

Designing payment tokens for safety, integrity, interoperability and usability



Contents

Executive summary	4
 Part I: Research journey and motivation	
Introduction	5
Scope and approach	6
 Part II: Design guidelines and key functionalities	
Design guidelines	14
Key functionalities	16
Mapping of existing token standards to functionalities	21
A new approach of composable standards	24
 Part III: Proposed designs for key areas	
Payments authorization following application of payments control	25
Administrative controls for remediation and recovery actions	28
 Part IV: Post-development findings and future considerations	
Improving payments efficiency while managing risks	31
Conclusion and future exploration	35

Acknowledgements

Authors

Wee Kee Toh

Global Head of Business Architecture for
Digital Payments, Kinexys by J.P. Morgan

Michael Maurer

Software Engineer
MIT Digital Currency Initiative

Emma Landriault

Vice President, Product Manager for Digital
Payments, Kinexys by J.P. Morgan

Ashwanth Samuel

Student Researcher
MIT Digital Currency Initiative

Lillian Wang

Student Researcher
MIT Digital Currency Initiative

Neha Narula

Director
MIT Digital Currency Initiative

Contributors

Umar Farooq

J.P. Morgan

F. Christopher Calabia

MIT Digital Currency Initiative

Naveen Mallela

Kinexys by J.P. Morgan

Sai Valiveti

Kinexys by J.P. Morgan

Muh Hwa Lee

Kinexys by J.P. Morgan

Raunak Rajpuria

Kinexys by J.P. Morgan

Basak Toprak

Kinexys by J.P. Morgan

The MIT Digital Currency Initiative collaborates with external stakeholders to explore practical ideas and design options for digital assets built for the public good. Our work is intended to inform public dialogue about the benefits and risks associated with various technologies and architectures. This report refers to certain products and services offered by J.P. Morgan, as well as products and services offered by other companies, which we draw on as concrete examples. Discussions of these products and services should not be interpreted as endorsements of their design or use or as validation of their performance.

Kinexys by J.P. Morgan is the firm's blockchain business unit focused on building the next generation of financial infrastructure utilizing blockchain technology. Kinexys Digital Payments offers a blockchain-based deposit product that allows clients to make domestic and cross-border payments between Kinexys Digital Payments accounts on a 24/7, near real-time basis. Deposit accounts are recorded on a private, permissioned blockchain ledger, and are referred to as Blockchain Deposit Accounts (BDAs). This is a live deposit product with daily transaction values exceeding \$2 billion.

Executive summary

Massachusetts Institute of Technology Digital Currency Initiative (MIT DCI) and Kinexys Digital Payments at J.P. Morgan collaborated on the research and development of a prototype for payment tokens on EVM-based blockchains. This research focuses on the additional capabilities required to meet compliance and regulatory requirements, identifies gaps in existing token standards and proposes two new sets of capabilities to address these gaps. Produced by a joint research team composed of staff from MIT DCI and Kinexys, this report captures the research journey and findings, serving as a valuable reference for the financial services ecosystem in their work on blockchain and tokenization.

Research journey and motivation: The first part of the report outlines the research journey, detailing the interests and motivations of MIT DCI and Kinexys in payment tokens, as well as the research approach undertaken. Trends in blockchain adoption within the financial services industry highlight the need for bank-issued payment tokens to support financial transactions. However, existing tokens and standards are not designed for financial services, necessitating additional capabilities to meet banks' regulatory compliance requirements.

Design guidelines and key functionalities: The second part of the report serves as a handbook for financial institutions developing payment tokens and token-based products. It provides a set of design guidelines, a list of key functionalities and a mapping of existing token standards that can be implemented to enable these functionalities. In designing the prototype, certain requirements were identified as critically important, forming the basis of many design choices. These have been distilled into the design guidelines, which we hope can be validated and further refined into design principles and best practices for the industry. The list of key functionalities describes capabilities that a payment token should provide and the mapping of token standards highlights existing means of implementing these capabilities. A key insight is the challenge of applying a traditional approach of broad standards to tokens and how a new approach of composable, narrow standards would be more appropriate.

Proposed designs for key areas: The third part of the report describes two key areas that are inadequately covered by existing token standards and presents high-level proposed designs to address these gaps: (1) the application of payment control processes, including the capturing and transmission of required payment-related information and (2) administrative controls for remediation and recovery actions, such as suspending transactions and seizing funds. Detailed technical designs and source code will be published subsequently.

Post-development findings and future considerations: The fourth and final part discusses the findings observed post-prototype development. A key observation is that directly applying current regulations and conventions of traditional payment processing to a token world limits potential efficiency gains. The report explores how risks can be adequately or even better managed in a tokenized environment, enabling changes in payment processing and application of payment controls to fully harness benefits.

Additionally, the report addresses other considerations not explored in detail in this project but relevant for future exploration, particularly regarding the governance of open blockchains, potential risks and concerns and possible mechanisms for managing them.

Part I: Research journey and motivation

Introduction

Public blockchains and the financial services industry have been converging as financial institutions explore blockchain and DeFi concepts in their products and as applications on public blockchains incorporate enhanced governance practices from the regulated world. As general understanding and perception of blockchains improve and as regulated financial institutions grow more comfortable with decentralized technologies, the convergence is shifting towards integration.

Financial institutions are moving beyond single-operator, private, permissioned blockchains to more open blockchains. We use this term to refer to public, permissionless blockchains as well as permissioned blockchains that are open and accessible to a large number of financial institutions. Open permissioned blockchains have been gaining interest in the form of concepts such as Unified Ledgers¹, Finternet² and Global Layer 1.³ This shift towards integration is driven by a growing recognition of the value of shared platforms and the potential for seamless integration of financial services to drive efficiency and liquidity.

Payments are an integral part of any economic exchange, and open blockchains will require new mechanisms for value transfers. Currently, payments on public blockchains are largely fulfilled by stablecoins, which provide relatively stable value by holding high-quality liquid assets, such as government treasury bills, as reserve assets to back the tokens they issue. Stablecoins have a current market capitalization of \$235 billion.⁴ As a point of comparison, deposits held across all commercial banks in the U.S. amount to approximately \$18 trillion.⁵ In order for payments on open blockchains to scale, there are multiple challenges to address. One among them is that regulated financial institutions, like commercial banks, will need to be able to safely enter this market and provide payment tokens on shared blockchain platforms.

Unfortunately, many of the existing stablecoin standards are insufficient for regulated financial institutions as they lack functionality for regulatory compliance. Regulations differ across jurisdictions, and a product compliant with the regulations of one jurisdiction may not necessarily be compliant in another. However, the need to apply certain processes and capture specific information is common. This work addresses this gap, proposing new capabilities for payment controls and administrative controls that facilitate compliance with applicable regulations for token issuers.

First, we define the requirements for regulated financial institutions to issue payment tokens to prioritize safety and integrity. Second, we identify design guidelines, or principles, that should underpin any payment token design. Third, we describe the key functionalities of any payment token system and analyze existing token standards to see how well they accommodate this list of functionalities. We find that there are gaps, and we propose new designs for composable standards to address these gaps. We intend to release a more technical design document and set of reference smart contracts showing how one might implement these standards. Finally, we discuss lessons learned about challenges unique to blockchain systems and discuss ways in which existing regulations might change and adapt to new technology.

While the research focused on a bank-issued payment token, sometimes referred to as deposit tokens or tokenized deposits, the research is applicable to all forms of digital money. This includes commercial bank deposit tokens, central bank digital currencies, stablecoins and other payment instruments issued by non-banks, and the term payment tokens is used to refer to this broader set of digital money.

Part I: Research journey and motivation

Scope and approach

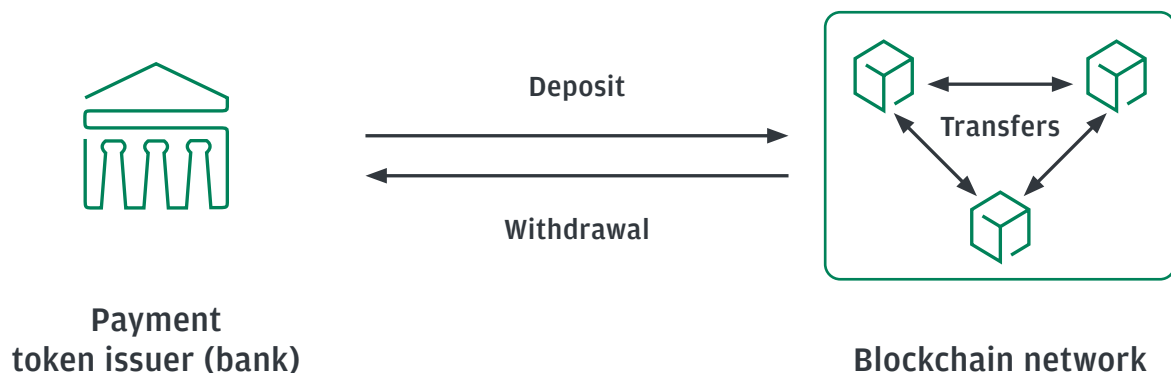
The research focused on additional capabilities required to meet the compliance and regulatory requirements of banks in issuing payment tokens on an open blockchain. Two key areas were identified: (1) the application of payment control processes, including the capturing and transmission of required payments-related information, and (2) administrative controls that allow for remediation and recovery actions, such as suspending transactions and seizing funds.

While there is a strong focus on bank-issued payment tokens, the insights, designs and code developed are applicable to various forms of digital money. Additionally, much of the work is also applicable to other regulated financial products, such as securities tokens, and even real-world assets (RWAs), like property tokens.

Scope





The project was focused on the design and development of a technical prototype that encompasses the entire payment token lifecycle, primarily viewed through the transactions of **Deposit**, **Transfer** and **Withdrawal**.

