consortium blockchain stores the verified credential information in an **immutable and publicly accessible registry**.
    - The credential data remains accessible for third-party verification while preserving student privacy through cryptographic techniques, according to the provisions in [1].
6. **Credential Verification by a Third Party**
    - An employer or another academic institution queries the consortium blockchain via the **verification interface** provided by SCAccess [1].
    - The system retrieves the credential verification data and cross-checks the university's **eIDAS identity**, confirming authenticity without requiring direct communication with the issuing university.

This solution enables seamless data exchange between independently operated blockchains, allowing academic institutions, employers, and verification authorities to interact without relying on a centralized system. By using standardized data encoding and smart contracts, different blockchain networks can communicate efficiently while maintaining autonomy.

The use of eIDAS and cryptographic elements ensures that all credential verification data is tamper-proof and verifiable. Third parties can independently verify their authenticity without needing to contact the issuing institution directly. This prevents fraud and unauthorized modifications.

The solution employs hashing techniques and selective data sharing to protect student privacy. Sensitive information is kept off-chain, ensuring that only authorized parties can access specific details while preserving the integrity of the credential verification process.

By leveraging a hybrid blockchain architecture, the system distributes the computational workload efficiently. Private blockchains handle the high-frequency generation and

management of academic records, while the consortium blockchain is reserved for broader verification and public access. This reduces congestion and improves transaction throughput across networks.

The automation of credential verification eliminates the need for manual processes, significantly reducing administrative workload and processing times. Institutions and employers can instantly verify credentials on the consortium blockchain, enabling faster hiring processes, seamless academic credit transfers, and streamlined compliance with accreditation requirements.

5. Discussion

This proposal can be seen as an ambitious attempt to promote a standard for the interoperability of blockchain-based systems applied to the issuance and verification of academic credentials. This discussion addresses the key aspects, implications and challenges derived from the integration of the outcomes of [1], with a broader scope, as this proposal is agnostic with respect to the actual application context.

In other words, the approach taken in this standardization proposal abstracts from specific technical issues or details, which is essential for the eventual formulation of a standard that can be adopted transversally in different technological contexts and application fields. By decoupling the proposal from specific implementations - such as the exclusive use of Ethereum, the specific field of academic credential verification, or other particular blockchain technologies - the integration of various blockchain platforms is facilitated, allowing the solution to evolve in parallel to technological advances without becoming obsolete. This abstraction translates into greater flexibility and adaptability, critical aspects in a global and dynamic environment such as international academic mobility.

Note that defining a standard for interoperability has significant technical and operational implications. First, a standardized framework will facilitate connection and communication among different blockchains, removing technical barriers that have so far prevented collaboration among services deployed on heterogeneous platforms. For example, in [1], it is proposed that issuing entities interact through APIs or similar instruments with a private blockchain, which will be the one to upload verification data to the consortium blockchain. Standardized inter-blockchain communication would enable private blockchains in the model to upload verification data directly to the consortium blockchain.

Additionally, a standard that combines technical interoperability with security, privacy and compliance requirements (such as those defined by the GDPR) represents a move towards systems that are both efficient, robust against security threats and adapted to current legal requirements.

Despite the clear advantages of this proposal, its implementation also faces several challenges. First, the inherent complexity of interconnecting multiple blockchains requires a coordination attitude and consensus that are still developing. Interconnecting multiple blockchains is inherently complex due to differences in consensus mechanisms, data structures, transaction finality, and governance models across networks. Unlike traditional distributed systems, blockchains operate independently, each with its own security assumptions and execution environments. Achieving seamless interoperability requires a coordinated effort among developers, standardization bodies, and network participants to establish protocols that

ensure secure, efficient, and trust-minimized communication. However, consensus on interoperability standards is still evolving, as different blockchain ecosystems prioritize security, scalability, and decentralization in varying ways. This ongoing development reflects the challenge of aligning diverse technological and economic incentives to create robust, widely adopted cross-chain solutions.

