ASSET SERVICING

Interoperability of Tokenized Assets: Towards a Secured and Unified Future in the Financial Industry

Singapore Blockchain Innovation Programme (SBIP) &
Northern Trust – Digital Assets and Financial Markets (DAFM) Group

September 2024

Singapore Blockchain Innovation Programme

NORTHERN TRUST

Co-authors

Alvin Chia
Head of Digital Assets Innovation
Asia Pacific
Northern Trust

Manish Kogundi
Head of Digital Assets Innovation Technology
Asia Pacific
Northern Trust

Nilesh Chandrakant Late
Digital Asset Innovation Lead
Northern Trust

Quang Trung Ta, Ph.D.
Senior Research Fellow
Singapore Blockchain Innovation Programme &
School of Computing
National University of Singapore

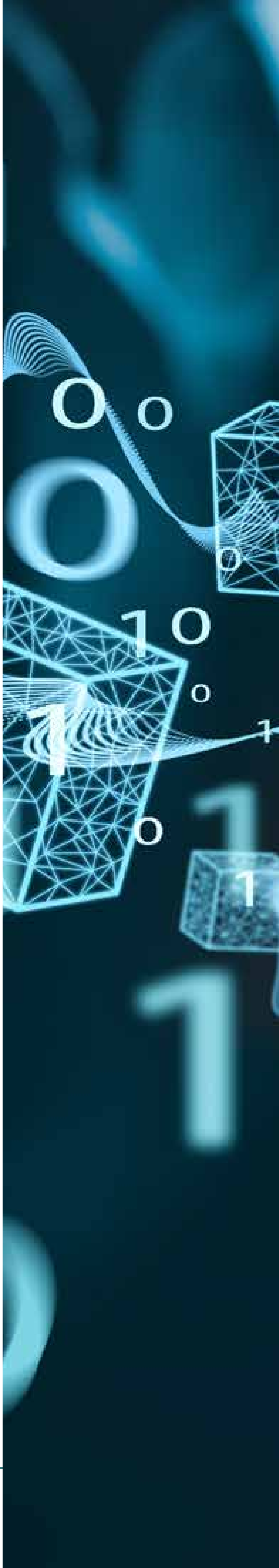
Nhut Minh Ho, Ph.D.
Senior Research Fellow
Singapore Blockchain Innovation Programme &
School of Computing
National University of Singapore

Contents

Contents

Executive Summary

The recent movement of tokenizing real-world assets using blockchain technology presents a significant opportunity for the banking industry to unlock tremendous value. Real-world assets are increasingly being listed and traded on many blockchain platforms. However, at the current stage, most of these chains are isolated and operating on their own protocol, making it difficult to trade across different chains. In order to scale digital asset trading and reach critical mass, we need to enable the movement of digital assets across chains, making it seamless and secure to tap into other liquidity pools. This whitepaper explores the challenges, best practices, technical considerations, and standards in building blockchain interoperability solutions for secondary trading of tokenized real-world assets.



1. INTRODUCTION

In recent years, blockchain technology has been disrupting the banking and financial industries. It enables a myriad of opportunities for tokenizing real-world assets so that owners and investors can trade them more securely and conveniently. In the asset tokenization process, the ownership and rights to real-world assets, such as real estate, commodities, stocks, or carbon credits, are recorded into digital tokens that are issued and stored on blockchains. The resulting tokens represent a stake of ownership in the underlying assets. They can be held, transferred, or traded.

Asset tokenization is beneficial from both the supply and demand perspectives. For example, from the supply side, tokenization offers a new channel for asset owners and financial institutions to raise funds. It creates efficient, cost-effective, and secure fundraising options for capital markets and trade finance by leveraging the latest advancements in blockchain technologies. From the demand side, tokenization makes investment opportunities more accessible and affordable to investors. It allows assets to be fractionalized into tokens, enabling both institutional as well as individual investors to easily buy and own fractions of the assets and then trade or transfer them on secondary markets.

While the boom of blockchain has made asset tokenization promising, it also fragments the market liquidity. Many blockchain platforms have been developed. Different financial institutions use different platforms to tokenize and manage assets. These chains are isolated, making it challenging to trade or transfer the assets across them. There are several protocols designed to make blockchains interoperable, however, they support mostly public chains, especially the ones based on the EVM (Ethereum Virtual Machine) execution model. Furthermore, many financial institutions, especially those in the banking sector, prefer to operate only on private blockchains or permissioned public chains. It can be arduous to connect these chains, especially public to private chains, or permissioned to permissionless chains, due to the difference in their design and implementation. Additionally, many institutions prefer to evolve their blockchain infrastructures, rather than migrating to public blockchains. All of these factors prevent digital asset trading from scaling across institutions and reaching critical mass.

In this report, we investigate the problem of designing and implementing an interoperability solution for trading tokenized assets across blockchains in secondary markets. We evaluate different considerations regarding technical feasibility, operational costs, security concerns, and regulatory compliance. We also implement a proof of concept to demonstrate the use case of transferring tokens between two blockchains and evaluate its operation and performance across testnets of several mainstream public blockchains. We also further discuss our findings when implementing such a system, and our next steps towards a future blockchain interoperability solution for trading tokenized assets.

2. PROBLEM STATEMENT

In this section, we outline the challenge of developing an interoperability platform that supports the trading of assets across different blockchains. The main objective of such a system is to enable the transfer of digital tokens from one blockchain to another blockchain. An interoperability system should have wide-ranging capabilities and should be able to support or be extensible to different kinds of source and destination blockchains. For example, the system should be capable of interfacing with private chains, which are the current preference of the banking industry and other financial institutions, and public chains, which are the predominate type in the blockchain industry. Furthermore, it's important to note that blockchains differ in their execution model. They can be classified into EVM chains such as Ethereum, BNB Chain, and Avalanche, or non-EVM chains, such as Polkadot, Sui, Aptos, Solana, etc. Since there are many options for the source and destination blockchains, different interoperability solutions will be needed to accommodate various chains. The complexity of each solution will be dependent on the compatibility of the two chains that it connects. For example, connecting two EVM chains is relatively easier than connecting an EVM chain to a non-EVM chain. Similarly, connecting a private chain to another private or public chain will require further authentication and verification.

While financial institutions mostly develop their business applications on private blockchains, some of them are investigating the potential of utilizing public blockchains. However, most of their applications work in silos and still do not support transferring or trading digital assets across these blockchains. To address the above problem, we surveyed and analyzed existing interoperability protocols for building systems that trade tokenized assets across blockchains. We also designed and developed a prototype system for public blockchains and evaluated its performance, execution model, security model, and operation cost.

In summary, this technical report aims to address the following three problems:

1. Investigate all requirements and considerations for developing a system that trades tokenized assets across different blockchains, including public, private, EVM, and non-EVM chains.

2 Survey existing interoperability protocols and evaluate them in the context of the cross-chain trading system.

3 Implement a proof of concept prototype for trading tokenized assets over public EVM blockchains and evaluate its performance, execution model, security model, and operation cost.

3. REQUIREMENTS AND DESIGN CONSIDERATIONS

Platforms for trading tokenized assets across both primary and secondary markets deal with highly valuable assets. These assets are stored on blockchains, and they require interoperability protocols to transfer between blockchains. Therefore, the platforms must be designed to not only ensure a smooth flow of business processes but also guarantee security, privacy, and integrity of the assets and transactions.

In the following, we discuss the requirements and design considerations of the system. We follow a top-down approach, outlined by the World Economic Forum [1], to categorize these requirements and considerations into two layers: the business and the platform layers.

Business model	• Governance model • Data Standardization • Legal Framework • Commercial Model
Platform	• Security • Regulatory Compliance and Privacy • Data Integrity and Auditability • Compatibility • Scalability, Efficiency, and User Experience

3.1 BUSINESS MODEL LAYER

The business model layer deals with different aspects related to the business application of the system for trading tokenized assets across blockchains. We discuss these aspects as follows.

3.1.1 GOVERNANCE MODEL

To ensure the credibility of participants such as users and traders in the system, a governance model must be established. The governance model should define who can participate in the system, their roles and responsibilities, the decision-making process, upgrade and maintenance management, and the dispute resolution mechanism. For instance, a user who is blacklisted by a regulatory body or a financial institution by a KYC (Know Your Customer) check should not be allowed to trade on the system. The governance model should also include the rules and procedures for updating the system and managing the system's capacity.

3.1.2 DATA STANDARDIZATION

When executing a cross-chain transaction, the system needs to handle data generated by the transaction in the source chain and process the data on the target chain. This data must follow a standardized format to be easily implemented and understood by all components of the system across different chains.

For example, to build a platform that supports transferring tokenized digital assets, it is important to choose the same token standard to represent the assets, such as ERC-20 for fungible tokens or ERC721 for non-fungible tokens (NFTs). These standards define a set of APIs that specify how tokens are created, managed, and transferred by smart contracts. Using these token standards not only simplifies the development of trading platforms but also makes them extensible to different kinds of compatible blockchains, thereby expanding their functionality and user base.

3.1.3 LEGAL FRAMEWORK

It is important to define a set of laws, regulations, rules, and guidelines that the system must comply with. These include legal considerations such as the jurisdiction where the system operates, the legal status of the digital assets, and the rights and obligations of the participants. The legal framework should also include the rules and procedures for resolving disputes and enforcing contracts to ensure transparency and fairness. This builds trust among users and upholds the integrity of the trading system.

3.1.4 COMMERCIAL MODEL

The commercial model of the system will decide whether it is attractive to a vast number of users and customers. Several considerations need to be taken into account. For example, who are the target customers? Are they individual or institutional users? What are the services provided to them? What are the transaction fees, platform costs, and incentives given to the users?

3.2 PLATFORM LAYER

The platform layer relates to the services provided by blockchain platforms and the interoperability protocols that connect them. These services should include the following requirements.