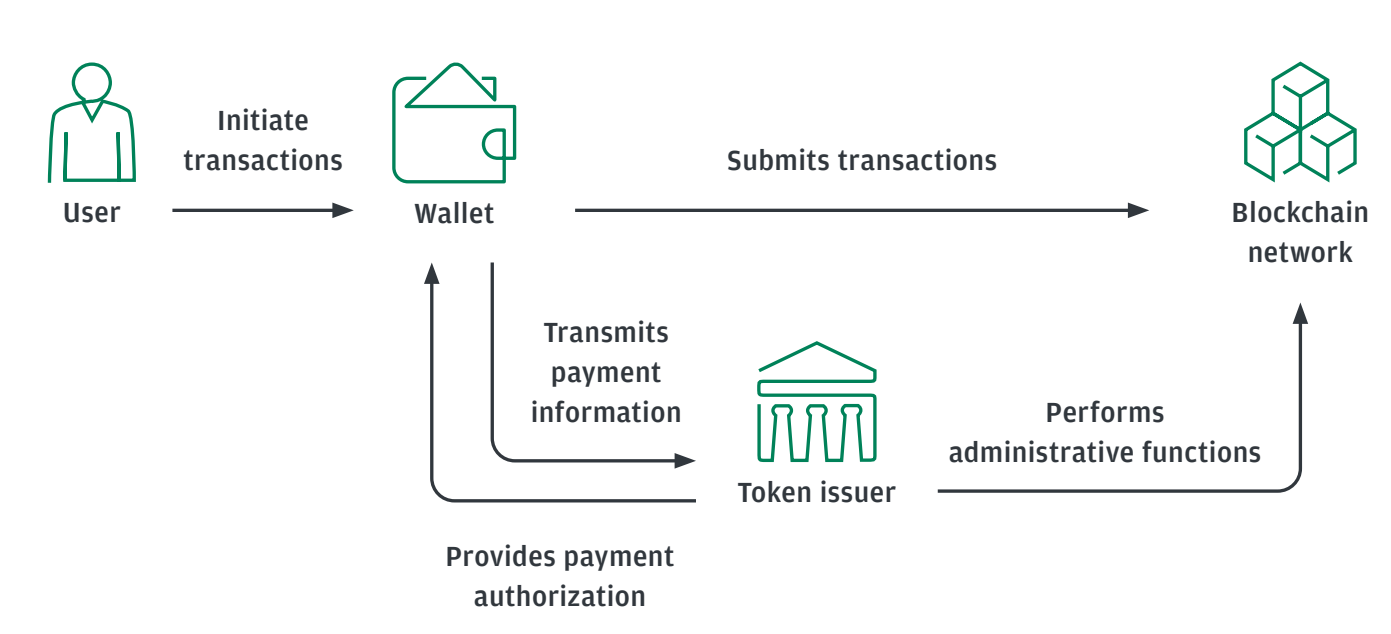
- **Deposit**, often referred to as the on-ramp, involves issuing payment tokens to the user's address.
- **Withdrawal**, the reverse process, is commonly known as the off-ramp, or the redemption of payment tokens.
- **Transfer** pertains to the movement of payment tokens between users and other counterparties.



The prototype incorporated functionalities enabling users to execute these transactions and allowing the issuing bank to implement payment control processes. Additionally, it included administrative functions that the bank can perform. The key roles and their associated activities are as follows:

The relationships and interactions between these roles are illustrated in the accompanying diagram:

Role	Activities
<div> User</div>	<ul style="list-style-type: none">• Holds and transacts payment tokens
<div> Wallet</div>	<ul style="list-style-type: none">• Manages private keys and signs transactions, serving as the interface to the blockchain network for the user
<div> Blockchain network</div>	<ul style="list-style-type: none">• Executes and records transactions of payment tokens and balances
<div> Payment token issuer</div>	<ul style="list-style-type: none">• Issues and manages payment tokens• Applies payment controls on the users and transactions, and provides authorization to users• Performs administrative functions on the blockchain network



These capabilities are primarily enabled through smart contracts developed as part of the prototype. The prototype also explores interactions with external systems, such as blockchain wallets for transfers, banks’ systems for deposits and withdrawals, and between blockchain wallets and banks’ systems for payment authorization. The specifications for these interactions are within the scope of the project, but the external systems themselves are not. In this regard, the interactions were simulated without connectivity to live systems.

In scope	Out of scope
Prototype design Prototype development Proposal for standards	Banks’ internal processing Governance of the underlying blockchain

Prototype design

The requirements, design guidelines, design considerations and proposed design are captured in this report and described in subsequent sections. During design discussions, we evaluated multiple technical design options for different requirements and quickly recognized that there is no one-size-fits-all solution. Instead, there are often trade-offs that need to be evaluated, and design choices are highly dependent on the requirements, which can vary between banks. Key design options, choices and rationale are explained in the report to highlight our thought process and how we arrived at specific designs, which we hope can serve as a framework for other institutions implementing their own payment tokens.

Additionally, we developed recommended principles and practices for building financial applications on open blockchains. By demonstrating how risks can be adequately managed, the recommended practices aim to improve the general level of safety and integrity of financial institutions operating such arrangements and enhance regulatory comfort.

Prototype development

A prototype was developed as part of the project, and the source code will be published under an open-source license to encourage discussion and experimentation. The prototype can also be viewed as a reference implementation of a payment token, showcasing the different token standards that can be implemented together to create a bank-issued payment token that fulfills banks' needs.

Proposal for standards

Standards can take on different forms and serve various purposes. In the technical space, the term commonly refers to interoperability standards, which aim to align on a common language and set of expectations to improve communication across parties. In the regulatory and compliance space, the term generally refers to regulatory standards, which aim to align on a baseline set of principles, practices and guidelines that institutions follow or comply with to meet minimum requirements.

The detailed design and code published will also serve as a starting point for proposal of standards. Two different standards will be proposed in the form of an Ethereum Improvement Project (EIP) for standards—one as an interoperability standard that enables new communication flow between blockchain wallets, banks and blockchains for payments information, and another as a regulatory standard that offers administrative controls that banks can implement to enable greater regulatory comfort.

We sought to ensure that designs were generic and flexible, allowing for easy reuse in different implementations. Even so, the recognition that there is no one-size-fits-all solution has strongly influenced our thinking on token standards and the need for composable standards, where institutions have the flexibility to pick and choose different standards for different components, rather than relying on a single all-encompassing standard.

Out of scope: Banks' internal processing

It is also worth noting that certain related or adjacent aspects are specifically out of scope for the project. The off-chain processing by the banks' internal systems is specifically out of scope, as they are likely to differ greatly across institutions. Additionally, where the interactions are primarily internal to the banks, i.e., between banks' on-chain and off-chain components, they are not a priority from a standards perspective and are out of scope. Internal processes will be referenced where they relate to interactions with other parties or if they impact the design of the payment token smart contracts.

Out of scope: Governance of the underlying blockchain

We assumed that the underlying blockchain, on which payment tokens are deployed, would function as expected. While we examined how the payment token can manage exceptions at the blockchain level, similar to how communication failures or platform failures are managed, we did not delve into the risks and governance of the underlying blockchain itself. These risks are also dependent on the technical design and governance structure of the underlying blockchain. Some of the adjacent risks and concerns are highlighted in the Future exploration section, which also includes potential mechanisms to manage these risks.

Approach

We progressed through the following steps while developing the prototype:

- 1. Define requirements for payment tokens:** Establish the necessary criteria and specifications for payment tokens.
- 2. Research current designs and implementations:** Investigate existing token designs and implementations, including crypto and related products that are live on public blockchains, as well as token standards that have been proposed and are currently in use.
- 3. Define guiding principles:** Establish guiding principles and recommended practices that should be included in the high-level design.
- 4. Integrate existing standards and identify gaps:** Incorporate existing standards, known techniques, and conventions into the prototype, and identify gaps where requirements are not covered by current standards.
- 5. Develop prototype:** Create a prototype based on the identified requirements, guiding principles, standards and other references.

Following the development of the prototype, we also explored additional considerations:

- 6. Identify regulatory constraints and opportunities:** Determine where design was constrained by existing regulations, and where risks could still be adequately managed, and describe how changes in regulations could enable further efficiency gains.
- 7. Identify need for new standards:** Recognize where new standards are required and outline the desired approach for proposing and implementing standards for tokens.
- 8. Identify challenges and concerns:** Identify other challenges and concerns in the deployment of payment tokens on open blockchains, including adjacent areas such as governance of open blockchains and differences in the decentralized landscape that could result in novel risks, as well as potential mechanisms to manage these risks.

Approach to defining requirements

A significant part of the project was dedicated to gathering and defining requirements, and the approach to defining these requirements deserves specific mention. Requirements were reviewed from multiple perspectives to ensure that the objectives of **safety**, **integrity**, **usability** and **interoperability** were well captured and incorporated. As the focus was on control mechanisms, where **safety** and **integrity** are key objectives, we first used a common cybersecurity framework of identify-protect-detect-respond-recover to develop a set of capabilities required for the payment tokens. Next, we reviewed **interoperability**, identifying connectivity points with other ecosystem players, where we attempted to reuse existing standards and proposed new ones where needed. We also identified where requirements needed to be more generic to cater for different ways that ecosystem players could design their products and flows. Finally, we reviewed **usability** and how the flow would work for the end user.

There is sometimes tension across the different objectives, and we often had to iterate across the different perspectives to ensure that the objectives were met and well-optimized.

- **Safety and integrity:** Safety and integrity are two of the most important criteria in our design of the payment tokens. Concerns we aim to address include money laundering and terrorism financing, malicious

activities, unauthorized transactions, including hacks, and other operational risk events such as outages of different components and communication failures. At this stage, we sought to apply the Kinexys enterprise risk management approach to payment tokens on open blockchains, while also factoring in risks specific to implementation on open blockchains.

