QualitaX

Deep Dive into ERC-1400: Enabling Secure and Compliant Digital Securities.

May 2024

AUTHOR

Bradley Stone - Research Lead, QualitaX

REVIEWERS

Julien Brodier - CTO & Co-Founder - Talium

Claus Skaaning - CEO & Co-Founder - DigiShares

CONTRIBUTORS

Anaïs Ofranc - QualitaX

François de Chezelles - Talium

Olesia Bilenka - Hacken

Olena Dikhtiaruk - Hacken

Jim Zhang - Kaleido

Brindrajsinh Chauhan - Kaleido

John Hosie - Kaleido

Samuel Edoumou - SecurTok

Chaals Nevile - Enterprise Ethereum Alliance

Ray Buckton – RWA World

Martin Hokta - DigiShares

Table of Contents

ERC-1400 Overview

- **Security Tokenization:** Specifically designed for the tokenization of securities, ERC-1400 accommodates multiple asset classes and partitions, allowing for the representation of different share classes within a single token structure.
- **Standardized Compliance and Identity Management:** ERC-1400 enables compliance through flexible on-chain and off-chain transfer restrictions. It supports identity management integration with existing systems, enabling verification of investor eligibility and compliance with KYC/AML regulations.
- **Transferability Rules and Compliance:** Features robust transferability rules that incorporate both on-chain and off-chain data. Enables the use of off-chain authorization, ensuring that transfers comply with legal and contractual obligations. Any transferability rule can be implemented (including complex, confidential and upgradable).
- **Document Management:** Incorporates ERC-1643 for managing documents associated with tokens, ensuring that legal and regulatory documents are securely linked to the token, enhancing transparency and trust.
- **Interoperability and Compatibility:** Ensures seamless interaction with existing Ethereum standards such as ERC-20, ERC-165, ERC-173, and ERC-725. It is designed to be compatible with current wallets, exchanges, and tools within the Ethereum ecosystem.

Summary

Problem: Tokenizing securities on Ethereum and EVM-compatible networks presents challenges in maintaining regulatory compliance and ensuring the necessary restrictions on token transfers. Existing token standards, such as ERC-20, do not provide built-in mechanisms for enforcing compliance.

Solution: ERC-1400 aims to address these challenges by providing a comprehensive set of standards for issuing and managing security tokens on Ethereum and EVM-compatible networks. The standard incorporates features such as transfer restrictions, and document management to facilitate compliance with regulatory requirements.

Technical level: ERC-1400 is a set of proposed smart contract standards that extend the functionality of ERC-20 tokens. It introduces additional functions and events to support the unique requirements of security tokens, such as transfer restrictions and controller operations.