3.2.1 SECURITY

Ensuring the security of the system is critical since the system will handle the trading of high-value tokenized assets. Any attack or exploit on it could lead to significant financial losses and reputational damage. Attackers can steal assets, manipulate transactions, or disrupt the system’s operation. Consequently, the system must be designed with security in mind from the beginning. However, security needs to be considered in balance with other aspects of the system, such as performance and user experience. Overly strict security measures can lead to poor performance and bad user experience, which may discourage users from using the system.

A rule of thumb to secure a system is to examine and protect all of its components, including both the on-chain and off-chain components. Here, the on-chain components refer to the smart contracts deployed and maintained by the bridge operators on each participating blockchain. The off-chain components refer to the software stacks that interact with the contracts and the validators’ network and their software.

To secure the on-chain components, it is important to follow the blockchain and cryptocurrency industries’ standard practices, such as conducting smart contract auditing to find security vulnerabilities and business logic errors and choosing secure, trusted, and well-tested interoperability protocols to connect the participant blockchains. Similarly, to secure the off-chain components, it is important to use trusted and verified software stacks from credible libraries and vendors. They also need to be monitored and updated regularly with up-to-date security patches.

Furthermore, it is important to set up an incident response framework to detect and manage cyberattacks on the system in order to minimize damage, recovery time, and total costs. For example, transactions should be monitored in real-time to detect any abnormalities and take necessary action, such as halting the transaction to the entire system.

3.2.2 REGULATORY COMPLIANCE AND PRIVACY

The system should adhere to regulatory compliance requirements related to data privacy, such as the General Data Protection Regulation (GDPR) in the European Union or the Personal Data Protection Act (PDPA) in Singapore. It also needs to implement other compliance checks such as know-your-customer (KYC) and anti-money laundering (AML) measures.

One of the key unique selling points of a blockchain system is that once source code is deployed and data is stored, it cannot be altered easily. This clearly has implications for data privacy, particularly where the relevant data is personal data or metadata sufficient to reveal someone's personal details. Data protection regulation may require that personal data be kept up-to-date and accurate or be deleted at the discretion of the individual, and the immutability of a blockchain system may not be consistent with such requirements.

3.2.3 DATA INTEGRITY, AUDITABILITY, AND FINALITY

Ensuring the integrity and auditability of data stored and transferred over the system is critical since it guarantees that the data used by all participants is correct, unanimous, reliable, and traceable. Data integrity also helps to prevent attacks such as replay and double spending attacks [2], [3]. Auditability and traceability are mandatory to comply with regulatory checks such as anti-money laundering (AML).

Another important challenge is the issue of transaction and data finality. The system needs to guarantee that the quantity of transacted tokens will be available on the destination chain once those are burnt on the source chain. Without finality, a reversed transaction on the source chain, such as a block reorganization, might cause issues on the destination chain, leading to the incorrect update of account balances on two chains.

3.2.4 COMPATIBILITY

The underlying blockchains that store digital assets need to support compatible or similar standards to enhance interoperability between them. This compatibility allows different blockchain networks to communicate and share data seamlessly, which is essential for creating a more connected and efficient ecosystem, without the need for complex and costly integration processes.

Furthermore, employing similar standards enables greater security and reliability in cross-chain transactions. When blockchains use standardized methods for data exchange and transaction verification, it reduces the likelihood of errors and vulnerabilities that could be exploited by malicious actors. Consistency in standards also facilitates more straightforward auditing and compliance processes, as regulatory bodies and third-party auditors can more easily verify transactions and data integrity across interconnected networks. This builds trust among users and stakeholders, which is fundamental for the widespread adoption of blockchain technology.

3.2.5 SCALABILITY, EFFICIENCY, AND USER EXPERIENCE

Depending on use cases, the system should be able to handle a suitable volume of transactions with acceptable delays. It should also keep the operational costs, such as transaction fees, affordable to users, and the operation and maintenance costs, if required, reasonable to the owners.

The system should have an intuitive and user-friendly interface for managing transactions. It is also important to provide a convenient yet secure method for users to interact with their accounts and transactions, such as opening and managing accounts or taking custody of their accounts and digital assets.

4. INTEROPERABILITY APPROACHES

Choosing a suitable interoperability solution to connect participating blockchains is one of the decisive factors in building a system for trading tokenized assets. Existing solutions often require on-chain verification to validate transactions in the source chains and rely on authentication to execute the corresponding transactions in the destination chains. These solutions are also called the *cross-authentication methods*.

In this section, we will investigate the three existing cross-authentication-based interoperability approaches: atomic swap, notary scheme, and cross-chain bridge. We will evaluate their pros and cons, and whether each of them can be a useful component of the trading system. We will further survey the existing interoperability protocols built on top of these approaches in section 6.

4.1 ATOMIC SWAP

Atomic swap [4] is a peer-to-peer (P2P) mechanism that allows users to exchange digital assets from different blockchains without needing third parties like notaries or cross-chain bridges. The goal of atomic swaps is to reduce the steps required to trade tokens without centralized intermediaries like exchanges.

Atomic swaps use a smart contract technology, called hashed time lock contract (HTLC) – a time-bound smart contract, which acts as a virtual vault or cryptographic escrow account that keeps digital assets safe and only executes when the correct number of tokens has been deposited into the contract. Each user must acknowledge receipt of tokens within a specified interval to unlock them. HTLC enforces that all the executed transactions are either completed in full or not at all, thus ensuring that digital asset holders maintain the integrity of their tokens until the transaction is complete.

Atomic swaps offer several benefits such as reducing counterparty risks, making users maintain complete control over their assets, enabling more P2P flexible asset transfer use cases, and incurring lower fees than relying on third-party administrators. However, atomic swap is complex to use since users must agree on the amount and price of the transaction, the length of the time lock, exchange data, and hashes, and wait for transactions to be processed. Moreover, it only supports blockchains that are compatible: they must use the same hashing algorithm for atomic swaps to work.

4.2 NOTARY SCHEME

A notary scheme [5] is an interoperability technique that relies on a trusted third party, called a notary, to facilitate transactions between users on different blockchains. The notary can have one or many accounts in each chain. When a user in one chain needs to transfer assets to a user in another chain, the user first transfers assets to the notary's account on the source chain. The notary then locks and confirms these assets before transferring the equivalent assets from its account on the destination chain to the target user.

There are two types of notaries: a single-signature (centralized) notary or a multi-signature (decentralized) notary. The single-signature notary scheme uses only one node to collect and validate transaction data on the source chain and execute the corresponding transaction on the destination chain. While this notary scheme is simple to set up and operate, it is also vulnerable to failure and misbehavior due to the centralized risk of only one node. On the other hand, a multi-signature notary requires multiple nodes to verify the transaction data: the majority of these nodes need to agree and sign the transaction in the source chain before executing the transaction in the destination chain. Compared to the single-signature notary scheme, the multi-signature notary is more robust and secure. However, it also introduces extra overhead to the transaction execution.

4.3 CROSS-CHAIN BRIDGE

A cross-chain bridge [6] is a system that is designed as a separate blockchain that can read and validate events and states of other blockchains. This design gives the bridge the ability to facilitate the movement of digital assets between different blockchains. The process generally involves locking or burning the digital assets on the source chain using a smart contract and then unlocking or minting the corresponding assets on the target chain with a separate smart contract. More specifically, there are three main mechanisms to handle digital assets on a cross-chain bridge, as follows:

- **Lock and mint.** This mechanism is used to connect two chains that manage different types of digital assets. A user locks digital assets in a smart contract on the source chain and then mints the wrapped versions of those locked assets on the destination chain. In the reverse direction, the wrapped tokens on the destination chain are burned to unlock the original coins on the source chain.

- **Burn and mint.** This mechanism is used to connect two chains that manage the same kind of digital assets. To move the assets from one chain to another, a user burns the digital assets on the source chain, and then re-issues or mints the same assets on the destination chain.
- **Lock and unlock.** This mechanism is used to connect two chains that involve liquidity pools. A user can lock tokens on the source chain and then unlock the same kind of tokens from a liquidity pool on the destination chain. These types of cross-chain bridges usually attract liquidity on both sides of the bridge through economic incentives such as revenue sharing.

In addition to transferring digital assets, cross-chain bridges can also send arbitrary data across blockchains using smart contracts. This data can include information about the digital assets, such as their provenance, ownership history, or other metadata. By enabling the transfer of data along with the assets, cross-chain bridges can enhance the transparency, traceability, and security of digital asset transactions. They also enable more complex cross-chain functionalities, such as swapping, lending, staking, or depositing digital assets.

4.4 SUMMARY

Among the three cross-validation approaches above, cross-chain bridge is the most often used in practice. It is more versatile than notary schemes since it can transfer not only tokens but also arbitrary data seamlessly, thus enabling various use cases. It is simpler to use than atomic swaps since users do not have to perform a sequence of actions, from agreeing on the price and the length of the time lock, and then exchanging data, and hashes, and waiting for transactions to be processed. In the next section, we will investigate further cross-chain bridges as the potential interoperability solution for building a trading system for tokenized assets.

5. INTEROPERABILITY SOLUTIONS USING CROSS-CHAIN BRIDGES

5.1 SURVEY EXISTING BRIDGES

There have been several protocols developed to enable interoperability of blockchains. These protocols target different use cases, ranging from transferring only tokens to sending both arbitrary data and digital assets. In this section, we will survey existing cross-chain bridges which can transfer both data and tokens programmably across chains. These bridges are relevant to our project's scope of building a platform for trading tokenized assets across different chains. We chose to investigate the most popular bridges, which are Axelar Network, LayerZero, and Chainlink CCIP (Cross-Chain Interoperability Protocol).

There are also other cross-chain bridges, such as Wormhole [7], Circle CCTP (Cross-Chain Transfer Protocol) [8], and zkBridge [9]. However, at the time of writing this report, these protocols are tailored to focus more on cryptocurrency use cases. Hence, we do not survey them in this technical report. Audiences interested in these protocols can refer to the references for more information.

5.1.1 AXELAR NETWORK

Axelar Network [10] is a cross-chain communication platform that allows users to interact with applications across different blockchains. Essentially, Axelar Network itself is a blockchain that connects other blockchains. It consists of three key components: a decentralized network, a set of gateway smart contracts, and a set of APIs, libraries, and developer tools.