- **Interoperability:** A key benefit of deploying financial services on open blockchains is the concept of integration, where financial services provided by a broad range of financial institutions can interact seamlessly with each other. From a user perspective, this would mean the ability to use any blockchain wallet to transact with any of their digital assets and payment tokens, and for these tokens to be easily usable and composable with other on-chain services and DeFi protocols. On a technical level, this means interoperability of these components.

Standards are one approach to enabling interoperability by defining how parties will interact with each other. Another common approach is middleware, where a software component bridges and translates communication between different components without needing components to speak a common language. The middleware approach is not ideal for open blockchains due to the wide variety of components and the large number of varied parties operating each component. Defining and adopting standards such that components can communicate directly with each other is therefore critical to enabling interoperability on open blockchains.

It is worthwhile noting that public blockchains have a flourishing ecosystem, even before the entrance of financial institutions. As such, there are already established players and well-adopted standards. Our approach is therefore to reuse existing standards and conventions where possible, and if new standards are indeed required, they should be designed to enable reuse of existing capabilities. For example, when proposing a new model of communication between the wallet and bank, we use a data structure for verifiable credentials to store authorizations, as many blockchain wallets already have the capability of storing and using verifiable credentials for other on-chain transactions.

- **Usability:** One recurring theme with the slow adoption of decentralized finance is the high mental barrier to entry. Using products on public blockchains often requires an understanding of concepts such as key management and gas fees, and learning to engage with these systems with good security practices is a challenge in itself. These complexities mean that adoption today has been largely confined to the technologically savvy.

On the traditional finance side, services are often accessed through financial institutions' applications and web portals. Where multiple services need to be accessed, such as making a securities trade on a securities exchange platform and making the corresponding payments on a bank platform, users will need to perform multiple steps on multiple platforms, often needing to replicate information across the platforms for these transactions to match. The number of apps and steps is further increased with the need for second-factor authentication (2FA). These complexities could potentially be simplified through common user interfaces such as a single blockchain wallet.

With different challenges faced on public blockchains and in traditional finance, there is an opportunity to implement best practices from both sides.

Managing tensions and optimizing across objectives

The various objectives of safety, integrity, usability and interoperability are sometimes in contention, and it is essential to strike an optimal balance among them. A common example is the tension between usability and safety, where making transactions easier to perform may increase the risks of phishing and hacks.

Another area of contention is interoperability, specifically in the adoption of existing standards. Many standards that Ethereum users are accustomed to today were not designed with regulated financial institutions in mind and may not be suitable for new entrants to the space. An example is the transmission of payments information, where there are simply no adequate existing standards or conventions. In such cases, it may be necessary to propose entirely new standards.

Finally, it is also worth considering another perspective on designing for safety and integrity, which is adherence to applicable financial regulations. In most cases, regulations and safety objectives are aligned—regulations are put in place to ensure that all financial institutions follow a baseline set of requirements, and large global banks like J.P. Morgan typically impose even stricter requirements on themselves.

Regulations are designed to be enduring, which also means that regulations applied today may have been enacted decades ago. Such regulations were designed to manage risks in specific ways, possibly due to limitations of legacy technologies, while modern technology might have enabled more effective methods. From a product perspective, banks will have to comply with applicable financial regulations, even if that means operating in a less efficient manner. The section on *Improving Payments Efficiency while Managing Risks* will explore these trade-offs and suggest possible changes in regulations and practices to enable more efficient payments while still safeguarding the safety and integrity of the financial system.

A structured approach to defining requirements for safety and integrity

We found the NIST cybersecurity framework⁶ of identify-protect-detect-respond-recover to be a useful structured model for defining requirements for control mechanisms, where safety and integrity are key objectives. Key risk factors and scenarios are identified to determine the mechanisms needed to manage them. We also observed that protection (with prevention being the primary focus) and detection mechanisms tend to be scenario-specific, while response and recovery mechanisms are more common across scenarios. It is important to note that the mechanisms identified are a combination of features that need to be built, as well as recommended practices from an operational perspective.

Protection / Prevention		Detection
Compromised access	Segregated roles and access controls will limit the impact of compromised accounts and keys. For example, an account or key used by operations for manual approvals (e.g., approval of flagged transactions) should not be used for other administrative functions like upgrading smart contracts.	The use of privileged administrative functions should be minimized so that any usage of such functions can be easily and clearly identified. This is also beneficial from a transparency perspective, as the use of privileged functions is obvious to participants on the network
Anti-money laundering	Transactions can be performed on the blockchain only with the authorization of banks. Payment information is captured and transmitted to the bank for screening and application of control processes. The transaction can be processed once authorization is provided.	Transactions on the network are monitored to detect anomalous groups of transactions.
Transacting across networks	When debits and credits occur across networks, such as the deposit and withdrawal of payment tokens, debits are always performed before credits to ensure that failures (e.g., communication issues) do not result in the loss of funds.	Debits and credits across networks are performed using a two-phase orchestration mechanism with an intermediate account. With this model, any failure in the orchestration will result in funds being left in the intermediate account, allowing for them to be detected.
Response		Recovery
All	Accounts involved in anomalous transactions can be suspended to prevent further flow of funds while investigations are conducted. In cases of mass anomalous transactions or potential catastrophic failures, transactions can be suspended on a global level, a mechanism commonly referred to as a kill switch. This mechanism is also useful in the event of a resolution or liquidation.	Following the results of investigations, funds from suspended accounts may be seized and handed over to law enforcement agencies.

Part II: Design guidelines and key functionalities

Design guidelines

In designing the prototype, we identified certain requirements as critically important, forming the basis of many of our design choices. We have distilled these requirements and considerations into a list of design guidelines that informed our design and development of the prototype. We hope that these guidelines, once validated by other parties, can be refined into a set of design principles and best practices for the industry.

These guidelines aim to balance the objectives of safety, integrity, usability and interoperability while addressing the unique challenges of deploying financial services on open blockchains:

- 1. Minimize on-chain data storage and processing:** Storing and processing data on the blockchain is computationally costly. On private networks, this affects performance and scalability. On open networks, it translates directly to costs in the form of gas fees, which can be significant on certain networks.
- 2. Avoid storing personally identifiable information (PII) on-chain:** On-chain data is distributed across all nodes and accessible by network participants. On public blockchains, data is accessible by everyone. Therefore, PII must not be stored on-chain to comply with privacy and bank secrecy laws. We take an even stricter approach by not storing encrypted forms of such data on-chain, due to the risk of harvest-nowdecrypt-later⁸ attacks. Consequently, data containing PII must be communicated off-chain, necessitating the ability to link off-chain communications with on-chain processing.
- 3. Complete payment transfers within a single blockchain transaction:** The payment token should be composable with other smart contracts to support key use cases, such as the atomic exchange of a payment token for a digital asset or programmable payments where payments are initiated upon business process completion. To allow composability, a payment transfer must start and complete within a single blockchain transaction. Traditional payment systems involve multiple steps and internal state changes, as processing is performed across multiple systems, which breaks composability and atomicity. Banks will need to rethink transaction processing flows, as traditional payment flows cannot be directly transposed to payment tokens.
- 4. Segregate roles for privileged administrative functions at smart contracts:** In enterprise systems, different roles are typically performed by different users. For example, software upgrades are handled by the technology team, manual transaction approvals are handled by the operations team and the configuration of product parameters is handled by the product team. Highly privileged and risky functions often require multiple approvals. While these roles are segregated internally, enterprises sometimes apply access control only within the organization, with all external transactions signed using a single key. For security reasons, it is important to segregate these roles at the smart contract level as well, using different keys for different roles and functions. This approach limits the impact in case of compromises, as an operational user role would not have the ability to make changes to smart contracts. It also reduces the likelihood of compromises, as the roles with authority for smart contract changes are assigned to fewer people and used less frequently than an operational user role.

- 5. Ensure observability of privileged administrative functions:** Privileged administrative functions, such as moving funds out of an account, are important because they can be used to help a legitimate user recover their funds or to seize funds when directed by law enforcement. However, the existence of such functions may raise concerns⁷ about potential misuse. The use of such functions, hence, must be observable and not obfuscated to ensure trust and transparency. Minimizing their use is important to prevent obfuscation. For example, in the redemption of tokens, a two-step process could be employed where funds are transferred to a specific redemption address before being removed from circulation (burned), or they could be directly burned from a user's address. While the second option is simpler, it could also obscure transactions involving the seizure of funds.
- 6. Initiate transactions from the system where funds are held:** Transactions across different systems, such as deposits or withdrawals, should be initiated from the system where the funds are held, ensuring they are processed as push rather than pull payments. Push payments are generally preferred due to the need for debit authorizations with pull payments. Also, such authorizations are typically set to a higher amount and frequency than necessary to prevent failures or the need to amend authorizations with every transaction. Therefore, deposit transactions should be initiated from the banks' systems, while withdrawal transactions should be initiated from the blockchain wallets.
- 7. Fail safely and obviously:** Where there is potential for exceptions and failures, systems should be designed to fail safely (ensuring no loss of funds) and obviously (making failures easily detectable for recovery operations). Communication failure is a common scenario that must be addressed, particularly for deposit and withdrawal transactions, where debits and credits occur across the banks' systems and payment token addresses. To fail safely, credits are never performed before debits are confirmed, ensuring that failures do not result in the loss of funds. For deposits, funds are first debited from bank accounts before being credited to payment token addresses. To fail obviously, debits and credits across networks are performed using a two-phase orchestration mechanism with intermediate accounts. In this model, any failure in the orchestration will result in funds being left in the intermediate accounts, allowing for the timely detection of such failures through monitoring of these accounts.
- 8. Make participation easy for users:** Transaction flows should minimize user touchpoints, enabling users to perform all transaction steps within the same application with minimal back-and-forth interactions. Technical requirements for users should be minimized, such as using request-response communication between blockchain wallets and banks instead of callbacks. Additionally, there should be minimal deviations in the form, function and usage of on-chain payment token products compared to existing bank products. Transaction flows should reference existing bank products and public blockchain flows that are familiar to users.