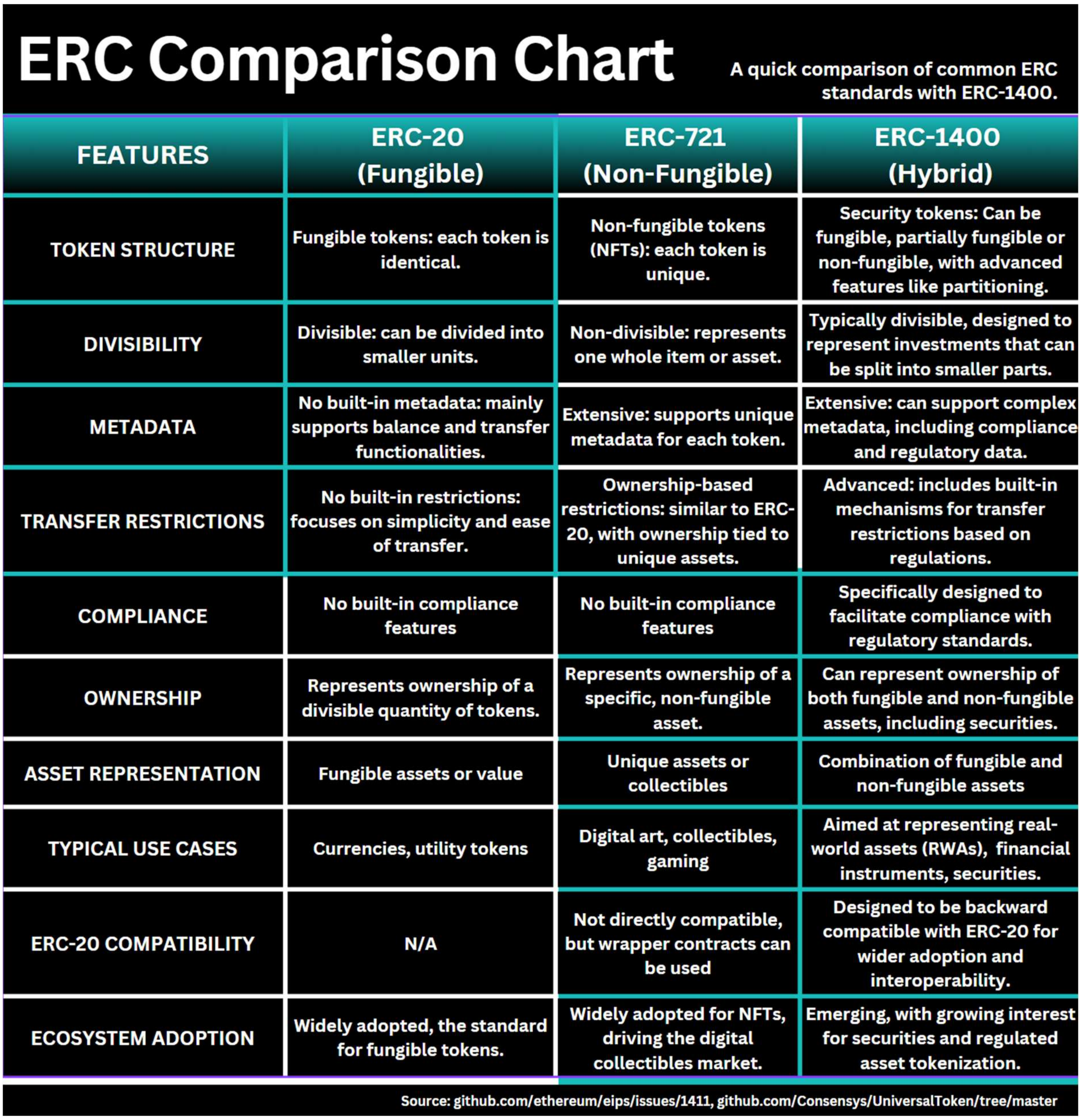
Practical applications: ERC-1400 can be used to tokenize various types of securities, including equity, and debt. ERC-1400 aims to enable more efficient and compliant trading and management of these assets on the Ethereum blockchain or EVM-compatible networks by providing a standardized framework for security tokens.

Introduction

With the maturity of blockchain technologies, token standards like ERC-20 (Fungible Token Standard) and ERC-721 (Non-Fungible Token Standard) have been instrumental in driving the growth of decentralized finance (DeFi) and digital collectibles markets. However, the tokenization of real-world assets (RWAs), such as real estate or financial instruments, presents unique challenges, particularly in the context of compliance with regulatory requirements.

Traditional securities markets are heavily regulated. The lack of compliance support at the token level can increase issuers' and investors' exposure to higher risks. This is why there is a need for token standards with built-in mechanisms to enforce necessary restrictions and compliance measures, such as identity verification, accreditation checks, and transfer restrictions. ERC-1400 provides such a framework, aiming to address the unique requirements of tokenized securities.

Figure 1: Comparative Analysis of ERC Token Standards



ERC Comparison Chart

A quick comparison of common ERC standards with ERC-1400.

FEATURES	ERC-20 (Fungible)	ERC-721 (Non-Fungible)	ERC-1400 (Hybrid)
TOKEN STRUCTURE	Fungible tokens: each token is identical.	Non-fungible tokens (NFTs): each token is unique.	Security tokens: Can be fungible, partially fungible or non-fungible, with advanced features like partitioning.
DIVISIBILITY	Divisible: can be divided into smaller units.	Non-divisible: represents one whole item or asset.	Typically divisible, designed to represent investments that can be split into smaller parts.
METADATA	No built-in metadata: mainly supports balance and transfer functionalities.	Extensive: supports unique metadata for each token.	Extensive: can support complex metadata, including compliance and regulatory data.
TRANSFER RESTRICTIONS	No built-in restrictions: focuses on simplicity and ease of transfer.	Ownership-based restrictions: similar to ERC-20, with ownership tied to unique assets.	Advanced: includes built-in mechanisms for transfer restrictions based on regulations.
COMPLIANCE	No built-in compliance features	No built-in compliance features	Specifically designed to facilitate compliance with regulatory standards.
OWNERSHIP	Represents ownership of a divisible quantity of tokens.	Represents ownership of a specific, non-fungible asset.	Can represent ownership of both fungible and non-fungible assets, including securities.
ASSET REPRESENTATION	Fungible assets or value	Unique assets or collectibles	Combination of fungible and non-fungible assets
TYPICAL USE CASES	Currencies, utility tokens	Digital art, collectibles, gaming	Aimed at representing real-world assets (RWAs), financial instruments, securities.
ERC-20 COMPATIBILITY	N/A	Not directly compatible, but wrapper contracts can be used	Designed to be backward compatible with ERC-20 for wider adoption and interoperability.
ECOSYSTEM ADOPTION	Widely adopted, the standard for fungible tokens.	Widely adopted for NFTs, driving the digital collectibles market.	Emerging, with growing interest for securities and regulated asset tokenization.