The first two components, the decentralized network and the gateway smart contracts are the core infrastructure of Axelar Network. The decentralized network includes a set of validators that run and maintain the Axelar Network. They are in charge of monitoring the connected blockchains and executing the cross-chain transactions. Axelar Network consists of 75 validator nodes [11] and can connect to 67 blockchains [12] (as of September 1, 2024). It supports all major EVM and non-EVM chains such as Ethereum, BNB Chain, Avalanche, Polygon, Cosmos, and Polkadot. Axelar Network will deploy on each of these connected chains a gateway, which is a set of smart contracts. The validators will monitor the gateway smart contracts for incoming transactions from a source chain. They will check for their consensus on the validity of the transactions, using the DPoS (Delegated Proof-of-Stake) consensus protocol, before deciding whether to execute the transaction on the destination chain.

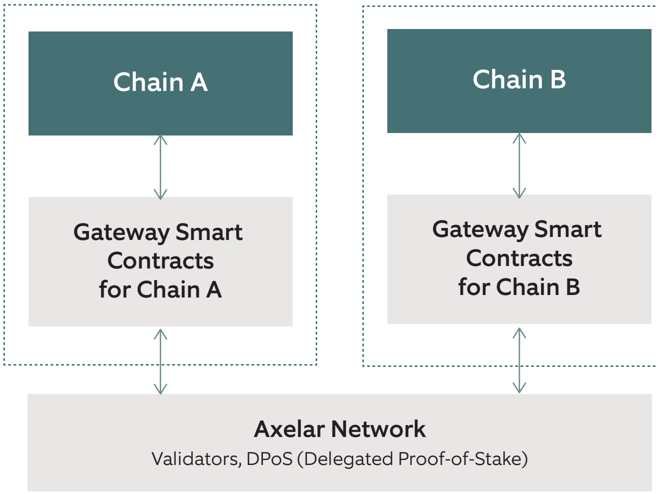

Chain A
Chain B
Gateway Smart Contracts for Chain A
Gateway Smart Contracts for Chain B
Axelar Network
Validators, DPoS (Delegated Proof-of-Stake)

Figure 1: Axelar Network architecture and workflow

Axelar Network provides APIs, libraries, and developer tools, which are built on top of the core infrastructure, for users to develop their applications. It includes services like General Message Passing (GMP), which facilitates secure, Turing-complete cross-chain computation, Interchain Token Service (ITS), which allows users to transfer tokens between different blockchains, or Axelar Gas Services, which automates the conversions of its native token AXL and the connected blockchain gas tokens to pay for transaction fees. These services allow Axelar Network to provide flexible cross-chain-communication capabilities and enable a wide range of use cases, from cross-chain swaps to the creation of wallets with universal borrow-lend features and various decentralized financial applications.

5.1.2 LAYER ZERO

LayerZero is a communication protocol that enables direct cross-chain transactions between different blockchain networks. Unlike Axelar Network, which relies on an independent blockchain to connect other chains, LayerZero does not require any intermediate blockchain. It includes LayerZero Endpoints, which are deployed to the connected blockchains. Each Endpoint consists of a set of smart contracts that implement a standardized interface for cross-chain applications. They create a direct link between every pair of the connected blockchains. These links form a fully connected network of blockchains, called LayerZero fabric, allowing users to send data messages between any two chains in the network.

Messages sent between two Endpoints are processed by off-chain components of the LayerZero network. In the latest version of LayerZero, V2, which was launched January 2024, there are two main off-chain components: a Decentralized Verifier Network (DVN) and an Executor. The DVN is responsible for verifying the messages and ensuring their correctness. It uses a set of verification algorithms to validate the messages and reach a consensus on their validity. Once a message is verified, it is sent to the Executor, which is a set of smart contracts that implement the logic of the cross-chain applications. It ensures that the messages are executed correctly and that the state of the destination chain is updated accordingly. This execution is in isolation from the verification conducted by DVN.

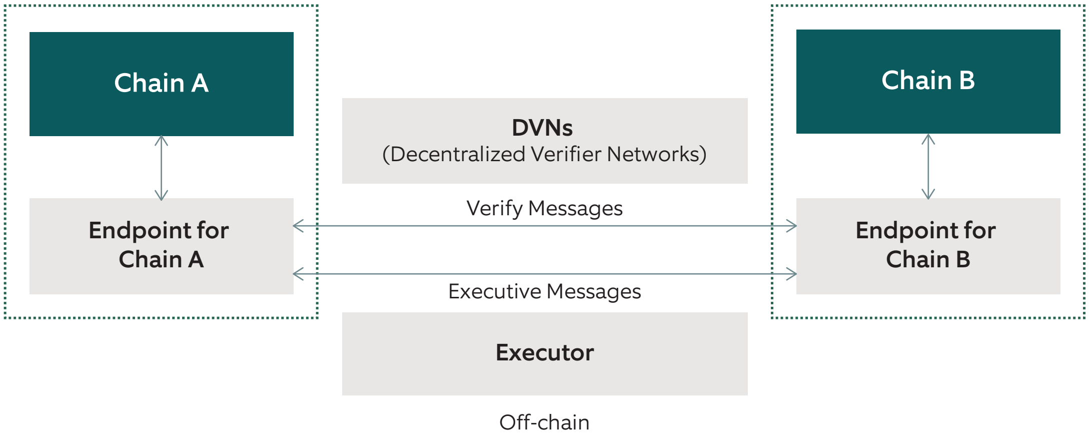


Figure 2: LayerZero V2. architecture and workflow

Currently, LayerZero supports 34 DVNs [13]. Developers can also run a custom DVN by running off-chain components and deploying a DVN contract on every chain they want to support. LayerZero protocol provides consistent security standards with application-level control where application owners can have their own security configurations, immutable core contracts to prevent smart contract upgrades from introducing vulnerabilities, and backward compatibility with previous LayerZero versions.

5.1.3 CHAINLINK CCIP

Chainlink CCIP is a cross-chain interoperability protocol that enables developers to make cross-chain transfer of arbitrary data and tokens. It aims to establish a universal connection to connect both private and public blockchains. Similar to LayerZero, Chainlink CCIP does not rely on any intermediate blockchain to transfer data across blockchains.

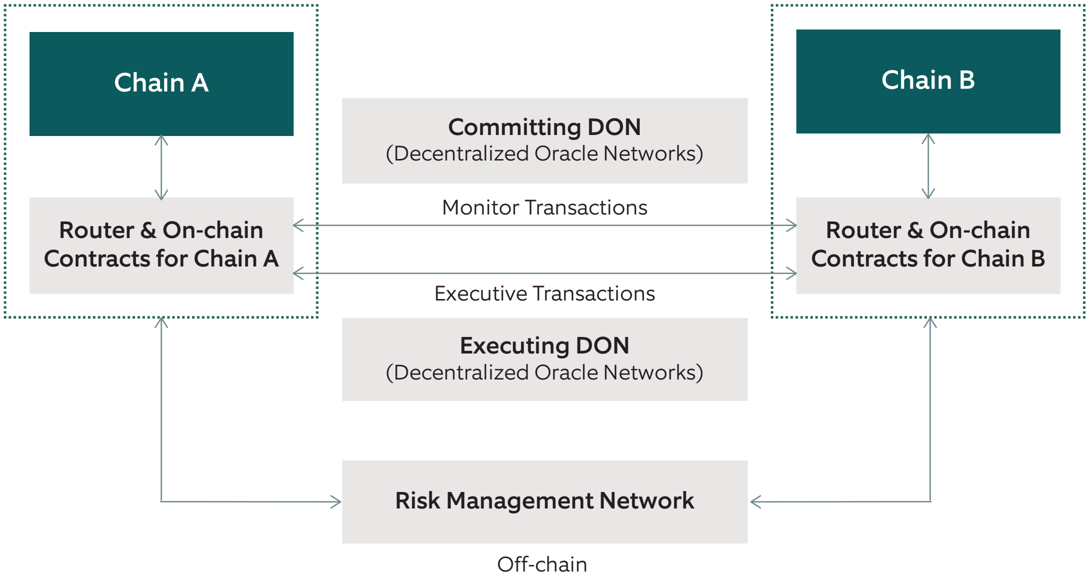

Chain A
Chain B
Committing DON
(Decentralized Oracle Networks)
Router & On-chain Contracts for Chain A
Monitor Transactions
Router & On-chain Contracts for Chain B
Executive Transactions
Executing DON
(Decentralized Oracle Networks)
Risk Management Network
Off-chain

Figure 3: Chainlink CCIP architecture and workflow

Chainlink CCIP's on-chain components include a set of smart contracts deployed on each participating blockchain, serving as routers, token pools, and network utilities to send, receive, and manage data and tokens. Off-chain components include two Decentralized Oracle Networks (DON): Committing DON and Executing DON, and a Risk Management Network. The two DONs are in charge of monitoring and executing the cross-chain transactions. The Risk Management Network is a secondary validation service, running parallel to DONs network to independently validate transaction data. Chainlink CCIP separates the Risk Management Network from the DONs to mitigate against security vulnerabilities that might affect the DONs' codebase.

Chainlink CCIP supports three main capabilities: arbitrary messaging, token transfer, and programmable token transfer. Arbitrary messaging is the ability to send data, encoded as bytes, from a smart contract on one blockchain to another smart contract on another chain. The token transfer feature allows users to transfer tokens directly to a smart contract or an externally owned account (EOA) on another chain. Programmable token transfer supports sending both data and tokens simultaneously within a single transfer. These three capabilities allow users to implement different application logic, from a simple token swap app to a more complex protocol like lending, with instructions to leverage those tokens as collateral for a loan and borrow another asset.

Chainlink CCIP can enable interoperability over 14 major blockchains on their mainnets [14] (as of September 1, 2024). They include Ethereum and other EVM-based blockchains such as Optimism, Avalanche, Arbitrum, Polygon, and BNB Chain.