Part II: Design guidelines and key functionalities

Key functionalities

The key functionalities of a payment token are described below. As the intention is to develop a blueprint for payment tokens, the list includes high-level functionalities that are likely to be required. However, some functionalities may not be required or be relevant to certain forms of payment tokens. For example, stablecoins today are different from deposit products in that they do not provide interest and are not likely to implement interest functionalities.

Ledger and accounting

1. Accounting ledger: Defines how balances and updates are recorded

User balances can be recorded in different forms on the blockchain. One commonly used model is the account model, which records users or wallet addresses and the balances associated with each address.

Another model is the token model, which records tokens and their ownership. A physical analogy is cash notes, where each note has a fixed denomination and can be uniquely identified by its serial number. A variation of the token model is the Unspent Transaction Output (UTXO) model, where a coin can be split into sub-denominations or coins can be aggregated, with different parts owned by different parties. The UTXO model is best known for its use in Bitcoin today. While the UTXO model is not commonly used for smart contract-based tokens, it offers potential privacy and transparency benefits that could be explored in future designs.

Enabling higher-level programmable logic is easier when the state is stored in an account-based form. This is because controls, privileges and features are typically applied on a per-entity basis or with respect to a global state. While it is possible to achieve this in a UTXO-based system with external coordination and with off-chain accounts, it is more complex, making the account-based system the preferred model.

2. Remuneration: Calculation, accrual, and crediting of returns

Holders of payment tokens are likely to expect to receive returns based on their holdings, similar to how they receive interest for their deposits with banks. While we generally think of interest as the main form of remuneration, token issuers may characterize the returns in other forms. The payment token product will need to provide capabilities to calculate, accrue and credit returns to the client. Interest is typically calculated and accrued daily for traditional bank deposits today. With a 24/7 product, returns could be accrued on an even tighter intra-day, perhaps even real-time, basis. This capability will likely be performed through a combination of off- and on-chain processes.

Transaction management

3. Deposit & withdrawal: Allows movement of funds between bank accounts and blockchain addresses

Users will need to move funds between their bank accounts and their addresses for payment tokens on an open blockchain. We refer to these transactions as deposits and withdrawals, which are commonly associated with a user depositing or withdrawing cash from their bank accounts. However, in this context, deposits refer to the issuance of payment tokens into the user's address, and withdrawals refer to the redemption of payment tokens. These operations are also commonly referred to as the on-ramp and off-ramp of funds.

4. Transfers: Allows users to send and receive tokens

The transfer of tokens is a basic capability that allows users to send and receive tokens. While the capability itself is straightforward, the requirements and implementation differ, particularly regarding the information included with the transfer function. In traditional payments, a payment request⁹ is typically initiated through a standard ISO20022 Payments Initiation request using the pain.001 message format, which contains hundreds of data fields, with only a fraction being critical to transaction processing. In comparison, the transfer function in ERC-20, one of the most commonly used token standards for payments on public blockchains today, uses only three pieces of information: the sender's address, the recipient's address and the amount. It is likely that transfers will require additional data fields beyond these three to fulfil the requirements of banks.

From a technical perspective, the information captured in the transfer function depends on the accounting ledger model used to record and update balances. Recording in an account form, such as that used in ERC-20, requires capturing the amount to be transferred, while recording in a UTXO form requires capturing the identifiers of the UTXO tokens to be transferred.

5. Delegated transfers: Allows transfers by authorized third parties, including other smart contracts

In addition to direct transfers by the owners of an address, there are scenarios where the authority to transact from an account is delegated to a third party. This third-party authorization is often used in financial services to grant access to other parties, such as for automated bill payments through debit authorizations, and for broker-dealers to initiate payments for securities settlement on behalf of a client.

In the blockchain world, authorization is also needed to allow interactions with other smart contracts. For example, in payments automation, a business smart contract may initiate a payment upon fulfillment of certain conditions. In such a scenario, the owner of an account must grant authorization to the business smart contract to initiate a payment on their behalf.

This feature is built into ERC-20, and the same approach and design will be retained. As mentioned in the Transfers functionality, this type of transfer may also require additional data to be included in the transaction.

6. Alias-based transfers: Allows transfers to easy-to-remember identifiers

Transfers in traditional payment systems typically require knowing the recipient's bank and account number. Addressing schemes have also been established to allow transfers using known identifiers, such as a phone number or email address, which are then mapped to the underlying account numbers. A similar capability is important on open blockchains, particularly because wallet addresses, the equivalent of account numbers, are long strings of hexadecimal characters that are difficult to remember.

Payments control and permissions

7. Permissioned transfer: Allows application of rules to approve or deny transfers

Permissioned transfers refer to the ability of administrators to approve or deny transactions based on specific compliance rules. Compliance logic can be embedded in the smart contract, depending on regulatory requirements, to ensure that all users systematically and automatically adhere to relevant frameworks. For instance, certain regions may require that token transfers only occur between preapproved KYC clients; this requirement can be incorporated into the system to ensure all users meet specific policy criteria.

There are different mechanisms to implement these rules, including codifying rules in the smart contracts, checking against approve/deny lists or validating transaction authorization attached to a transfer request before processing it.

8. Collection and communication of payments information: Allows capturing and sharing of payments information with banks

Banks will need to collect and process payment information to apply payment control processes. This is likely to include sanctions screening, anti-money laundering processes and potentially other analyses such as fraud checks. Where these processes are performed off-chain, mechanisms must be in place to seamlessly collect the necessary information at the point of transaction initiation, i.e., through the user's blockchain wallet. This information must then be communicated to the bank, and the subsequent authorization must be received and attached to the on-chain transfer.

9. Federated access control: Allows trusted entities to enforce token access policies

In the traditional correspondent banking world, certain control processes are applied by respondent banks rather than correspondent banks. Similarly, there can be a delegation of control, where control processes are applied by other parties. Mechanisms must be in place to allow for such delegation and to validate that controls have been applied by the relevant parties before allowing a transaction to be processed. It is also important that this federated access can be managed and revoked in a timely manner.

Governance and administrative controls (privileged functions)

10. Account recovery and key rotation: Allows users to update keys or recover accounts

Blockchain transactions are managed through public and private keys. Public keys are externally facing addresses that interact with other public addresses on the blockchain. Private keys are intended to be known only by the original keyholder, and losing these private keys can result in account loss or compromise. Some early proponents of blockchain technology praised private key features as critical and essential to the ethos of the system they supported, particularly in a decentralized ecosystem. However, managing private keys has proven to be a significant barrier to mainstream adoption and use of public blockchains due to security considerations, especially concerning the inability to recover lost keys.

With banking products, where clients or users are known, banks retain sufficient identifying information and have the ability to validate users and assist in the recovery of accounts and/or funds held in these accounts, similar to resetting passwords. While a lost private key cannot be recovered, there are mechanisms to change or rotate the keys used to access a wallet address, allowing users to recover their account if they lose their keys. ERC-4337 is the most common method of enabling key rotation.¹⁰ This is something we aim to preserve in this system so that even less technologically savvy users can experience a similar level of comfort with this technology as they do with their more traditional digital banking infrastructure. As with all privileged administrative functions, ensuring observability is an important design guideline to be followed.

11. Account suspension: Allows suspension of accounts and freezing of tokens

To comply with regulatory frameworks and legitimate requests from law enforcement agencies, banks will need to develop capabilities for suspending accounts and freezing tokens. Additionally, the scenarios for using such features should be clearly outlined, and there should be strict adherence to the outlined procedures to ensure transparency and trust from users and regulatory agencies.

12. Asset seizure: Allows seizure of tokens under specific conditions

In addition to suspending accounts, assets and funds may need to be seized and recovered, necessitating capabilities for administrators to conduct transfers on suspended accounts. Including this capability is important to ensure compliance with regulatory frameworks that often require this feature.

13. Global pause: Allows halting of all token transactions

There may be a need for a global suspension or halting of all token transactions in crisis scenarios. This last-resort capability is sometimes referred to as a circuit breaker or a kill switch. This capability is also important for compliance with requirements for orderly resolution. If an order for resolution is issued, there will be a need to stop further transactions while the institution is taken into receivership or conservatorship. Such orders were previously issued after business hours, when accounting books were closed for the day and transactions were no longer processed. With 24/7 transactions, it is important that any such orders are followed and instituted promptly, as transaction processing continues even while the order is being issued.

Blockchain-specific considerations

14. Smart contract upgrades: Allows upgrading of token logic

As with all software, there are constant improvements, issues and bugs that are identified and resolved. Therefore, it is important that the software for tokens, in the form of smart contracts, can be upgraded. At first glance, this may seem inconsistent with the tamper-resistant property of blockchains.

Records stored on blockchains are tamper-resistant because of the use of cryptographic hash functions. A deployed smart contract and the data records it accepts and stores cannot be changed without detection. However, this does not mean that balances cannot be updated. Instead, updating balances is achieved by appending a new record of the transaction. Similarly, smart contracts can be upgraded by deploying a new smart contract with updated logic and updating the pointer to this new logic. There are various technical models and patterns for enabling smart contract upgrades.

A proxy contract model is used to allow the controller to upgrade the smart contract implementation, even after initial deployment. When using this pattern, to preserve user trust, payment token issuers must reliably indicate when an upgrade occurs and specify the purpose of the upgrade.