Source: github.com/ethereum/eips/issues/1411, github.com/Consensys/UniversalToken/tree/master

By helping to address the requirements associated with security tokens, ERC-1400 aims to facilitate the growth and adoption of tokenized securities in a legally compliant way. ERC-1400 is particularly relevant in addressing the requirements for the tokenization of financial instruments.

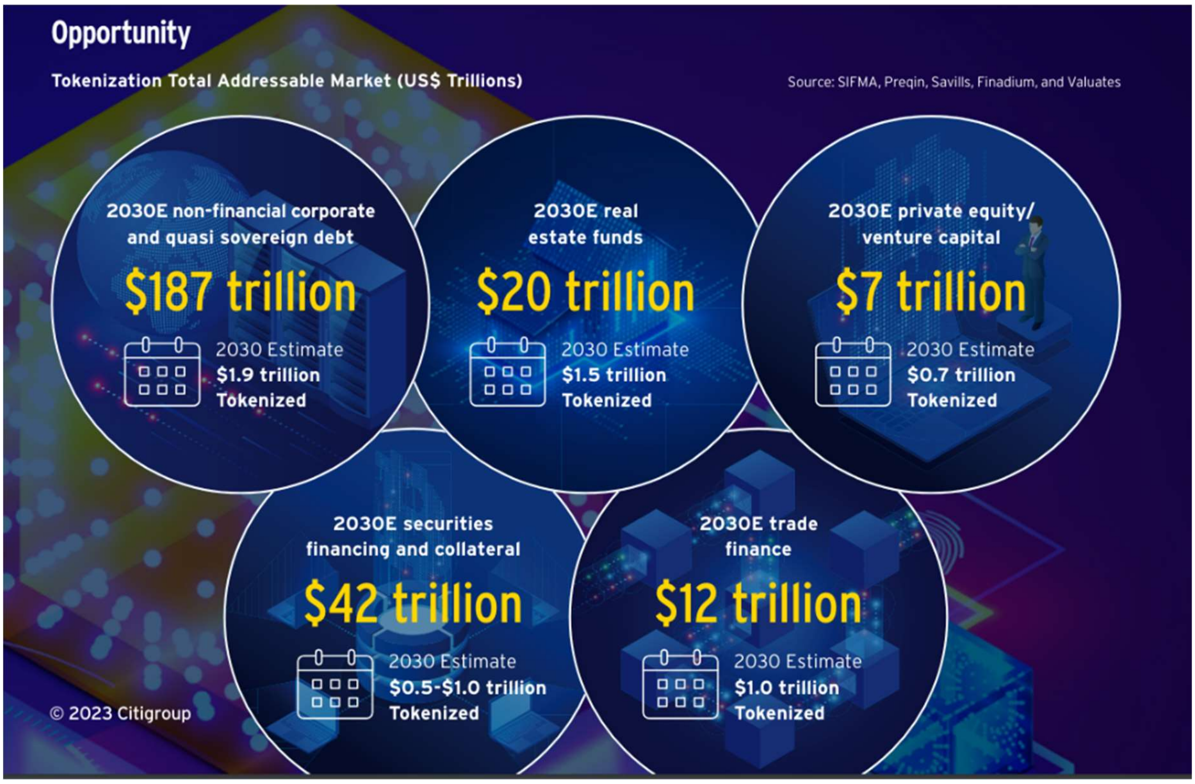
In the following section, we will briefly explore some of the different types of financial instruments that can be tokenized using the ERC-1400 standard.