5.2 BRIDGE ASSESSMENT

In this section, we explore and evaluate various cross-chain protocols. With the rise of interoperability solutions, the ability to securely and efficiently send messages and transfer tokens across different blockchain networks has become essential. This assessment aims to provide a detailed analysis of multiple bridges, focusing on their architecture, security features, and overall performance. Due to the time limitations of our study, we focused on the protocols we interviewed in the latter half of 2023 and the beginning of 2024. The candidates were initially shortlisted by Northern Trust based on their desired use case and existing track record supporting similar financial institutions.

Our objective is to compare their security and reliability in interoperable tokenized asset use cases. We understand the complexity and risks associated with these technologies, their continuous improvement, and enhancement, and strive to present a balanced evaluation within the scope of this study. This report will include an overview of the current landscape, criteria for assessment, and detailed findings for each bridge examined. Our assessment is inspired by the report by the Uniswap Foundation for their governance use case [15]. In June 2023, the foundation published the assessment report for the use case of relaying messages (governance decisions) between blockchains. Their use cases resemble our interoperable token application where the governance decision is analogous to the command to mint/burn remotely sent by cross-chain bridges. As the assessment was conducted last year with a different use case, we reassess them in this section.

5.2.1 ASSESSMENT FRAMEWORK

Compatibility and Flexibility — This refers to the number of supported blockchains and whether private deployment is possible. It also covers the potential use cases: is it simply a token transfer bridge or supporting general message passing? What are the extra features on top of message passing?

Security and Privacy — This covers the security guarantee of the bridges. In other words, it represents the risks associated with operating the protocol relating to the validity and correctness of the cross-chain transaction. In the previous framework [15], this involves (1) Protocol Architecture Risk Validation Mechanism, (2) Protocol Implementation and Operational Risk, and (3) Network Risk.

The criteria for protocol implementation and operational risk are usually the first to be minimized by the bridges. Their source code is audited by reputable organizations; the validators' model and incident response protocol and history have been updated following the bridge assessment report. Risk mitigation mechanisms such as rate limits, isolation, and monitoring frameworks have been introduced. We note that the popular bridges assessed in this report satisfy these criteria. For network risks, bridges have been using heuristics and well-known configurations for finality and have been considering rate-limiting mechanisms and isolation of failure. Hence, our security assessment focuses more on the protocol architecture risk and the track record and reputation of the solution.

Ease of Deployment and Development — This encompasses the ease of deploying a private setup based on the open-source repository and the complexity of integrating existing applications with the bridge contract APIs.

Performance — This covers the throughput and latency of the cross-chain protocol. We note that this is not the main focus for bridges operating on public blockchains due to the long finality time and low throughput being the bottleneck. In private settings, we can measure the peak throughput between two performant blockchains in theoretical settings where all transactions are cross-chain transactions.

5.2.2 DETAILED ASSESSMENT

From the list of bridges, we have shortlisted for interview and assessment, each has its own strengths and weaknesses, along with varying complexity in integration. We note that each bridge offers additional services such as interchain tokens, remote query, and multicast, which are beyond the scope of this assessment. The security of smart contracts is also a critical topic, and all teams have adhered to industry-standard practices by having external audits from leading organizations and maintaining high-quality source code; hence, this is also not within the scope of our assessments. Below are the assessments of the selected bridges:

Axelar Network. The usage of a blockchain to connect blockchains enhances security but with the cost of complexity to run and integrate.

- **Compatibility and Flexibility:** Axelar Network can be launched in private blockchain settings. According to the website [16], they support 66 chains, with the majority being EVM chains. They support both token transfer and general message-passing methods.
- **Security and Privacy:** The security of Axelar Network on public blockchains is guaranteed by the consensus of the 75 validator nodes. Based on the number of nodes and the security record, we conclude that the Axelar Network is highly secure on public blockchains. However, certain considerations need to be considered when deploying a private setup. The number of validators is configurable in such cases, reflecting the tradeoff between security and cost. We also note that the number of validators supporting each blockchain may vary and can be fewer than 75 as they are not required to run all supported chains. Privacy is not supported as this is a normal linkable and transparent bridge.
- **Ease of Deployment and Development:** The blockchain itself requires manual configuration and is not straightforward to deploy as a standalone service given the limited time budget of our study. The smart contract interfaces are simplified, allowing easy development of arbitrary applications.
- **Performance:** Due to the difficulty of benchmarking the real Axelar Network setup within our time budget, we used the indirect method of viewing comparative data. A detailed comparison will be presented in Section 6.4.

LayerZero V2. It is designed with an enhanced and configurable security feature. The integration and blockchain support are the best. However, the configurable security and libraries make it more difficult for development compared to other solutions.

- **Compatibility and Flexibility:** We deployed the LayerZero solution in private setups with little effort. Therefore, we conclude that it has the highest compatibility and flexibility in terms of usage. LayerZero V2 supports 78 blockchains, the most out of the assessed bridges. Like other bridges, it supports both token transfer and the more general message-passing method. Additionally, it offers more composability for other bridge services (e.g., Stargate) to operate on top of its message-passing solution.
- **Security and Privacy:** The security of LayerZero is configurable based on the choices and number of DVNs. As a result, its security is very reliant on the DVN choices. By default, two DVNs are configured, with each having three signers, requiring two signatures to reach a quorum. This default setting is considered less secure compared to other bridges. However, they can also utilize other bridge services to become one of their DVNs, thereby aggregating the security of other off-chain bridges

if chosen. Similar to other public bridges, LayerZero V2 is a normal linkable and transparent bridge and does not support any privacy-preserving schemes.

- **Ease of Deployment and Development:** Given the open-source repository and documentation, we managed to deploy the self-managed LayerZero solution on private settings, proving its deployability. However, development suffers from the complexity of configurability and requires setting up peers before running as well as customized configuration in hex value. The deployment complexity is considerably higher than other bridges.
- **Performance:** Our analysis and measurements show that the performance of LayerZero is generally constrained by off-chain processes and a chain of sequential intermediate steps required for finalization. Furthermore, the need to deliver messages sequentially reduces the observed throughput. A detailed evaluation of both public chains and private setups will be included in Section 6.4.

Chainlink CCIP. The reputable industry oracle service with a strong safety record offers Chainlink CCIP an edge in security. However, the integration and blockchain support are still limited.

- **Compatibility and Flexibility:** The Chainlink CCIP solution offers limited compatibility and flexibility due to the closed-source nature of the project. Running a private setup requires the team to manage and maintain the deployment. Additionally, it currently supports only nine blockchains, a much smaller number compared to other solutions. They support a variety of interoperability operations, including token transfer, token transfer with messages, and message passing with and without receipts.
- **Security and Privacy:** The security of Chainlink CCIP is ensured by three different Decentralized Oracle Networks, greatly enhancing security. Furthermore, each DON serves a different purpose. One feature is that the Risk Management Network can mitigate risks on cross-chain transactions by monitoring them and applying countermeasures such as emergency shutdowns or adjusting transfer limits. With its long-standing reputation in the industry and risk-mitigated architecture, we have no doubt that Chainlink CCIP is among the safest protocols available. However, the security also depends on the number of nodes and the decentralization of the DONs. These parameters are a trade-off between cost and security in private deployments. Like Axelar Network, Chainlink CCIP is a normal linkable and transparent bridge and does not support any privacy-preserving schemes.
- **Ease of Deployment and Development:** For deployment, Chainlink CCIP does not have an open-source self-managed solution that can be deployed. Development on top of Chainlink CCIP is extremely easy with their experience in providing contract libraries.
- **Performance:** Due to the closed-source nature of Chainlink CCIP, we used indirect methods to view comparative data. A detailed comparison will be presented in Section 6.4.

5.2.3 DISCUSSION

Our observation is that the bridges are secure enough for the interoperable tokenized solution, as they have been battle-tested on public blockchains with high-volume transfers. Each protocol has its own strengths for each assessment criterion.

Compatibility and Flexibility: LayerZero V2 supports the greatest number of blockchains. Axelar Network is second in this property for supporting a considerably high number of blockchains. Chainlink CCIP has limited support but is expected to expand soon. All protocols support message transfer used in our PoC and hence are equally flexible for use cases.

Security and Privacy: The official bridges on public blockchains are secure enough, with proven track records and extensive testing. However, custom configurations make direct comparisons difficult as they can change the definition of security. As a standard practice, regardless of how secure a solution is developed, there should be mechanisms in place to mitigate risk. We have observed that Chainlink CCIP has a secure and risk-mitigation architecture involving a DON for risk management. Axelar Network has risk-limiting measures and plans for failures, while LayerZero leaves it to the user to configure with their rate-limiting feature. Notably, LayerZero can use other bridges' off-chain networks as its DVN if the developer configures it properly and is willing to pay the associated fees, effectively aggregating their security. However, for on-chain smart contracts of all the service providers, although they have been audited, there is still a risk of compromises. No bridges support privacy-preserving schemes.

Ease of Deployment and Development: We conclude that LayerZero is relatively easy to deploy for a private setup without extensive support from the team. In contrast, Axelar Network and Chainlink CCIP deployments are more complex. Axelar Network is perceived as easier to deploy due to its open-source nature, although additional time is required to configure the bridge appropriately. For deployment in industry settings, Axelar Network provides technical support and consultancy, while Chainlink CCIP requires management and deployment by their team due to its proprietary nature. In terms of development, all bridges are equally developer-friendly and support our interoperable token use case. However, Chainlink CCIP maintains a lead in developer friendliness due to its long-standing reputation and industry experience. Axelar Network is on par with Chainlink CCIP by offering extensive sample code, SDK, and simplified gateway interaction. Among the bridges, LayerZero V2 requires more setup steps compared to the others.

In summary, although the bridges are highly secure and battle-tested on public blockchains risks can still exist when using the service providers due to on-chain smart contracts, custom deployment configurations, or human error during development or deployment. We note that reassessment of those risks is necessary following the protocol upgrade. Given the ever-evolving Web3 security landscape, mitigating such risks by using multiple protocols is recommended in the context of low-frequency token movement.