On the other hand, continuous advancements in blockchain technology mean that even if an interconnection standard exists, integrating a new blockchain platform may require technical developments that extend the deployment schedule over time. In the blockchain field, new developments are continually occurring, as it is a relatively young technology and therefore difficult to keep up with updates.

Finally, while the abstraction of specific technologies broadens the applicability of an eventual standard, it also poses the challenge of defining sufficiently generic interfaces and protocols that can effectively integrate with emerging blockchain solutions, without losing robustness in terms of security and efficiency.

Thus, the proposed standardization for blockchain-based interoperability in academic credential issuance and verification represents a significant step toward establishing a globally trusted and scalable system. By integrating with eIDAS and ensuring compliance with data protection regulations, this framework provides a secure, transparent, and verifiable mechanism for academic institutions to share credential information across independent blockchain networks. This approach enhances trust and security and facilitates seamless verification processes, reducing administrative burdens while maintaining strict privacy controls.

A key strength of this proposal is its technology-agnostic design, which abstracts specific blockchain implementations and allows for broad adaptability. By defining interoperability mechanisms that do not depend on any single blockchain technology, this framework ensures long-term viability and flexibility. This approach enables institutions to adopt the solution without requiring major modifications to their existing digital infrastructure, fostering a more inclusive and accessible ecosystem for academic credential verification.

Furthermore, this proposal has the potential to extend beyond academic credentials to other domains requiring secure and verifiable data exchange, such as healthcare, public administration, and international trade. The ability to transfer complex data structures across blockchain networks using standardized communication protocols ensures a high level of reliability and efficiency. By providing a unified interoperability standard, this framework contributes to broader blockchain adoption in multiple sectors where decentralized and tamper-proof data verification is essential. For example, this proposal can help to leverage EBSI to serve as a clearinghouse or exchange point for other blockchain infrastructures

operating in other sectors across Europe, extending present EBSI's use cases to include others benefiting from inter-blockchain interoperability, such as worldwide healthcare data exchange, supply chain transparency and traceability or digital identity.

However, despite these advancements, several technical and regulatory challenges must be addressed before large-scale implementation can be achieved. The complexity of interconnecting multiple blockchain networks requires ongoing research and development to ensure optimal performance, scalability, and security. Additionally, establishing governance mechanisms for maintaining and evolving the standard will be crucial to accommodate emerging technologies and regulatory changes while preventing fragmentation of the ecosystem.

In conclusion, this proposal lays the foundation for a standardized, secure, and scalable blockchain interoperability framework for academic credential verification. While challenges remain, its potential to streamline credential verification, enhance global academic mobility, and provide a replicable model for other industries underscores its importance. Future work should focus on refining the technical specifications, conducting real-world pilot implementations, and fostering collaboration among stakeholders to ensure widespread adoption and long-term success.

6. Acknowledgement

The author would like to thank Manuel J. Fernández-Iglesias and Christian von Eitzen Delgado for their active contributions to this work.

References

[1] Anido, L. A Proposal for a Blockchain-based Academic Certification System that Complies with the GDPR. Blockstand report, 2024. Available at blockstand.eu

[2] Agencia española de protección de datos (AEPD) and European Data Protection Supervisor (EDPS), Introduction to the hash function as a personal data pseudonymisation technique, 2019. Available at https://edps.europa.eu/data-protection/our-work/publications/papers/introduction-hash-function-personal-data_en

[3] Kannengießer N, Pfister M, Greulich M, Lins S, Sunyaev A. Bridges Between Islands: Cross-Chain Technology for Distributed Ledger Technology. In: Proceedings of the 53rd Hawaii International Conference on System Sciences. University of Hawaii at M¯anoa, Hamilton Library 2020:5298 –5307.24.