Tokenization of Financial Instruments

Tokenization of financial instruments refers to the process of representing traditional financial assets, such as stocks, bonds, derivatives, or other securities, as digital tokens on a blockchain network. It involves creating a digital representation of the ownership and rights associated with the underlying financial instrument. This process of digitalization offers several unique advantages compared to traditional securities. By leveraging blockchain technology, tokenization can streamline operational processes, reduce transaction and asset servicing costs, and mitigate risks through immutable data and smart contracts. Moreover, tokenization can open up new opportunities by making previously illiquid assets more accessible to investors, ultimately enhancing market efficiency and expanding investment possibilities.

Citigroup forecasts $4 trillion to $5 trillion of tokenized digital securities and $1 trillion of distributed ledger technology (DLT)-based trade finance volumes by 2030.

Figure 2: Citigroup Tokenization Total Addressable Market



Opportunity

Tokenization Total Addressable Market (US$ Trillions)

Source: SIFMA, Preqin, Savills, Finadium, and Valuates

2030E non-financial corporate and quasi sovereign debt
$187 trillion
2030 Estimate
$1.9 trillion
Tokenized

2030E real estate funds
$20 trillion
2030 Estimate
$1.5 trillion
Tokenized

2030E private equity/ venture capital
$7 trillion
2030 Estimate
$0.7 trillion
Tokenized

2030E securities financing and collateral
$42 trillion
2030 Estimate
$0.5-$1.0 trillion
Tokenized

2030E trade finance
$12 trillion
2030 Estimate
$1.0 trillion
Tokenized

© 2023 Citigroup

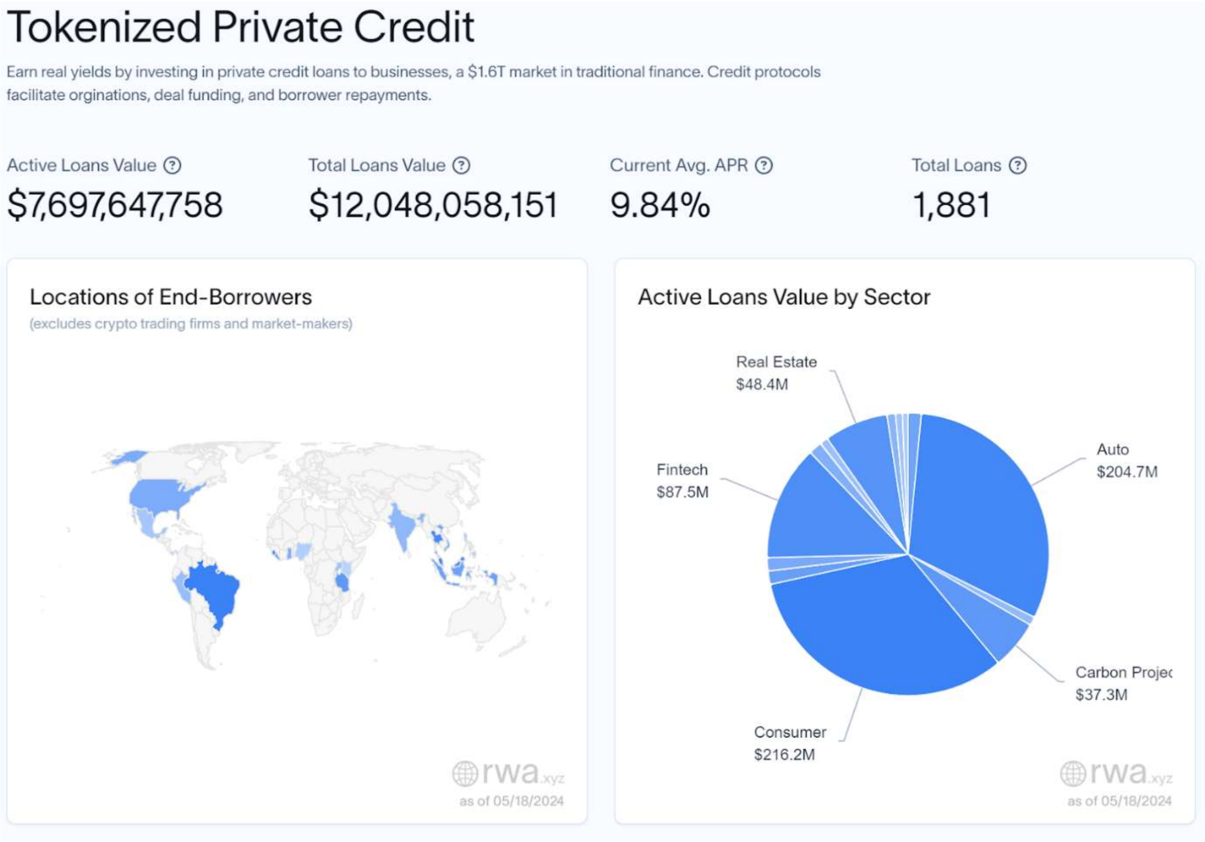
Source: Money, Tokens and Game, Citi Group (2023)