6. PROOF OF CONCEPT

6.1 SCOPE

The scope of this proof of concept is to implement a universal solution to move tokens between blockchains. Due to the dynamic landscape of Web3 and regulatory compliance, the solution must not be tied to any specific bridge implementation. Furthermore, the security of the solution is of utmost importance, given the context of recent exploits in commercial bridges.

Consider a scenario with two blockchains: one called the host/local chain (the blockchain either owned, or controlled by Organization A, a regulated financial institution and token minting authority), and the other called the remote blockchain that will faithfully execute commands from the host chain.

As Organization A expands its tokenization efforts to bring more assets on-chain, the ability to move tokens across multiple chains becomes important as they seek to tap into a wider liquidity pool and transact with other counterparts who are on other chains.

Organization A will play the host/local chain. The system must comply with the regulatory requirement of not having two versions of the same token exist on multiple blockchains, and the tally of the total token supply should be consistent across all participating blockchains. In a real deployment, there can be multiple host and remote chains; however, for the sake of auditability and bookkeeping, we recommend that there should be only one host chain with the authority to send commands and perform minting/burning, while the rest are remote blockchains. The token moving feature works in both directions, burning an amount of tokens on the local blockchain while minting the equivalent amount on the remote chain and vice versa. This moving feature is coupled with additional requirements for compliance and auditability. In summary, the PoC scope comprises the following features:

- Move tokens between blockchains (mint and burn scheme)
- Track the supply and movement of tokens and arbitrary data across chains
- Comply with regulations and audit requirements

The use case of the bridge is to move high-value assets, such as tokenized securities and carbon credits between blockchains, where security is of utmost importance. The nature of these assets means that they will typically be “high value, low volume” transactions. With that in mind, the use case will prioritize security, ahead of other considerations such as throughput, and efficiency.

6.2 DESIGN

Given the scope, use case, and requirements, the team discussed and designed the system to be bridge-agnostic, comply with regulatory requirements, and minimize risks associated with cross-chain bridges. The main idea to mitigate risk is to utilize multiple bridges to verify the same cross-chain message. The on-chain components translate our token movement command into a unique message that is sent by multiple bridges to the same destination. Different service providers will deliver the message to the destination blockchain. The destination smart contract will only execute the message if it is delivered by a configurable threshold of different bridges (e.g., 2 out of 3).

6.2.1 ON-CHAIN COMPONENTS

The on-chain components ensure regulatory compliance, risk minimization, and bridge agnosticism. They provide an interface for user requests to be executed. Aggregating bridges helps in risk minimization. For bridge-agnosticism, we will only use the bridges to send raw data representing the command to burn or mint. Although the bridges usually come with their own standard for pegged or wrapped tokens, utilizing these introduces additional legal and auditing complexities and migration issues. For example, when a bridge is out of service or exploited, its associated tokens will need to be managed. Additionally, the usage of bridge tokens prevents us from using multiple services as the tokens are controlled by their respective service providers. On the other hand, message passing abstracts away any specific usage and leaves the logic to the application level at the business logic layer. We, therefore, design our own contract to manage the minting and burning of tokens, some of which are the commands from a remote blockchain, delivered by different service providers. We divide the components into three layers with a focus on EVM-to-EVM connections.

Bridge Adapters. This layer provides an abstraction level on top of existing bridges. It facilitates a unified interface for interacting with all existing bridges. Adding and removing service providers will be implemented at this layer. Figure 4 shows this design following the inheritance pattern.

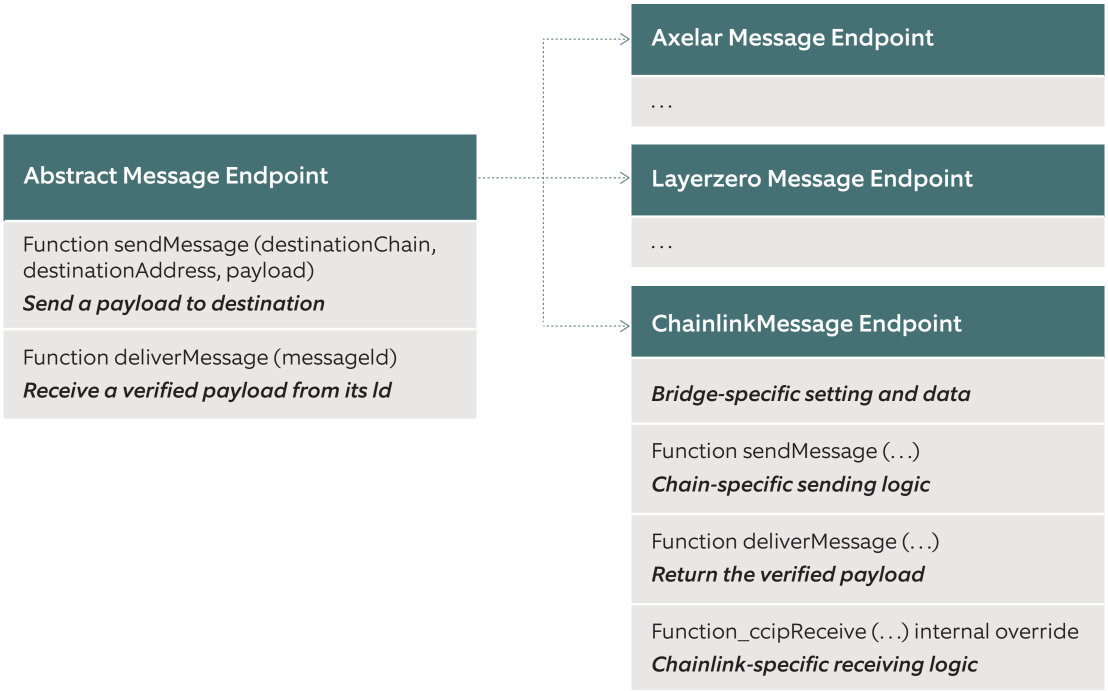

Abstract Message Endpoint
Function sendMessage (destinationChain, destinationAddress, payload)
Send a payload to destination
Function deliverMessage (messageId)
Receive a verified payload from its Id
Axelar Message Endpoint
. . .
Layerzero Message Endpoint
. . .
ChainlinkMessage Endpoint
Bridge-specific setting and data
Function sendMessage (. . .)
Chain-specific sending logic
Function deliverMessage (. . .)
Return the verified payload
Function_ccipReceive (. . .) internal override
Chainlink-specific receiving logic

Figure 4: Bridge adapter's design. The bridge-specific logic is simplified to a sending and a receiving function exposing to other contracts.

To add a new bridge, the developer simply implements the logic of the exposed interface in *Message Endpoint*, using the correct function calls and settings for each specific bridge. For the example in Figure 4, to implement the function *sendMessage* in Chainlink CCIP, we construct the appropriate data structure *EVM2AnyMessage* from the payload and configurations and call the actual *ccipSend* from Chainlink CCIP's router to send the payload. In our design, the same message endpoint can act as both sender and receiver for payload.

Bridge Aggregator. This layer ensures security and minimizes the risks of using a single endpoint. This allows an upper-layer contract to utilize multiple service providers and mitigate the risk of a minority of bridges being exploited or out of service. The bridge aggregator owns all the endpoints and can add/remove endpoints. The main configurable feature of the Bridge Aggregator is the threshold of distinct endpoints in which the message is considered securely verified by a threshold of bridges. It also allows other smart contracts in the business logic layer to access the endpoints collectively in a single function call. Figure 5 shows the design of the bridge aggregator and the interface it exposes.

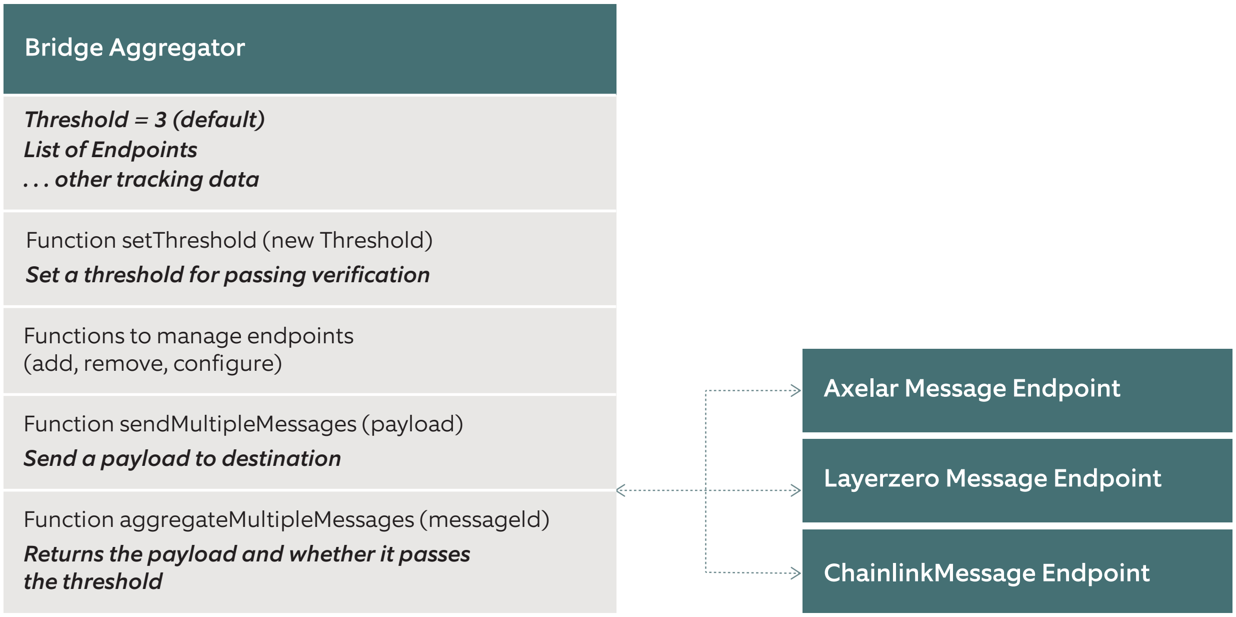

Bridge Aggregator
Threshold = 3 (default)
List of Endpoints
. . . other tracking data
Function setThreshold (new Threshold)
Set a threshold for passing verification
Functions to manage endpoints (add, remove, configure)
Function sendMultipleMessages (payload)
Send a payload to destination
Function aggregateMultipleMessages (messageId)
Returns the payload and whether it passes the threshold
Axelar Message Endpoint
Layerzero Message Endpoint
ChainlinkMessage Endpoint

Figure 5: Bridge aggregator's simplified interface and its interaction with bridge adapters.