[4] Koens T, Poll E. Assessing interoperability solutions for distributed ledgers. Pervasive and Mobile Computing. 2019;59:101079. doi: 10.1016/j.pmcj.2019.101079

[5] Ou W, Huang S, Zheng J, Zhang Q, Zeng G, Han W. An overview on cross-chain: Mechanism, platforms, challenges and advances. Computer Networks. 2022;218:109378. doi: 10.1016/j.comnet.2022.109378

[6] Falazi G, Breitenbücher U, Leymann F, Schulte S. Cross-Chain Smart Contract Invocations: A Systematic Multi-Vocal Literature Review. ACM Comput. Surv. 2024;56(6). doi: 10.1145/3638045

[7] Tam Vo H, Wang Z, Karunamoorthy D, Wagner J, Abebe E, Mohania M. Internet of Blockchains: Techniques and Challenges Ahead. 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). 2018:1574–1581

[8] Westerkamp M, Küpper A. SmartSync: Cross-Blockchain Smart Contract Interaction and Synchronization. In: 2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). 2022:1 – 9

[9] Antonopoulos AM, Wood G. Mastering Ethereum: Building Smart Contracts and DApps. ch. 7 Smart Contracts and Solidity:127–160; 1005 Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly Media. 1st. ed. 2019.

[10] Siris VA, Tsenos M, Dimopoulos D, Fotiou N, Polyzos GC. Decentralized Interledger Gateway Architectures in Authorization Scenarios with Multiple Ledgers. In: 2020 Global Internet of Things Summit (GIoTS). 2020:1-6

[11] Siris VA, Dimopoulos D, Fotiou N, Voulgaris S, Polyzos GC. Interledger Smart Contracts for Decentralized Authorization to Constrained Things. In: IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). 2019:336 – 341

[12] Kan L, Wei Y, Hafiz Muhammad A, Siyuan W, Gao LC, Kai H. A Multiple Blockchains Architecture on Inter-Blockchain Communication. In: 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). 2018:139 – 145

[13] Wang W, Zhang Z, Wang G, Yuan Y. Efficient Cross-Chain Transaction Processing on Blockchains. Applied Sciences. 2022;12(9). doi: 10.3390/app12094434

[14] Robinson P, Ramesh R. General Purpose Atomic Crosschain Transactions. In: 2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS). 2021:61 – 68

[15] Robinson P, Ramesh R, Johnson S. Atomic Crosschain Transactions for Ethereum Private Sidechains. Blockchain: Research and Applications. 2022;3(1):100030. doi: 10.1016/j.bcra.2021.100030

[16] Zamyatin A, Al-Bassam M, Zindros D, et al. SoK: Communication Across Distributed Ledgers. In: Borisov N, Diaz C., eds. Financial Cryptography and Data SecurityFinancial Cryptography and Data Security. Springer Berlin Heidelberg 2021; Berlin, Heidelberg:3–36

[17] Breidenbach L, Cachin C, Chan B, et al. Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks. https://research.chain.link/whitepaper-v2.pdf; 2021.

[18] Kwon J, Buchman E. Cosmos Whitepaper. A Network of Distributed Ledgers. https://github.com/cosmos/cosmos/blob/master/WHITEPAPER.md; 2019.

[19] Dinesha DL, Patil B. A Conceptual Insight Into Achieving Interoperability Between Heterogeneous Blockchain Enabled Interconnected Smart Microgrids. In: 2023 IEEE PES Innovative Smart Grid Technologies - Asia (ISGT Asia). 2023:1-5

[20] Wood G. Polkadot: Vision for a Heterogeneous Milti-chain Framework. https://polkadot.com/papers/Polkadot-whitepaper.pdf; 2016.

[21] Zarick R, Pellegrino B, Banister C. LayerZero: Trustless Omnichain Interoperability Protocol. arXiv. 2021. doi: 10.48550/arXiv.2110.13871

[22] Xie T, Zhang J, Cheng Z, et al. zkBridge: Trustless Cross-chain Bridges Made Practical. In: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. Association of Computer Machinery 2022; New York, NY, USA:3003–3017

[23] Goldreich O, Oren Y. Definitions and properties of zero-knowledge proof systems. Journal of Cryptology. 1994;7(1):1–32. doi:10.1007/BF00195207

[24] Datachain. A Hyperledger Labs project that aims to achieve interoperability between multiple heterogeneous ledgers via IBC. https://www.datachain.jp/yui; 2023.

[25] Cosmos. IBC-Go Documentation, version 8.5. 2025. https://ibc.cosmos.network/v8/