Tokenized Equity

Tokenized equity represents digital ownership of shares in a company using blockchain or DLT technology. This method of raising capital has gained popularity due to its flexibility, accessibility, and cost-effectiveness compared to traditional methods.

Tokenized Private Credit

Tokenized private credit refers to the representation of traditional private credit assets—such as loans, bonds, and funds—as digital tokens on a blockchain. This tokenization potentially increases efficiency and transparency, leading to faster settlements and broader access to these assets. By leveraging smart contracts and automation, tokenizing private credit enables fractionalization of loans, quicker distribution of funds for investments and payments, reduced costs, enhanced risk management, and the creation of new financial products and services.



Tokenized Private Credit

Earn real yields by investing in private credit loans to businesses, a $1.6T market in traditional finance. Credit protocols facilitate orginations, deal funding, and borrower repayments.

Active Loans Value	Total Loans Value	Current Avg. APR	Total Loans
$7,697,647,758	$12,048,058,151	9.84%	1,881

Locations of End-Borrowers

(excludes crypto trading firms and market-makers)

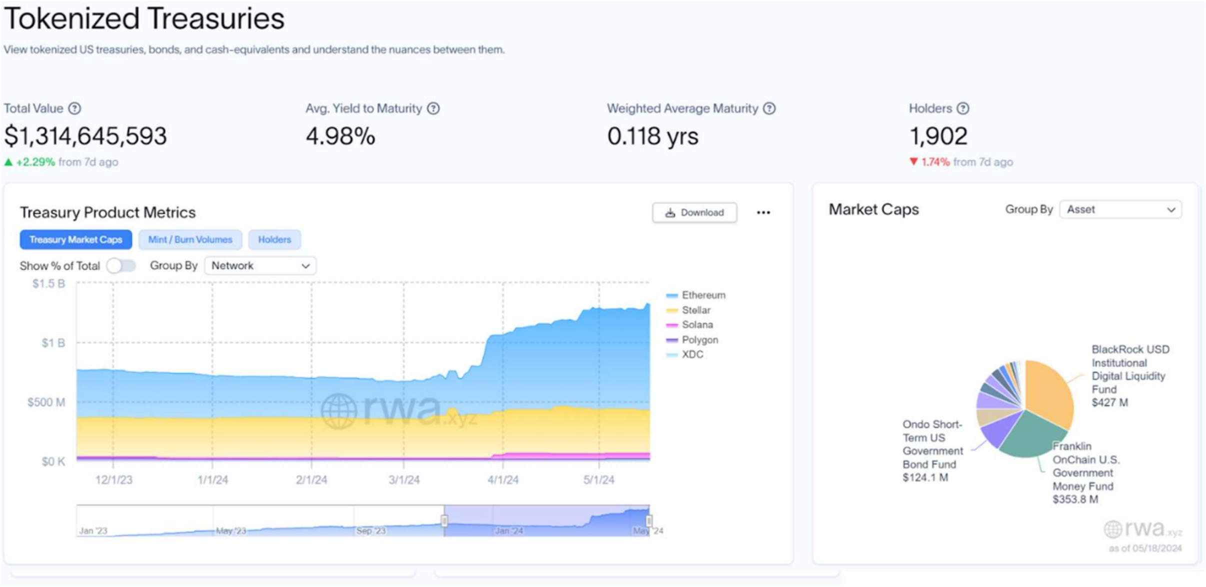
Active Loans Value by Sector

Source: https://app.rwa.xyz/ as of 18 May 2024

According to data from rwa.xyz as of May 18, 2024, the current state of tokenized private credit shows significant growth and evolution. The total value of loans issued through tokenized private credit platforms has reached $12 billion, with active loans accounting for $7.6 billion. This substantial figure demonstrates the increasing adoption and trust in tokenized private credit as a viable alternative to traditional lending methods.

Tokenized Treasuries

Tokenized treasuries are digital representations of government bonds that can be traded as tokens on the blockchain. This allows investors to access the government securities market through digital tokens, offering benefits such as increased liquidity, global participation, cost savings, transparency, and the potential for new financial products.