15. Gasless transactions: Allows Third Parties to Pay Gas Fees on Behalf of Users

On public blockchains, validators and miners are incentivized to process transactions through the payment of gas fees. These fees are typically determined and paid by the initiator of the transaction. Fees vary based on computational usage, i.e., how much computation and storage is used, and price, i.e., surges in pricing during times of high volume and usage. If transactions are submitted with gas fees that are too low, they may not be picked up for processing. Deciding the appropriate gas fees for each transaction is complex for users and requires them to hold native crypto tokens. Gasless transactions allow for fees to be paid by a third party, simplifying transaction processing for users by abstracting the complexities of holding and managing native tokens and managing gas fees.

Part II: Design guidelines and key functionalities

Mapping of existing token standards to functionalities

We took a deliberate effort to review existing token standards and reuse them whenever possible. The subsequent table maps relevant token standards against the list of functionalities described in the previous section. We included a short summary for each of the highlighted token standards, as well as our thoughts on which ones best fulfil the listed requirements.

Category	Feature	Description	Relevant designs (bold = best fit, <i>italics</i> = other options)
Ledger and accounting	1. Accounting ledger	Defines how balances and updates are recorded.	<ul style="list-style-type: none"> • ERC-20: account-based • ERC-721: fixed denomination • e-Cash: fixed denomination tokens • No prominent standard that records in UTXO form on EVM
	2. Remuneration	The calculation, accrual, and crediting of returns.	<ul style="list-style-type: none"> • DeFi implementations exist to calculate interest, but no existing standard incorporates this capability in a standardized, widely adopted way
Transaction management	3. Deposits and withdrawals	Allows movement of funds between bank accounts and blockchain addresses.	<ul style="list-style-type: none"> • Typically performed internally by a single party, i.e. token issuer, and may not be relevant from a standards perspective
	4. Transfers	Allows users to send and receive tokens.	<ul style="list-style-type: none"> • ERC-20: Provides the standard set of functions (transfer, TransferFrom, approve) with account-based ledgers that lay the foundation for all other standards • <i>ERC-1155: Supports both single and batch transfers for multiple token types</i>
	5. Delegated transfers	Allows transfers by authorized third parties, including other smart contracts.	<ul style="list-style-type: none"> • ERC-20: Supports delegated transfers via the approve/TransferFrom mechanism • <i>ERC-1400: Provides delegated transfers with compliance checks</i> • <i>ERC-1155: Introduces operator approvals for delegated transfers</i> • <i>ERC-3643: Inherits the ERC-20 model with added compliance verification for each transfer</i> • <i>ERC-4337: Facilitates delegated transfers through meta-transactions with account abstraction</i>

Category	Feature	Description	Relevant designs (bold = best fit, italics = other options)
	6. Alias-based transfers	Allows transfers to easy-to-remember identifiers, rather than long alphanumeric wallet addresses.	<ul style="list-style-type: none"> • ERC-3643: Supports on-chain identities which allow users to transact using aliases (ONCHAINID) • ERC-4337: Most used account-abstraction standard
Payments control and permissions	7. Permissioned transfers	Allows application of rules to approve or deny transfers of tokens.	<ul style="list-style-type: none"> • <i>ERC-1400</i>: Combines transfer restrictions with regulatory compliance to enforce permissioned transfers • <i>ERC-3643</i>: Enforces permissioned transfers via on-chain identity verification and whitelist controls • <i>ERC-6997</i>: <i>ERC-721 with follow-on transaction validation step</i>
	8. Collection and communication of payment information	Allows the capturing and sharing of payment information with banks.	<ul style="list-style-type: none"> • No prominent standard meets this feature requirement as we need it • <i>ERC-735</i> • <i>ERC-3643</i>: Enables specification of off-chain rules, through an on-chain registry.
	9. Federated access control	Allows trusted entities to enforce token access policies.	<ul style="list-style-type: none"> • <i>ERC-1400</i>: Can implement access control as part of its compliance framework • <i>ERC-3643</i>: Offers built-in mechanisms to allow multiple trusted entities to manage and enforce access policies over token transfers • <i>ERC-6617</i>: Bit-based permissioning scheme • <i>EIP-7820</i>: Defines a standard access control registry
Governance and administrative controls (privileged functions)	10. Account recovery and key rotation	Allows users to update keys or recover accounts in case of key loss.	<ul style="list-style-type: none"> • <i>ERC-1400</i>: Offers partial key recovery support, but not comprehensive • <i>ERC-3643</i>: Integrates on-chain identity, enabling key recovery and rotation mechanisms for regulated assets
	11. Account suspension	Allows suspension of accounts and freezing of tokens.	<ul style="list-style-type: none"> • <i>ERC-1400</i>: Provides control mechanisms via modules to suspend transfers • <i>ERC-1404</i>: Builds on 1400 and can restrict/suspend/freeze based on specified conditions • <i>ERC-3643</i>: Built-in compliance features that allow administrators to freeze and suspend transfers during regulatory or crisis events
	12. Asset seizure	Allows seizure of tokens under specific conditions.	<ul style="list-style-type: none"> • No prominent standard meets this feature requirement as we need it

Category	Feature	Description	Relevant designs (bold = best fit, italics = other options)
	13. Global pause	Allows halting of all token transactions.	<ul style="list-style-type: none"> • ERC-3643: Provides functions to pause or suspend transfers across the token system in case of emergency • <i>Pausable: A smart contract module, by OpenZeppelin, that allows a smart contract to temporarily halt critical functions</i>
Blockchain-specific considerations	14. Smart contract upgrades	Allows upgrading of token logic.	<ul style="list-style-type: none"> • ERC-1882: Provides a minimal, efficient upgrade path through UUPS proxies • <i>ERC-2535: Known as the Diamond Standard and is specifically engineered to allow modular upgrades</i> • <i>ERC-3643: Contract logic can be updated without disrupting token balances</i>
	15. Gasless transactions	Allows third parties to cover gas fees on behalf of users.	<ul style="list-style-type: none"> • ERC-3009: Permits a third party to submit authorized transactions and pay associated gas fees, enabling gasless transactions • <i>ERC-4337: Indirectly enables gasless transactions via bundled transaction objects that decentralized entities can sponsor gas fees for</i>

Part II: Design guidelines and key functionalities

A new approach of composable standards

In the previous section, we reviewed existing token standards and mapped them against a list of functionalities and requirements for payment tokens. Analysis of these standards highlighted concerns with how standards are currently being proposed and adopted, and how the process could be further improved. For each requirement, there are often multiple token standards. In some cases, different design options exist, and various token standards exist (rightfully) to offer different options for implementation. For example, ERC-20 is used for fungible tokens, and ERC-721 for non-fungible tokens.

In most other cases, there are multiple duplicative standards supporting similar implementation models. These situations typically arise because a current standard lacks support for certain requirements, leading to the proposal of a new standard to address the new requirements while replicating similar existing standards for other needs. This results in multiple overlapping standards, making it challenging to drive convergence.

In the payments space, ISO 20022 is the most widely used standard. It is an example of a broad standard, encompassing more than 750 messages.⁹ Defining it was a massive industry effort that spanned multiple years, supported by hundreds of organizations and a strong governance structure to drive consensus across the financial ecosystem. It is unlikely that such efforts can be replicated for token standards in the near term, especially as the industry is still relatively nascent and lacks an existing governance body or structure to drive consensus across diverse players.

Instead, we believe that the approach for standards in the token space should involve composable standards, where a token implements multiple standards for different parts or components. For this approach to succeed, the standards must not conflict, and there should be minimal overlap between them. Standards should therefore be designed to be narrow in scope and componentized in a way that allows them to be easily composed with other standards. This is the approach we are taking with the two standards we are proposing in the next section.

The decision to propose new standards was not taken lightly. Through discussions with the ecosystem, including an Interoperability Workshop for Financial Services¹¹ organized by the Linux Foundation Decentralized Trust and co-hosted by Kinexys, we observed general alignment on the need for standards but no clear direction on how to achieve it. In fact, there is general apprehension about the proliferation of standards, where specifications developed by individual parties are framed as standards even when adoption is limited. The benefits of standards are realized through convergence and alignment, and there is a need to move beyond “your” standards versus “my” standards, towards the convergence of “our” standards.

We will also be publishing our reference implementation of a payment token. This reference implementation should not be viewed as a standard, but rather as a blueprint of the different token standards that can be implemented together to create a bank-issued payment token that fulfills banks’ needs. While our reference implementation is currently one-dimensional, as it covers one view of the requirements, we hope that as more implementations take shape, the blueprint will encompass multiple design options for each requirement, enabling truly composable standards. We envision a future where users can select between different standards based on their requirements, and where their selection of different standards can be easily implemented in a coherent and harmonized manner.

Part III: Proposed designs for key areas

Payments authorization following application of payments control

The next two sections describe two key areas inadequately covered by existing token standards and present high-level proposed designs to address these gaps. This section explores the application of payment control processes, including the capturing and transmission of required payment-related information, before a payment authorization is provided.

Background and design considerations

Banks are required to apply control processes, such as AML/CFT measures, to payments, and payment tokens are no exception. In traditional payments, a sender interacts directly with their bank, and processing is handled by a set of the bank's systems. With payment tokens on open blockchains, significant changes necessitate a shift in the flow of information and processing. Several design guidelines are relevant here:

- Minimize on-chain data storage and processing
- Complete payment transfers within a single blockchain transaction
- Avoid storing personally identifiable information (PII) on-chain

Process and information flow: While some payment control processes, such as sanction screening, which involves matching against a known set of text, could potentially be performed on-chain, most other processes, like AML analysis and fraud analysis, rely on processing large datasets and are likely to continue being processed off-chain. Even for sanction screening, there are often mechanisms such as fuzzy matching to improve matches (reduce false negatives) and additional analysis to reduce false positives. Therefore, having payment control processes fully applied on-chain is unlikely.