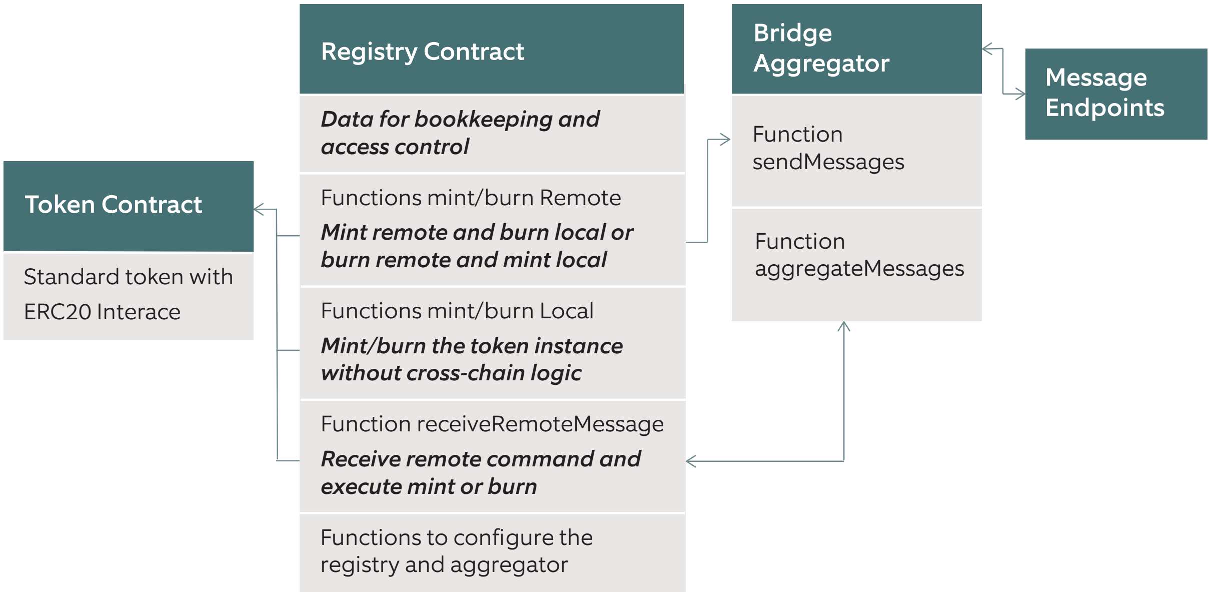

Registry Contract
Data for bookkeeping and access control
Functions mint/burn Remote
Mint remote and burn local or burn remote and mint local
Functions mint/burn Local
Mint/burn the token instance without cross-chain logic
Function receiveRemoteMessage
Receive remote command and execute mint or burn
Functions to configure the registry and aggregator
Token Contract
Standard token with ERC20 Interace
Bridge Aggregator
Function sendMessages
Function aggregateMessages
Message Endpoints

Figure 6: Business logic layer: Token registry and token contract interacting with the bridge aggregator.

Business Logic. Secured by the bridge aggregator and achieved bridge-agnosticism by the adapter layer, this is the final layer to implement arbitrary business logic. In the context of this PoC, the logic implemented includes defining and managing the smart contracts of token for auditability and traceability across historical events. Figure 6 shows the two contracts involved in our business layer design. The registry contract serves as the endpoint for user interaction and controls the token contract. It manages local token minting and burning by interacting with the *token contract* and handles remote minting/burning by interacting with the *bridge aggregator*. Since this design is simplified to suit the proof of concept, additional logic can be incorporated into the registry contract or implemented in separate contracts in the final deployment to meet specific use cases.

There are other smart contracts belonging to different service providers (e.g., gateways, routers, endpoints), with the method to interact with them documented by the respective service provider and reviewed in Section 6. Our bridge adapters will interact with those smart contracts and execute cross-chain message passing seamlessly while providing an abstraction for upper-layer contracts to interact with available service providers.

6.2.2 OFF-CHAIN COMPONENTS

The off-chain components of this solution comprise the components run by us and by the service providers.

Service providers run nodes for validation services and deliver messages from one blockchain to another. They ensure the validity and correctness of cross-chain transactions. These nodes can be run by the host in a private bridge setup or owned and operated by the service provider in a bridge-as-a-service or public setup. Due to their importance, the off-chain nodes are the focus and criteria in the security assessment for bridges. As the landscape of cross-chain bridges continues to evolve and novel exploits emerge over time, we propose using multiple services from different providers to mitigate the risk of any single provider being compromised. This approach can be compared to utilizing the off-chain components of other service providers as DVNs in LayerZero. However, we note that we also isolate the risk of the providers' owned on-chain smart contracts, not only the off-chain components. In other words, a solution like LayerZero, while being more secure off-chain by using multiple DVNs, does not have additional measures to fortify their on-chain smart contracts, rendering off-chain additional security from multiple DVNs less effective. On the other hand, we do not assume any of the service providers' on-chain or off-chain components are completely secure and aggregate their security as a whole to mitigate risks.

For the off-chain components run by us, we provide a web service for users to interact with the on-chain smart contracts. From there, the users' commands will be encoded and sent by the bridge service using their off-chain components.

6.2.3 BLOCKCHAIN AND BRIDGE CHOICES

Regarding the scope of implementation for this PoC, we analyzed different sets of settings. The first one is private testnets. This setup helps to measure performance and resembles the use cases of real-world financial institutions. Ideally, the private testnets can connect with other private testnets either EVM or non-EVM blockchains (Tendermint, Hyperledger Fabric, etc.). They should also be able to connect with public testnets when necessary. However, during the investigation of this project, we faced significant challenges in setting up the appropriate environment. Notably, some services are proprietary and require business commitments before we can test them in a private setup, hence it is not feasible to set up the service on our own. More specifically, although we target private EVM blockchains which most service providers support, Axelar Network and Chainlink CCIP deployment proves difficult and requires either assistance or business commitment from their team. Due to the constraints of connecting to a private blockchain, a private-public setting also faces similar issues. Within the time budget, we investigate the deployment of private setups of bridges to connect two private EVM chains when possible. We were able to deploy LayerZero on these settings and measured its performance. Considering the cost of setting up and experimenting with private setups, we moved towards real public testnets. This allows us to experiment with the real workings of the bridges with their existing infrastructure for testnets. It also reflects the cost and the delay when interacting with the bridge services. For this setting, we select two EVM testnets supported by three aforementioned bridges (Axelar Network, LayerZero, Chainlink CCIP) with low finality time for reduced delay observed by users.

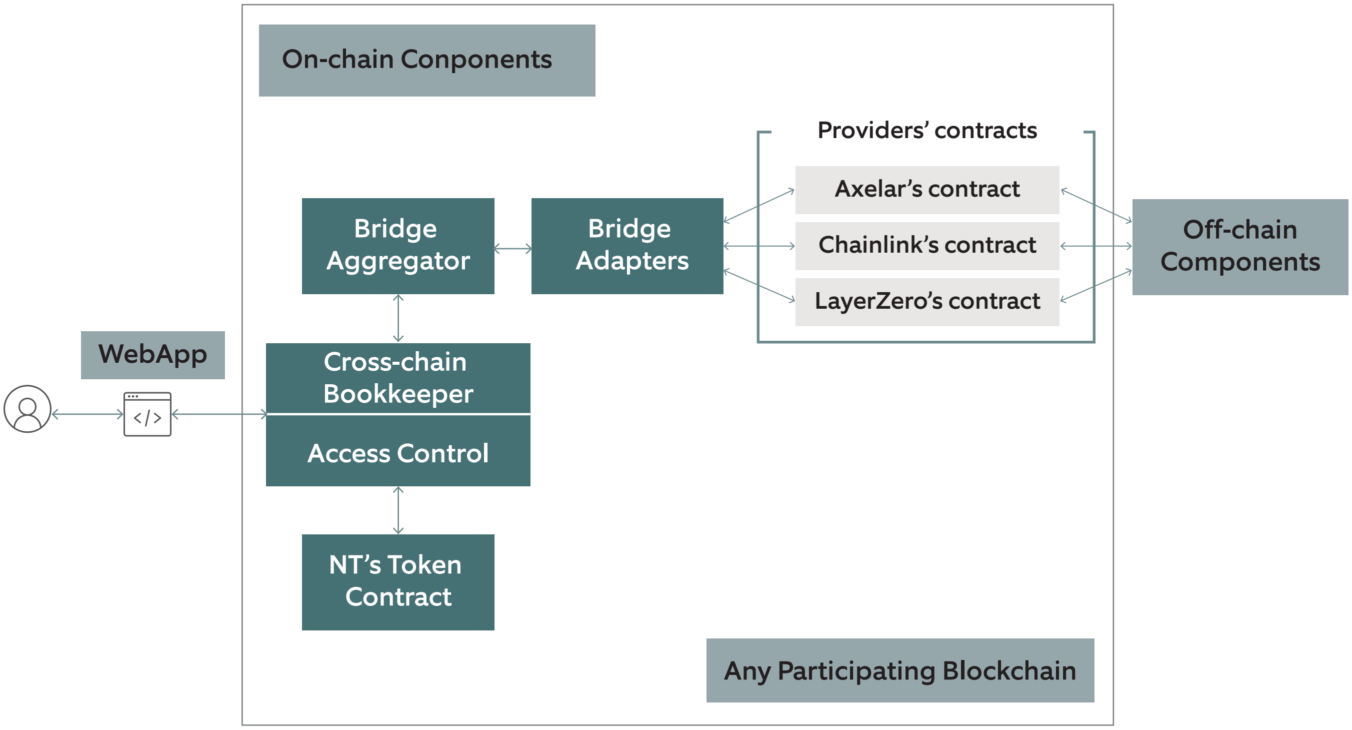


Figure 7: End-to-end implementation of the proof of concept

6.3 IMPLEMENTATION AND DEPLOYMENT

6.3.1 OVERVIEW

Figure 7 shows the end-to-end implementation and deployment of the PoC. Starting with user interaction with our web app, the user may request to view cross-chain history, check their balance, and execute token commands. Our web app then translates these to RPC calls to query blockchain data for viewing or submitting signed transactions to execute commands.