Tokenized Treasuries

View tokenized US treasuries, bonds, and cash-equivalents and understand the nuances between them.

Total Value: $1,314,645,593 ▲ +2.29% from 7d ago

Avg. Yield to Maturity: 4.98%

Weighted Average Maturity: 0.118 yrs

Holders: 1,902 ▼ 1.74% from 7d ago

Show % of Total ○ Group By Network

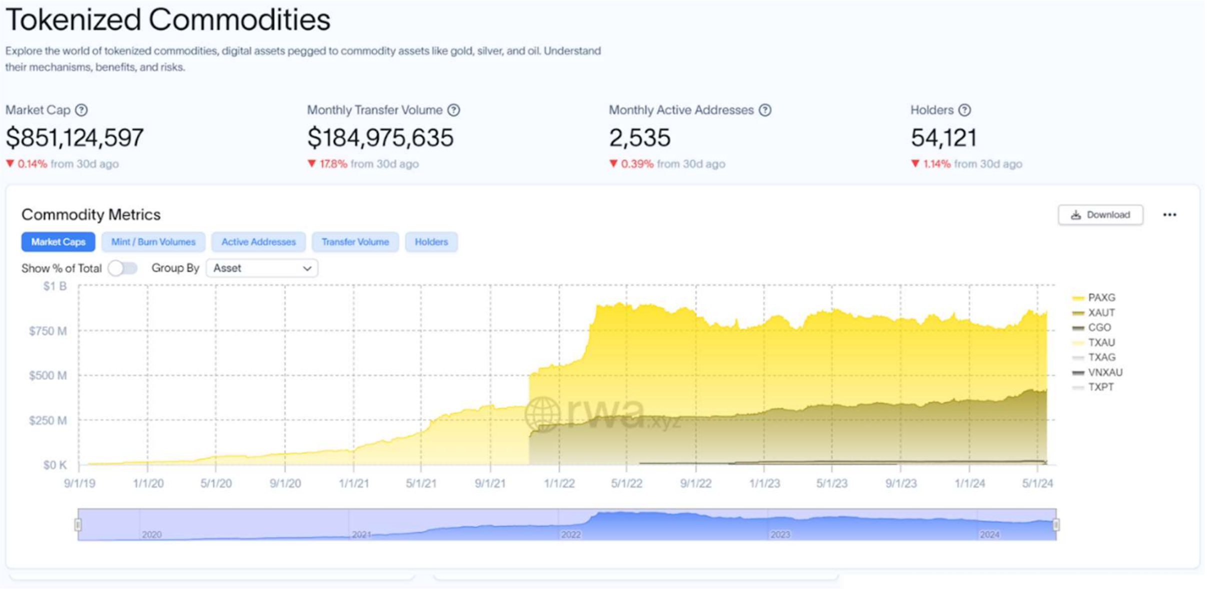
Market Caps Group By Asset

Source:https://app.rwa.xyz/ as of 18 May 2024

According to data from rwa.xyz as of May 18, 2024, the tokenized US Treasury market has experienced substantial growth, reaching nearly $1.3 billion. This growth has been largely driven by BlackRock's entry into the market earlier this year. In April 2024, BlackRock's BUIDL fund became the largest tokenized treasury fund, surpassing Franklin Templeton's offering, with $375 million in deposits after significant inflows. The fund, represented by the BUIDL token on the Ethereum network, is backed by U.S. Treasury bills, repo agreements, and cash.

Tokenized Commodities

Tokenized commodities refer to the process of representing traditional physical commodities, such as gold, oil, agricultural products, or rare earth metals, through digital tokens on a blockchain network. These tokens are backed by the actual underlying physical assets and provide ownership rights to the holders. The tokenization of commodities allows for fractional ownership, easier transferability, and improved market efficiency, making it more accessible and appealing to a broader audience.



Tokenized Commodities

Explore the world of tokenized commodities, digital assets pegged to commodity assets like gold, silver, and oil. Understand their mechanisms, benefits, and risks.

Market Cap ⓘ	Monthly Transfer Volume ⓘ	Monthly Active Addresses ⓘ	Holders ⓘ
$851,124,597	$184,975,635	2,535	54,121
▼ 0.14% from 30d ago	▼ 17.8% from 30d ago	▼ 0.39% from 30d ago	▼ 1.14% from 30d ago

Commodity Metrics

Show % of Total: ○ Group By: Asset

Source: https://app.rwa.xyz/ as of 18 May 2024

According to data from rwa.xyz as of May 18, 2024, the tokenized commodities market has reached above $850m with $184m of commodity tokens transferred over the last 30 days.