Applying controls after a transaction is initiated but before it is completed leads to an atomicity problem. It is also unclear whether current regulations allow banks to apply controls on a post-facto basis, i.e., after a transaction is completed. The natural conclusion, then, is that controls must be applied before a payment is initiated on-chain.

The next consideration is how the authorization of a payment, following successful application of payment controls, can be made known to the smart contracts on the blockchain as controls are transaction-specific, i.e., to a specific recipient and for a specific amount. Authorization will also have to be specific and granular.

One option is for the token issuer to update the smart contract directly. For example, an account is added to an allow-list when a payment is authorized. This is not ideal, as it requires a sizable amount of information to be stored on-chain. Allow-listing only the address without further information is not viable, as a user could seek authorization for

one payment and then make other transactions that have not been authorized, while their account is on the allow-list. This leads us to a design of having a user request authorization through the wallet, having the authorization included when making an on-chain transfer and having the transfer and authorization validated on-chain to ensure that the transfer is within the parameters of the authorization.

Multi-use authorization: Another consideration is whether an authorization could be used multiple times. It is possible that users may be authorized to perform certain lower-risk transactions for a longer period of time, such as payments to the bank and government entities, or low-value transactions below a threshold. Supporting long-lived authorization necessitates a capability to revoke authorization, and for storage and retrieval of authorization in the blockchain wallet, which adds complexity.

Delegated authorization: Token issuers may wish to delegate the authority to issue authorizations to other intermediaries, such as a bank that distributes the payment tokens to its clients. The flow is hence designed to allow authorization to be requested from different parties, and for the signed authorization to be validated against a registry of public keys of parties authorized to issue payment authorizations.

Flexibility for different jurisdictions and banks: The payment information captured is likely to differ across jurisdictions and banks. Rather than specifying the specific payment information to be captured, a two-step process is taken, where a list of required fields of information is first retrieved, before the users' inputs to the required information are captured and transmitted. It would also be possible for the wallet to cache both the fields required and users' inputs, to minimize the repeated communications and the complexity borne by users.

There are three key parts to the proposed design:

1. Information flow between the blockchain wallet, the blockchain and the authorizer
2. Authorization structure and parameters
3. On-chain validation of authorization and transaction

Information flow: Our proposed design is for the blockchain wallet to look up the API endpoint for a specific authorizer, request a schema of the payments information required from the authorizer of the information required, present the fields for the user's inputs and then submit the payments information to the authorizer. The authorizer will perform payment control processes based on the payment information submitted and respond to the wallet with an authorization. This authorization is included with the transfer request on-chain. Note that the steps may not need to be performed at the same time. For example, the wallet may maintain a cached copy of the schema and not request it from the authorizer or the user for every transaction. The authorization may also be long-lived and valid for longer than a day.

Authorization: The authorization will include multiple parameters, including validity period, maximum transaction amount, debit addresses and credit addresses.

On-chain validation: The transfer will be completed only after successful validation of the authorization. The digital signature of the authorization will be validated to check that it is valid and signed by a valid authorizer, that the authorization has not been revoked and that the transfer is within the parameters of the authorization.

Standards proposal

The proposed design will see interactions between the blockchain wallet, the bank and the payment token smart contract on the blockchain for the transmission of payment information for applying payment control. Since the interaction involves multiple parties and there are various solution providers for each component, defining the interaction as an interoperability standard is important to ensure that the proposed information flow can be implemented across all parties. For this reason, we will work with the financial and blockchain ecosystems to propose a new standard. The design and code will be published under an open-source license to serve as a starting point for further standards work in this area, in collaboration with the financial services and blockchain ecosystems.

Part III: Proposed designs for key areas

Administrative controls for remediation and recovery actions

This section explores administrative controls for remediation and recovery actions, such as suspending transactions and seizing funds.

Background and design considerations

It is common in traditional financial systems for banks to have the ability to suspend accounts, freeze funds or even seize them. This could be for the benefit of the account holder, such as when there is suspected fraudulent activity, and accounts are suspended to protect users and minimize financial losses. They could also be performed to protect the integrity of the financial system or when directed by law enforcement agencies, such as when money laundering activities are detected.

It is important that these capabilities are built into the payment tokens to allow similar operational processes to be performed. Public blockchains and early crypto tokens were designed to be censorship-resistant, meaning that no individuals or organizations, including regulatory authorities, can prevent the processing of transactions. It should be noted that these are design choices, not inherent characteristics of tokens on decentralized ledgers. Smart contracts allow for any logic to be implemented, including logic that stops or prevents further processing, such as the onward transfer of tokens.

In addition to suspending accounts, assets and funds may need to be seized and recovered, necessitating capabilities for administrators to conduct transfers on suspended accounts. There are often bank-specific processes to follow, which are likely to differ across banks and jurisdictions. Therefore, the function needs to be designed flexibly to support various possible processes.

Another important capability is for a global suspension or halting of all token transactions in crisis scenarios. This last resort capability is sometimes referred to as a circuit breaker or a kill switch. In traditional finance, exchanges typically impose circuit breakers where anomalous trading activities, such as a significant sharp price change, trigger an automated pause on trading for further review and investigation. Even within the crypto world, pause functions are increasingly incorporated into smart contracts to enable rapid responses in emergencies. Such capabilities have been observed in many stablecoins, including USDT and USDC.

This capability is also important for compliance with requirements for orderly resolution. If an order for resolution is issued, there will be a need to stop further transactions while the institution is taken into receivership or conservatorship. Such orders were previously issued after business hours, when accounting books were closed for the day and transactions were no longer processed. With 24/7 transactions, it is important that any such orders are followed and instituted promptly, as transaction processing continues even while the order is being issued.

The capability is also useful when significant risk events materialize, including technology risks such as the identification of a critical bug or major hack, and systemic risks such as ecosystem-level failures and massive outflows of funds, where a global pause is needed while the situation is being remedied. It is worth noting that this is not a new capability. Traditional systems can be easily disconnected or switched off. However, blockchains are designed to be resilient, so switching off applications on a blockchain requires such switches to be built into the smart contracts.

However, the existence of such functions may raise concerns⁸ about potential misuse. The use of such functions hence, must be observable and not obfuscated to ensure trust and transparency. In the case of seizing funds, one possibility could be the use of `TransferFrom`, where the bank's wallet address is given the authority to transfer on behalf of the user. However, using the same function for both administrative and regular purposes would obfuscate usage. We hence encourage a separate function specifically for administrative use.

It is also important to note that transparency and privacy are often at odds. While the general preference on open blockchains is for transparency, there may be other reasons, including regulatory requirements, that favor or necessitate privacy over transparency. An example would be the policy or regulatory requirement of not tipping off a user that they are the subject of investigations. Another may simply be that users prefer the existing paradigm whereby transaction information is only delivered on a need-to-know basis, and is not publicly available to any observer.

High-level design

The payment token prototype we are developing will include three sets of capabilities, to allow for suspension of accounts, seizure of tokens and pausing of all token transactions.

Suspension of accounts and freezing of tokens: Accounts will have an associated Boolean flag that determines whether an account is able to transact. Transactions that are initiated by users or their delegates will first have the flag checked to see if the account is suspended and if the transaction can be processed. There will be an administrative function for authorized parties to suspend the accounts, which switches the flag to prevent further transactions. It is recommended that token issuers clearly outline the scenarios for using such features, and there should be strict adherence to the outlined procedures to ensure transparency and trust from users and regulatory agencies.

Seizure of tokens: There will be an additional administrative function to move funds out of a suspended account. This will be a separate function from the regular `TransferFrom` function and can be used on an account that is suspended.

Halting or pausing all token transactions: There will be an additional administrative function to pause all regular non-administrative transactions. When activated, users will not be able to deposit, transfer or withdraw their tokens. This will likely be implemented using the Pausable design pattern and library made available by Open Zeppelin. Functionally, this should only be used as a last resort, and permissions should be granted only to pre-approved administrators, likely through a multi-party solution. In addition to incorporating the capability, it is important to develop an operational plan around its use, including determining the parties with the necessary authorization, the process to obtain the necessary digital signatures to initiate the operation on the blockchain, and the process of performing the operation, noting the potential for front-running of transactions similar to MEV front-running¹² attacks.

Standards proposal

Even though there are no specific regulatory requirements for such on-chain functions today, we believe that defining a common set of functions for banks to implement as part of their token offerings can enhance regulatory comfort. Token issuers will be able to demonstrate that safety and security are designed and incorporated into these on-chain products, and there is an ability to perform remediation and recovery actions when the need arises. We hope to work with international organizations and standards-setting bodies to further develop and promote the implementation of these recommended practices.

From an interoperability perspective, enabling industry-level threat detection and response is also important. National or industry-level threat monitoring for cyber risks is already common, often facilitated by Information Sharing and Analysis Centers (ISACs). With applications deployed on a common platform, there is potential for cooperation to advance beyond information sharing to joint responses. For example, the discovery of a critical vulnerability might necessitate a global pause of all affected smart contracts, and a common implementation framework for such functions would enable easier coordination of these responses.

Part III: Post-development findings and future considerations

Improving payments efficiency while managing risks

A key observation in this report is that directly applying current regulations and conventions of traditional payment processing to a token world limits potential efficiency gains. In this section, we explore how risks can be adequately or even better managed in a tokenized environment, enabling changes in payment processing and application of payment controls to fully harness benefits. We share a few areas where the full potential of the design was constrained by existing regulations, and where risks could still be adequately managed, describing how changes in regulations could enable further efficiency gains.