When executing commands, our transactions interact with the business logic contract (namely the *Registry Contract*), which is responsible for bookkeeping, access control, and interacting with the token contracts and bridge aggregator. The transaction invokes the corresponding function based on the user's request (e.g., mint remote). Each remote request has two corresponding actions to be executed on two blockchains with the corresponding account. For example, the mint remote request will trigger minting tokens on the destination chain and burn the same number of tokens on the source blockchains of the same account.

Given the request, the *Registry Contract* then marks the local action as pending (e.g., burning) and encodes the remote action (e.g., minting) using our predefined message format and invokes the Bridge Aggregator to send the message by multiple bridges. The list of configured message endpoints in the Bridge Aggregator will be responsible for invoking the bridge-specific sending logic. After this step, the unique payload ID is recorded both on the on-chain contracts and in our web app. After the transaction is confirmed, the off-chain components of the bridges pick up the transaction payload, verify it, and deliver it to the destination blockchain.

On the destination blockchain, anyone can query the validity of the message and trigger the receiving logic on the destination blockchain by submitting a transaction to the destination *Registry Contract*. The Registry Contract then verifies with the *Bridge Aggregator* on the destination chain that the message has been verified by a sufficient number of bridges. If this check is successful, the Registry Contract decodes the message content and mints/burns the token accordingly. At the same time, our web app acts as a coordinator and also submits the delivery receipt to execute the corresponding logic at the source blockchain.[1] The two actions on the source and the destination blockchain will then be finalized and the corresponding token changes are recorded on both blockchains.

6.3.2 ON-CHAIN COMPONENTS

To reduce implementation complexity, we connect two EVM chains and write all smart contracts in Solidity language. The same set of contracts will be deployed on every participating blockchain. After deployment, they require further transactions to set up the endpoint ownership and link the smart contracts before use.

We use an ownable design pattern for all the contracts, which means only the owner of the contracts can perform critical actions. Those actions are minting and burning tokens, updating configurations of aggregator and endpoints, and updating the admin account. Although the aggregator can be open to the public for sending arbitrary messages, it is better to isolate the usage of our aggregator to only serve the target application with the allowed message format and endpoints. Figure 7 shows the design and connection of smart contracts. The bridge adapters connect to different service providers' contracts and provide a unified interface to be called by the bridge aggregator. The bridge aggregator is controlled by the registry contract, which performs bookkeeping and access control to determine which accounts have the right to perform operations on the on-chain components. It also controls the token contract used cross-chain. Although the token contract is designed to be controlled only by the Registry, we also allow admins to control local minting and burning to simplify test scenarios.

The business layer consists of smart contracts implementing the ERC20 token standard. We note that our solution is universal, hence it can be used for any token standard (e.g. ERC721, ERC1155, ERC1400, ERC404, etc.). The token contract will be deployed and owned by the Registry contract, all cross-chain messages are encoded in bytes data and delivered in an application-agnostic manner.

Table 1: Message format for remote token command

Type	Name	Description
address	receiver	The receiver of this command
uint256	amount	The quantity to be minted/burned remotely
bool	isMint	Indicate minting or burning command
uint256	nonce	Unique nonce to prevent replay or duplicate action

For encoding fungible token operations, Table 1 shows the message format. The message is created by the local contract and decoded and executed by the remote contract. After passing the aggregator, the destination blockchain identifier will be added to the message.

The remote and local contracts will execute the pair of equivalent actions whenever a message is finalized on the remote blockchain. For example, the remote mint command after being delivered will trigger the minting logic at the remote contract and the burning logic at the local contract simultaneously.

[1] We use trusted coordinator for simplicity and low cost. Eliminating the coordinator by sending back the receipt with the bridges incurs more cost and additional delay and deemed not within the scope of this proof of concept.

The remote and local contracts need to execute the two equivalent operations by either communicating via bridge or coordinated by a trusted server. For the PoC, we execute this using a trusted coordinator.

6.3.3 WEB APPLICATION

We developed web applications to allow users to interact with our deployed smart contract on the participating blockchains. The web application consists of a front-end component and a back-end service.

For the front-end, we designed and implemented a user-friendly interface using Next.js. The interface has multiple tabs that allow users to set the cross-chain wallet to track token contracts and token movement functions and to view historical transactions. In the scope of this demo, the blockchain wallet is managed by the back-end service instead of users to simplify the front-end implementation.

We also developed a lightweight back-end service to serve test users. Receiving requests from users, this back-end service also interacts with our deployed smart contract by submitting transactions to blockchains. Since we deployed on a public testnet, we do not run any blockchain nodes but use public RPC endpoints to request public nodes for blockchain interactions. For simplicity, we use basic authentication for access control. The service is written in Python using the Flask web framework.

6.3.4 DEPLOYMENT

Blockchain Settings and Choices. For private blockchains, we use the popular Geth client of Ethereum to set up private blockchains using Clique consensus (Proof of Authority) with a block time of 5 seconds and a block gas limit of 30 million gas units. The nodes run on multiple machines on our cluster and consist of multiple workers. Each of the tested workers has Intel® Xeon® W-1290P CPU and 128GB RAM running Ubuntu 20.04.

For public testnets, we consider the delays it takes for transaction finality and the ease of obtaining test tokens. The pair we chose is the BNB testnet and Avalanche testnet (Avax Fuji). The transaction finality in the BNB testnet is approximately 3 seconds and in Avax Fuji it is around 2 seconds. We also took into account the waiting time for cross-chain transaction confirmation when choosing the pair. One example is that, according to the Axelar Network scan website, the average cross-chain delay for this pair is less than 2 minutes, significantly lower than any pair involving the Ethereum testnet (up to 20 mins). It is because the finality time of Ethereum alone is around 15 minutes [17].

Web Application. Our web service interacts with the blockchain via public RPC endpoints [18]. The web service is deployed on a DigitalOcean droplet instance with 512 MB of RAM and 10 GB of disk space, running Ubuntu 22.04.

6.4 EVALUATION

6.4.1 PUBLIC TESTNETS

We evaluate the effectiveness of this method by observing and analyzing the on-chain transaction records. Table 2 shows the median time and fee used by different service providers. The aggregator fee is the summation of all providers' fees, and the aggregator delay is the longest delay of all the service providers. Furthermore, we require two more transactions for settling the cross-chain operation with the confirmation time equal to a block time (for simplicity, we do not count finality time in this test). We also convert the fee to USD using the exchange rate of US$ 600 for BNB. From this sample data, we can see that the total fee for Axelar Network is the cheapest at the moment while the latency is lowest for LayerZero. We note that those use default settings on the testnets without considering other custom features or configurations [19].

Although the cost and delay of the aggregator are higher than those of individual service providers, the security and risk mitigation it brings are much greater. It isolates all risks related to the on-chain and off-chain components of each service provider. Furthermore, this unified design can swap in and out certain providers and adjust the cost/security trade-off as needed. In scenarios where only one provider is chosen for each blockchain pair, this design also eases integration with different providers to improve coverage, as the coverage of each provider varies. This provides flexibility for decision-makers and planners to choose an appropriate setup with minimal overhead when changing providers.

	Service Fee (BNB)	Time (s)	Gas Fee (BNB)	Total Fee (US$)
Chainlink CCIP	0.001742	82	0.00119	1.76
Axelar Network	0.001185	80	0.00041	0.96
LayerZero V2	0.000905	44	0.00136	1.36
Aggregator	0.003832	82	0.00296	4.08

Table 2: Fee and latency of cross-chain transactions on different service providers.

6.4.2 PRIVATE SETUPS

As mentioned in Section 6.2, due to the nature of each project and the limited time budget, we were only able to deploy LayerZero in a private setup to prepare for future deployments. In public blockchains, especially highly secured chains, the bottleneck mainly arises from the transaction finality, as off-chain validators need to wait for its finality before committing to the destination chain to prevent chain forks and rollbacks, which can invalidate the source transaction. For example, Ethereum's finality time is approximately 15 minutes, significantly longer than any off-chain processes run in the bridge service. In contrast, off-chain latency can be a bottleneck in private setups when the block time is low. This evaluation experiments under the assumption that every transaction is a cross-chain transaction to stress test and find the peak throughput of the bridge.

We deployed LayerZero V2 in private setups with Go-Ethereum (geth) clients running clique consensus (Proof-of-Authority). The block time and gas limit are set at 5 seconds and 30M gas, respectively. To assess the performance, we leveraged our Blockbench V3 framework [20] and the ongoing performance study in SBIP to conduct experiments on this setup. Assuming one-block finality for simplicity, when the system reaches its peak throughput of 18.7 transactions per second, the latency for cross-chain transactions is approximately 65 seconds. This latency is significantly higher than the block time because each cross-chain transaction requires multiple transaction confirmations on the destination blockchains. Furthermore, latency is high in LayerZero when the system sends a large number of transactions, as it enforces sequential confirmation, meaning messages have a unique nonce and must be delivered in order of increasing nonce. Consequently, an additional delay is expected while the off-chain components wait for the previous nonce to be delivered.