Tokenized Real Estate

Tokenized real estate covers a wide variety of investment products, including tokenized real estate funds (digital tokens representing shares in real estate investment funds) , tokenized real estate bonds (debt instruments backed by real estate assets, issued as digital tokens), and tokenized real estate development projects (tokens representing investments in specific real estate development projects).

Traditional asset fractionalization, such as REITs, has already existed for several years, focusing on equity and real estate asset classes. However, the tokenization of real estate on blockchain networks takes this concept further by enabling greater liquidity and transparency for investors while lowering investment thresholds allowing more investors to participate in real estate markets.

Citi forecasts the market size of tokenized real estate assets to reach $1.5 trillion by 2030, out of a $20 trillion total addressable market for this category alone.

Introducing the ERC-1400 Standard

ERC-1400 was created to address the lack of standardization in creating, issuing, and managing security tokens on Ethereum and EVM-compatible networks.

It has evolved into an umbrella for ERC-1410 (Partially Fungible Token Standard), ERC-1594 (Core Security Token Standard), ERC-1643 (Document Management Standard), and ERC-1644 (Controller Token Operation Standard) to improve adoption by decomposing the security token standard into a library of related and interoperable standards, making the implementation more flexible and adaptable to different use cases.

It was designed to support:

- **Compliance with Regulatory Requirements:** Ensuring that only verified and approved investors can hold and transfer a security token.
- **Transfer Restrictions:** Allowing for restrictions on token transfers to comply with securities regulations, preventing unauthorized transfers.
- **Partitioning:** Supporting the concept of partitions, allowing a single security token to represent multiple asset classes or segregate different securities tranches. This is particularly useful for managing complex financial products and structured finance.
- **Document Management:** Providing mechanisms to link off-chain documentation to the tokens, ensuring that relevant legal and financial documents are always associated with the token, facilitating transparency and compliance.
- **Compatibility:** ERC-1400 is designed to be compatible with other ERC standards, such as ERC-20 and ERC-777, allowing for interoperability with existing infrastructure and tools on the Ethereum network.
- **Event Management:** Including functions for managing events such as issuance, redemption, and transfer of securities, which are critical for regulatory reporting and audit purposes.
- **Investor Rights:** These include functionalities to enforce investor rights, such as voting and dividend distribution, directly within the smart contract.

Clarifying ERC-1400 Technical Standard Status:

Within the Ethereum ecosystem, Ethereum Request for Comments (ERCs) specify application layer standards that determine how applications running on Ethereum can interact with each other.

It is important to note that ERC-1400 is not currently an official part of the Ethereum ERC Track (https://github.com/ethereum/ERCs/tree/master/ERCS). However, its credibility is likely to be further reinforced once it completes the due process and becomes an official ERC standard rather than being referred to as such by the industry in name only. This formal recognition will solidify its position as one of the ecosystem's go-to standards for security tokens.

Currently, ERC-1400 is being hosted by the Security Token Standard organization (https://thesecuritytokenstandard.org).