Evolving beyond a message-based architecture and legacy regulation

Today, payments are not executed in isolation. The value chain, from initiation to settlement, often requires a single payment instruction to move through multiple institutions, and, in some cases, even flow through financial market infrastructure and central banks before settling. Data is transmitted across these institutions and infrastructures, such that each institution can apply in-line payment controls within their own siloed systems.

The payment message and fund settlement are decoupled, requiring each intermediate to bilaterally reconcile their own distinct ledgers. This introduces latency risk and lengthy settlement times, necessitating institutions to manage capital risk across jurisdictions, which results in substantial operational costs. These costs are further exacerbated in the exception scenarios, where payment breaks, false positives, inconsistent data formatting and potential outages across a chain of API and message-based systems contribute to further operational inefficiencies. In joint research with J.P. Morgan, Oliver Wyman estimated that transaction costs alone for processing \$23.5 trillion in cross-border payments exceed \$120 billion—without considering FX conversion, trapped liquidity and delayed settlements.¹³

In recent years, distributed ledger technology (DLT) has emerged as a promising solution to address these challenges in the payment system. Unlike legacy technologies that depend on message transmission between institutions, DLT offers a unified or shared environment where payment and settlement are seamlessly integrated into a single transaction, and participants leverage a shared global state. Even where payments must travel across multiple institutions—such as cross-border and cross-currency transactions requiring several intermediaries—DLT allows these transactions to execute instantly and atomically. This significantly reduces the end-to-end processing time, minimizes system silos and eliminates the need for extensive bilateral reconciliation efforts.

However, in this research, we observed that even with a DLT-based payment system, there are current limitations to the efficiencies that the design can bring due to current financial regulations. Regulations are designed to be longstanding, meaning that many of the current financial regulations would have been enacted years ago, and were designed to mitigate risks according to the capabilities and constraints of legacy technologies.

Banks building products such as payment tokens remain compliant with these regulations, even when they contribute to inefficiency. With modern technologies leveraged in this design, we observed how these technologies can be utilized to alleviate some of these trade-offs, pinpointing possible changes in regulations and practices to enable more efficient payments while still safeguarding the safety and integrity of the financial system. Due to the nature of the work, we identify several areas for development pertaining to payment controls such as Anti-Money Laundering / Countering the Financing of Terrorism (AML/CFT), know-your-customer (KYC), sanctions and operational safeguards which are required for banks (and their payment tokens) to be compliant across most jurisdictions. These processes collectively ensure that banks do not transmit illicit or malicious transactions.

Opportunities to improve payment efficiency

Banks take active measures to ensure they are not banking or transmitting payments for illicit or malicious actors. As an example, when we look at current payment ecosystems, key requirements for AML/CFT compliance programs include transactions monitoring and compliance with the Financial Action Task Force (FATF) Recommendation 16 on wire transfers (also known as the travel rule) that have been designed to mitigate illicit finance risk present in relatively open-ended transactions.

For instance, a wire payment can be routed to any beneficiary in the world, provided they have a bank account at an institution with a BIC code. Disclosures embedded in the Travel Rule help receiving institutions and law enforcement gain context and transparency around the transactions they are transmitting. An institution's ability to control the transmission of these funds are limited to the period in which they are within the institution, requiring in-line payment controls and disclosures to ensure safety and soundness of the payment and recipients.

Remedial actions are increasingly difficult to conduct as funds and messages move disjointedly, so coordination across intermediaries is required to track down the payment and there is currently no shared real-time view for fast coordination and action. Post-facto remedial actions are hence difficult to apply in legacy payments, which are broad and unrestricted.

In contrast, payment tokens introduce three key functionalities:

1. Issuance on open blockchain networks which provides a shared, auditable ledger, introducing transparency to parties in the value chain where previously payments were opaque
2. Programmability, including attestation functionality, which introduces the ability to apply automated conditional controls logic and provide proof of rule adherence, which must be executed in order for a participant to access the payment tokens
3. Technically-enforced access authorizations via role-based permissions and administrative controls embedded into the payment token

These features establish a foundation where peer-to-peer transactions can occur without introducing processing trade-offs, as banks apply controls to ensure they are not transmitting payments for illicit or malicious actors, which is an aspect of utmost importance. If regulation evolves accordingly, we could explore how controls can be

implemented within a new boundary, which is at any time that the payment token is in circulation in the network. This allows for self-contained application of in-line controls and enhanced post-facto review and remediation, helping to achieve the desired efficiency gains by reducing on/off-chain processing, without sacrificing robust operational safeguards. This is explored in two key ways:

- Allowing payment controls to be applied post-facto, to minimize dependencies of on-chain settlement on off-chain processing
- Allowing the splitting of control processes to minimize cross-institutional dependencies within trusted ecosystems

Moving from in-line contingent application of payment controls to asynchronous controls with post-facto review, analysis, and remediation

Throughout this work, we have explored a number of pillars in our design guidelines which improve the ability to trace payments and automatically execute actions where compliance logic is required. This includes permissioned transfers, delegated controls, account or asset seizure and global pauses of all payment token transactions, as well as an attestation model which ensures that users are expressly given the authorization to transact based on certain conditions by their bank. These features are embedded to the payment token for automatic execution, ensuring governance and compliance itself are embedded in the token. In scenarios where seizures of funds are required due to potential regulatory violations, payment tokens can be traced in near-real-time, and wallets or funds can be subsequently seized, even if there have been further transfers with the use of proposed admin controls.

This is a clear contrast from legacy payments; in correspondent banking today, if funds were released from one bank to the next, tracking and seizing funds through the payment chain would be difficult. As such, controls are currently applied before a bank performs a transfer to ensure that all AML/CFT compliance, sanctions obligations and other regulatory obligations are met so that remedial actions—such as suspending transactions and seizing funds—can be performed if needed while the funds are still within an institution in the value chain.

The design of payment tokens offers a new opportunity to reduce the friction and latency of proactive controls, and minimize processing time by having reactive controls with an enhanced ability to review, analyze and remediate payments post-facto. This is largely due to the change in the control boundary introduced by the technology—it moves away from being limited to when funds are within an institution's processing (ending for that institution when routed to another) to the edges of the common network itself. As long as the payment token is in circulation, controls may be implemented. This ensures that, unlike traditional correspondent banking where issuing institutions and remedial institutions must both track payments down the chain to take remedial actions, a payment token issuer can execute controls and monitor payment tokens dynamically at any point in time.

This should continue to be complemented with in-line and contingent controls when applied as the payment token is entering or exiting the boundary at the on and off-ramp. In order to mitigate any potential risks of transmitting illicit or malicious payments, this should be considered as a complement to attestations explored in this work, where institutions can pre-validate certain transaction capabilities for end users. This sets a strong foundation for a balance of proactive and reactive controls without sacrificing the ability for peer-to-peer and composable payment tokens, as transactions work with self-sufficient and post-facto controls.

Asynchronous controls

Another opportunity to enhance efficiency in payment controls, as observed from this work, is the ability to decouple the application of payment controls between the payer and payee, allowing banks to enforce compliance measures asynchronously. As highlighted earlier, institutions currently can apply controls or remedial actions only while funds are held by them, meaning that controls must be applied by two independent institutions (sender and recipient) on either side of the transaction for funds to settle.

The design of payment tokens enables the testing of new innovations; both senders and receivers can have their respective KYC verified, with an attestation issued by their sponsoring institutions to ensure their technical authorization to transact. The sender may have an additional set of attestations enabling their ability to enter into a specific transaction, while the recipient have not received the specific authorization. The payment may nonetheless execute atomically. However, the funds will settle to the recipient's wallet instantly while remaining in a locked state until further payment controls are applied on the payee by their institution. The issuing institution may unlock the funds when controls are sufficiently validated, and this step could happen at any point between the receiving party receiving the funds to spending them.

One possible change from the current process is that, rather than verifying both payer and payee, the responsibility of screening a party could fall solely on the party's bank. This means that a sending bank can apply risk assessments and regulatory checks on the payer before initiating a transaction, while the receiving bank can independently enforce its own controls on the payee upon receipt—ensuring compliance without delaying settlement. This can offer a number of efficiencies and the ability to maintain transaction atomicity.

From this work, we conclude that the new design features explored in this implementation of payment tokens offer a promising approach to aligning with financial regulations aimed at safeguarding the stability and integrity of financial systems. These features provide a valuable balance between efficiency and risk reduction, particularly in preventing illicit or malicious transactions. Further exploration is warranted, especially in the context of regulatory evolution, to create a permissive environment that fully leverages the comprehensive feature set of payment tokens.

Part III: Post-development findings and future considerations

Conclusion and future exploration

The collaboration between MIT DCI and Kinexys developed resources for financial institutions looking to develop payment tokens, in the form of a handbook detailing design guidelines, key functionalities (particularly those required for regulatory compliance) and a map of existing token standards to these functionalities. It also identified areas where new token standards are needed, and described the high level design, with source code and detailed design to be published subsequently.

Further dialogue and collaboration with the ecosystem

We hope that these resources can serve as a starting point for further dialogue and collaboration with the ecosystem, in a few areas:

- 1. Design principles and best practices:** Establishing a set of design principles and best practices for token issuers can significantly enhance the general level of safety and integrity in payment token products, while also increasing regulatory confidence. The design guidelines listed can be enriched by incorporating the experiences of other ecosystem participants with large-scale, live products, and refined into design principles and best practices for the industry.
- 2. Blueprint for payment tokens:** We foresee a future where users can choose between various standards based on their needs and implement them seamlessly in a coherent and harmonized manner. A blueprint that outlines a list of functionalities, along with the token standards or software libraries available to implement these functionalities, would be invaluable. The functionalities and token standards mapping in this report could serve as an initial blueprint, which must be regularly updated as new token standards and libraries emerge.
- 3. Approach to token standards:** The blueprint approach is effective only if standards are modular, composable and non-overlapping. While reaching consensus on a single set of standards may be challenging for ecosystem participants, agreeing on the approach might be more feasible. We hope that continued dialogue within the ecosystem will facilitate alignment on the approach to proposing new standards.
- 4. Proposal of new token standards:** We have identified two areas where new capabilities are needed and plan to collaborate with the financial and blockchain ecosystems to propose new standards. The source code and detailed design, which will be published subsequently, could serve as an initial draft for these initiatives.