7. DISCUSSION

We have observed how different protocols behave differently in the proof of concept. The main differences observable on public testnets are the cost and latency for transaction settlement, which vary widely across each pair of blockchains. Regarding security, each bridge has its own strengths and security checks according to the assessment in Section 6.2. Although some bridges are perceived to be more secure using certain assessment frameworks, the risks associated with running them on public blockchains still exist. Evidence shows that recent exploiters have penetrated supposedly secure bridges, causing reputational damage and financial loss. Therefore, risk mitigation is always necessary. We have incorporated this risk mitigation into our design by leveraging multiple attestation models to prevent any single point of failure from affecting the operation of the interoperable token. The aggregation of different service providers will result in increased costs and longer delays. This approach is considered one of the solutions for assessment in our study.

Given the use case of this study, the cost and reasonable delay do not significantly impact the higher application level. For each transaction, the value transfer is high, and hence security and compliance are of utmost importance. Our design takes this into consideration, producing a sufficiently bridge-agnostic token teleporting application that enables auditability and regulatory compliance. Furthermore, security is enhanced through the use of multiple service providers. Each provider may support different types of blockchains and connections. Therefore, combining and allowing flexibility among providers not only enhances security and mitigates risks but also increases coverage to support more blockchains, allowing the interoperable token to reach a vast and diverse Web3 ecosystem.

8. CONCLUSION

Asset tokenization has undoubtedly brought great opportunities to enable the trading of real-world assets on blockchain platforms. It is now the task of the banking industry and financial institutions to unlock these opportunities further for both primary and secondary market trading across different blockchains. To pursue that mission, this white paper has studied the requirements and design considerations to implement such a solution. We further surveyed state-of-the-art blockchain interoperability approaches and investigated the latest industrial-ready cross-chain bridge protocols. Finally, we implemented a prototype system to demonstrate cross-chain trading of tokenized assets and evaluated the performance of the existing bridges. With the insights and information presented in this white paper, we aim to push the financial and banking industry forward by encouraging the development of more advanced interoperability solutions, fostering collaboration, and driving innovation in cross-chain tokenized asset trading.

9. NEXT STEPS

This white paper is the result of the first collaboration between Northern Trust and the Singapore Blockchain Innovation Programme (SBIP) in Project Opera. The project team has successfully evaluated existing interoperability solutions and developed a proof of concept that demonstrates the feasibility of cross-chain trading of tokenized assets. The SBIP team plans to continue our fruitful collaboration in the second phase of Project Opera by exploring how to apply it to other scenarios within the financial services industry with the help of Northern Trust.

More specifically, before commercial application, we believe that more work needs to be done to analyze the existing blockchain interoperability standards and frameworks such as alignment and integration with ISO20022. We can also build on the bridge assessment framework to investigate its deployment outside of the financial industry. The framework developed is one that could see usage outside of financial services and with the Singapore Blockchain Innovation Programme being a blockchain technology research lab, it is primed to leverage it in our future work.

While the scope of our current proof of concept is limited to ERC-20 tokens, we could also further experiment with interoperability with other digital assets such as Central Bank Digital Currencies (CBDC) and stablecoins. On a longer-term basis, the ideal state is for financial institutions to seamlessly transact and pay using the Delivery versus Payment (DvP) method. Only then, can we fully utilize the potential of digital assets.

Lastly, the Singapore Blockchain Innovation Programme can also continue to explore how the interoperability bridge aggregator can be applied to various financial instruments, using Northern Trust as an example. There are multiple pathways to deploy the solution, ranging from tokenized assets such as tokenized carbon credit, an asset that Northern Trust has brought on-chain, or other traditional financial instruments such as bonds and other forms of securities. We will work together to identify suitable opportunities to further test and experiment before commercialization.

References

1 World Economic Forumn. *A Framework for Blockchain Interoperability*. 2020.
URL: https://www3.weforum.org/docs/WEF_A_Framework_for_Blockchain_Interoperability_2020.pdf

2 Chainlink Education Hub. *What Is a Replay Attack?*
URL: https://chain.link/education-hub/replay-attack

3 The Investopedia Team. *Understanding Double-Spending and How to Prevent Attacks*.
URL: https://www.investopedia.com/terms/d/doublespending.asp

4 The Investopedia Team. *Atomic Swap: Definition and How It Works With Cryptocurrency Trade*.
URL: https://www.investopedia.com/terms/a/atomic-swaps.asp

5 Kunpeng Ren et al. “Interoperability in Blockchain: A Survey”. In: *IEEE Transactions on Knowledge and Data Engineering* 35.12 (2023), pp. 12750–12769.

6 *Chainlink Education Hub: Cross Chain Bridges*.
URL: https://chain.link/education-hub/cross-chain-bridge

7 *Wormhole Protocol*.
URL: https://wormhole.com/

8 *Circle Cross-Chain Transfer Protocol*.
URL: https://www.circle.com/en/cross-chain-transfer-protocol

9 *zkBridge Protocol*.
URL: https://www.zkbridge.com/

10 *An Introduction to the Axelar Network*.
URL: https://www.axelar.network/blog/an-introduction-to-the-axelar-network

11 *Axelar Network: Validator Nodes*.
URL: https://axelarscan.io/validators

12 *Axelar Network: Supported Blockchains*.
URL: https://axelarscan.io/resources/chains

13 *Chainlink CCIP: DVN Addresses*.
URL: https://docs.layerzero.network/v2/developers/evm/technical-reference/dvn-addresses

14 *Chainlink CCIP: Supported Blockchains*.
URL: https://docs.chain.link/ccip/supported-networks/v1_2_0/mainnet

15 *Uniswap Foundation. Bridge Assessment Report*.
URL: https://archive.ph/fd665

16 *Axelar. Axelar Scan*.
URL: https://axelarscan.io/

17 *Single slot finality*.
URL: https://ethereum.org/en/roadmap/single-slot-finality/

18 *Public Node*.
URL: https://publicnode.com/

19 *Axelar. General Message Passing*.
URL: https://docs.axelar.dev/dev/general-message-passing/overview

20 Kunpeng Ren et al. “BBSF: blockchain benchmarking standardized framework”. *In: Proceedings of the 1st Workshop on Verifiable Database Systems*. 2023, pp. 10–18.

Head Office: 50 South La Salle Street, Chicago, Illinois 60603 U.S.A. Incorporated with limited liability as an Illinois corporation under number 0014019. Products and services provided by subsidiaries of Northern Trust Corporation may vary in different markets and are offered in accordance with local regulation. **This material is directed to professional clients (or equivalent) only and is not intended for retail clients and should not be relied upon by any other persons.** This information is provided for informational purposes only and does not constitute marketing material. The contents of this communication should not be construed as a recommendation, solicitation or offer to buy, sell or procure any securities or related financial products or to enter into an investment, service or product agreement in any jurisdiction in which such solicitation is unlawful or to any person to whom it is unlawful. This communication does not constitute investment advice, does not constitute a personal recommendation and has been prepared without regard to the individual financial circumstances, needs or objectives of persons who receive it. Moreover, it neither constitutes an offer to enter into an investment, service or product agreement with the recipient of this document nor the invitation to respond to it by making an offer to enter into an investment, service or product agreement. **For Asia-Pacific markets, this communication is directed to expert, institutional, professional and wholesale clients or investors only and should not be relied upon by retail clients or investors. For legal and regulatory information about our offices and legal entities, visit northerntrust.com/disclosures.** The views, thoughts, and opinions expressed in the text belong solely to the author, and not necessarily to the author's employer, organization, committee or other group or individual.

The following information is provided to comply with local disclosure requirements: The Northern Trust Company, London Branch, Northern Trust Global Investments Limited, Northern Trust Securities LLP and Northern Trust Investor Services Limited, 50 Bank Street, London E14 5NT. Northern Trust Global Services SE, 10 rue du Château d'Eau, L-3364 Leudelange, Grand-Duché de Luxembourg, incorporated with limited liability in Luxembourg at the RCS under number B232281; authorised by the ECB and subject to the prudential supervision of the ECB and the CSSF; Northern Trust Global Services SE UK Branch, UK establishment number BR023423 and UK office at 50 Bank Street, London E14 5NT; Northern Trust Global Services SE Sweden Bankfilial, Ingmar Bergmans gata 4, 1st Floor, 114 34 Stockholm, Sweden, registered with the Swedish Companies Registration Office (Sw. Bolagsverket) with registration number 516405-3786 and the Swedish Financial Supervisory Authority (Sw. Finansinspektionen) with institution number 11654; Northern Trust Global Services SE Netherlands Branch, Viñoly 7th floor, Claude Debussylaan 18 A, 1082 MD Amsterdam; Northern Trust Global Services SE Abu Dhabi Branch, registration Number 000000519 licenced by ADGM under FSRA #160018; Northern Trust Global Services SE Norway Branch, org. no. 925 952 567 (Foretaksregisteret), address Third Floor, Haakon VIIs gate 6 0161 Oslo, is a Norwegian branch of Northern Trust Global Services SE supervised by Finanstilsynet. Northern Trust Global Services SE Leudelange, Luxembourg, Zweigniederlassung Basel is a branch of Northern Trust Global Services SE. The Branch has its registered office at Grosspeter Tower, Grosspeteranlage 29, 4052 Basel, Switzerland, and is authorised and regulated by the Swiss Financial Market Supervisory Authority FINMA. The Northern Trust Company Saudi Arabia, PO Box 7508, Level 20, Kingdom Tower, Al Urubah Road, Olaya District, Riyadh, Kingdom of Saudi Arabia 11214-9597, a Saudi Joint Stock Company – capital 52 million SAR. Regulated and Authorised by the Capital Market Authority License #12163-26 CR 1010366439. Northern Trust (Guernsey) Limited (2651)/Northern Trust Fiduciary Services (Guernsey) Limited (29806)/Northern Trust International Fund Administration Services (Guernsey) Limited (15532) are licensed by the Guernsey Financial Services Commission. Registered Office: Trafalgar Court, Les Banques, St Peter Port, Guernsey GY1 3DA. Northern Trust International Fund Administration Services (Ireland) Limited (160579)/Northern Trust Fiduciary Services (Ireland) Limited (161386), Registered Office: Georges Court, 54-62 Townsend Street, Dublin 2, D02 R156, Ireland.

1020150-AS-BRO-IoTATaSaUFitFI-092424