In the course of the project, there were also areas of considerations that were identified as open issues for further research, and not specifically addressed in the project.

Managing risks of censorship and front-running attacks

A key concern is censorship, where transactions are censored or not processed. Censorship can occur at different levels: nodes may refuse to relay transactions to other nodes, preventing them from entering the mempool for pick-up and processing, or mining pools may refuse to include the transactions in the blocks they create. Such censorship can be by design, such as when gas fees for transactions are set too low, but it can also be malicious, such as when nodes actively censor transactions from specific addresses or those interacting with specific smart contracts.

Another concern is front-running, where transactions are reordered to achieve specific results. For example, a party might redeem all its payment tokens if it detects that the token issuer plans to suspend transactions. This could be achieved by monitoring the use of administrative functions, such as suspending addresses or initiating a global pause, and then pushing its redeem transaction ahead by offering higher gas fees or outright censoring the targeted transactions.

Managing these issues would require having a stake at the blockchain layer, such as by hosting nodes and mining transactions. Hosting a node and having direct connections to other nodes can prevent censorship of relayed transactions, while mining transactions can prevent censorship of transactions being included in blocks. Interestingly, the solution to front-running would be to mine transactions directly, without relaying the unconfirmed transactions to other nodes. Not relaying transactions can prevent others from learning about an upcoming transaction and thus front-running it. However, there needs to be sufficient mining power (i.e., enough stake or hashing power) to ensure that blocks can be mined in a timely manner.

Use of open blockchains for record-keeping

There are strict requirements for storing official books and records to ensure that records are available and cannot be tampered with. Meeting these requirements, when a system is not operated by a single entity, requires a different approach. There are two main areas of concern: integrity and availability.

Integrity: Blockchains provide inherent security for data integrity, as blockchain records are tamper-resistant, and digital signing of transactions makes it difficult for counterfeit or fraudulent records to be created. Integrity was not a critical concern, as it is well managed by most blockchains in their underlying technical design.

Availability: While it is computationally infeasible to change or fraudulently create new records, it may be easier to remove past records. Blockchain nodes do this automatically when there are block reorganizations, and some blocks are dropped in favor of a longer chain of blocks. This relates to finality, and a risk-based risk management strategy necessitates waiting for a set number of block confirmations before a transaction is deemed final and complete. However, it is also possible that nodes are deliberately attacked or changed to force a significant block reorganization outside of typical parameters.

In such cases, risks can be managed through monitoring and data backups. This risk is not new, and the possibility of catastrophic system failure necessitates regular backups even in traditional architectures. With blockchain, one model of enabling backups would be through running nodes with customized consensus logic that prioritizes the preservation of blocks over achieving consensus. For example, there could be logic to not accept block reorganization

that exceeds a certain depth, opting instead to fork the network rather than reorganize a long chain of blocks to continue synchronization with the rest of the network. Resynchronization would be possible through ledger adjustment entries, similar to how logfiles are used to bring a backup up-to-date with the latest set of records.

While record-keeping on open blockchains introduces new considerations, there are mechanisms available to manage these risks. Detailing these considerations and how to manage them, potentially in a set of industry best practices, can help improve the level of understanding across the ecosystem, and bring greater comfort to regulators in the process.

Governance of open blockchains

Concerns stem mainly from the governance of open blockchains, particularly public blockchains where there is no central operator. In such scenarios, operating part of the infrastructure, such as hosting nodes and mining blocks, might be required to mitigate the risks. One possibility is for banks to manage their own infrastructure. However, it is unlikely that individual banks will acquire sufficient mining power to mine new blocks regularly, so there might be a need to develop direct relationships with mining pools, governed by legal contracts and service level agreements.

Even with strong mitigation mechanisms in place, residual risks are likely. The use of cloud services provides a useful analogy for the use of public blockchains. Cloud services are ubiquitous today, with even critical processing performed in the cloud. This was unthinkable barely a decade ago. Moving from on-premise systems, where the entire software and hardware stack was under the sole control of an entity, to cloud services, where hardware and operating systems are managed by a third party, was a complete change in design paradigm. Regulatory clarity, through clarification and changes in regulations and guidelines, helped enable the use of cloud services in financial services. It is likely that similar clarity would encourage more pervasive adoption of public blockchains in financial services.

Defining settlement finality

In the traditional banking system, finality is a legally defined stage in the payment process that signifies when ownership of property is irrevocably confirmed and unconditional. Finality is crucial for users, especially the recipients, as it provides certainty on the ownership of the property (funds or assets) received.

The decentralized nature of public blockchains and the consensus mechanisms used to achieve decentralization allow for blocks to be proposed by different parties, sometimes simultaneously. This can lead to situations where some blocks are discarded in favor of a longer chain. Consequently, finality in this context is probabilistic rather than deterministic. As the depth of a block increases, the likelihood of a transaction being discarded decreases, thereby increasing the confidence that the transaction is final.

The situation becomes more complex with Layer 2 (L2) networks. These networks operate semi-independently from Layer 1 (L1), inheriting the trust properties of the underlying network but often featuring different architectures, efficiencies or other unique characteristics. There are often dependencies on L1, and transaction finality on L2 depends on certain records being updated on L1 and specific conditions being met.

Administrators of payment token arrangements will need to understand this new control scheme and effectively manage the associated risks while providing an adequate level of assurance to payment token holders.

Disclaimer

The information in this report, or on which this report is based, has been obtained from sources that the authors believe to be reliable and accurate. However, such information has not been independently verified and no representation or warranty, express or implied, is made as to the accuracy or completeness of any information obtained from third parties.

The information and conclusions are provided as at the date of this report and are subject to change without notice, and MIT and J.P. Morgan undertake no obligation to update or revise any information or conclusions contained herein, whether as a result of new information, future events, or otherwise. The information and conclusions provided in this report take no account of any relevant persons' individual circumstances, should not be taken as specific advice on the merits of any investment decision, product, or service and should not be deemed to be a reasonably sufficient basis upon which to make an investment decision or undertake any product or service.

This report is not intended to provide and should not be relied on for, accounting, legal, or tax advice, or investment recommendations. Please consult your own tax, legal, accounting, or investment advisor concerning such matters. MIT, J.P. Morgan and their respective affiliates accept no liability for any loss arising from any action taken or refrained from as a result of information and conclusions contained in this report or any reports or sources of information referred to herein, or for any consequential, special or similar damages even if advised of the possibility of such damages. This report has been provided solely for information purposes and does not constitute a recommendation, advice, or an offer or solicitation to buy or sell any securities or financial instruments or of any product or service. It should not be so construed, nor should it or any part of it form the basis of, or be relied on in connection with, any contract or commitment whatsoever.

Further, this report shall not be considered advice on the merits of acquiring or disposing of any particular investment or as an invitation or inducement to engage in any investment activity or other product or service. By accepting this report, you agree to be bound by the foregoing limitations. J.P. Morgan is a marketing name for the payments business of JPMorgan Chase Bank, N.A. and its affiliates worldwide. JPMorgan Chase Bank, N.A., is organized under the laws of USA with limited liability.

References

1. Bank for International Settlements. (2023). *Annual report 2023: Chapter III*. Retrieved from <https://www.bis.org/publ/arpdf/ar2023e3.htm>
2. Bank for International Settlements. (2024). *Working paper No. 1178*. Retrieved from <https://www.bis.org/publ/work1178.htm>
3. Monetary Authority of Singapore. (2024). *GL1 Whitepaper*. Retrieved from <https://www.mas.gov.sg/publications/monographs-or-information-paper/2024/gl1-whitepaper>
4. CoinMarketCap. (2025). *Stablecoin*. Retrieved from <https://coinmarketcap.com/view/stablecoin/>
5. Federal Reserve Bank of St. Louis. (2025). *Deposits, All Commercial Banks (DPSACBW027SBOG)*. Retrieved from <https://fred.stlouisfed.org/series/DPSACBW027SBOG>
6. National Institute of Standards and Technology. (2024). *Five functions*. Retrieved from <https://www.nist.gov/cyberframework/getting-started/online-learning/five-functions>
7. National Institute of Standards and Technology. (2025). *What is post-quantum cryptography?*. Retrieved from <https://www.nist.gov/cybersecurity/what-post-quantum-cryptography>
8. Blockworks. (2023). *PayPal PYUSD stablecoin centralization*. Retrieved from <https://blockworks.co/news/paypal-pyusd-stablecoin-centralization>
9. ISO 20022. (n.d.). *ISO 20022 message definitions*. Retrieved from <https://www.iso20022.org/iso-20022-message-definitions>
10. MetaMask. (2023). *Account abstraction: Past, present, future*. Retrieved from <https://metamask.io/news/account-abstraction-past-present-future>
11. LF Decentralized Trust. (2025). *Interoperability workshop*. Retrieved from <https://www.lfdecentralizedtrust.org/events/interoperability-workshop>
12. Halborn. (2021). *What is a front-running attack?*. Retrieved from <https://www.halborn.com/blog/post/what-is-a-front-running-attack>
13. Oliver Wyman. (2021). *Unlocking \$120 billion value in cross-border payments*. Retrieved from <https://www.oliverwyman.com/our-expertise/insights/2021/nov/unlocking-120-billion-value-in-cross-border-payments.html>