ERC-1400: Modular Design

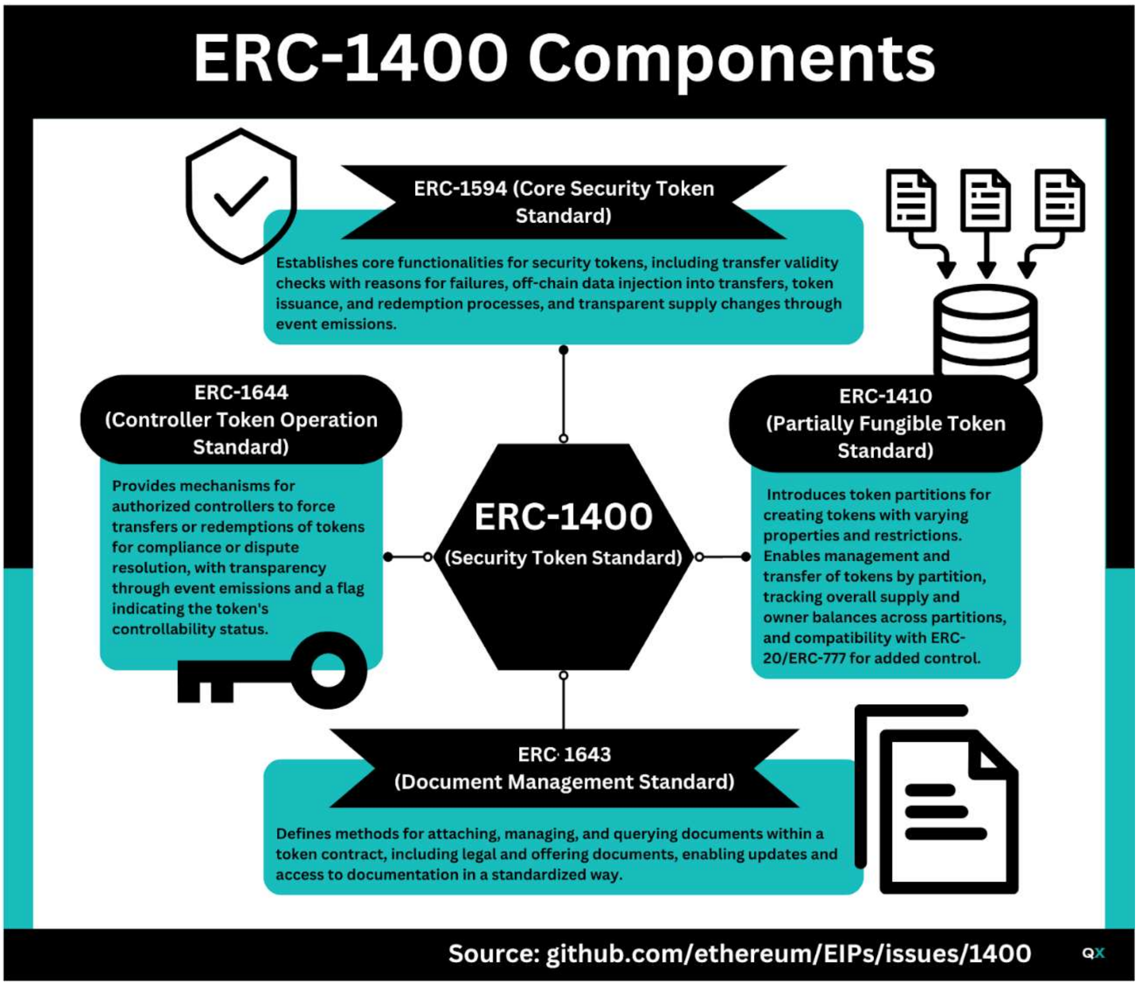
Designing a comprehensive and effective security token standard involves carefully considering various factors, including the balance between on-chain and off-chain transfer restrictions, identity management, transparency, and interoperability with existing standards and systems.

To better understand the rationale behind ERC-1400's modular design and the importance of its component standards, let's first explore a few core principles and concepts.

Interface Segregation Principle

The Interface Segregation Principle (ISP), a core tenet of software engineering, underscores the importance of creating focused and minimal interfaces to prevent unnecessary dependencies. Applied to ERC-1400, a standard divided into several sub-standards, this principle facilitates a modular and flexible approach to tokenizing financial instruments on a blockchain. By segregating functionalities into dedicated components such as ERC-1594 (Core Security Token Standard) and ERC-1410 (Partially Fungible Tokens), the ERC-1400 suite ensures that users interact only with what is relevant to their needs; this enhances the system's adaptability, maintainability, and ease of use.

Figure 3: ERC-1400 Components



ERC-1400 Components

ERC-1594 (Core Security Token Standard)

Establishes core functionalities for security tokens, including transfer validity checks with reasons for failures, off-chain data injection into transfers, token issuance, and redemption processes, and transparent supply changes through event emissions.

ERC-1644 (Controller Token Operation Standard)

Provides mechanisms for authorized controllers to force transfers or redemptions of tokens for compliance or dispute resolution, with transparency through event emissions and a flag indicating the token's controllability status.

ERC-1400 (Security Token Standard)

ERC-1410 (Partially Fungible Token Standard)

Introduces token partitions for creating tokens with varying properties and restrictions. Enables management and transfer of tokens by partition, tracking overall supply and owner balances across partitions, and compatibility with ERC-20/ERC-777 for added control.

ERC· 1643 (Document Management Standard)

Defines methods for attaching, managing, and querying documents within a token contract, including legal and offering documents, enabling updates and access to documentation in a standardized way.

Source: github.com/ethereum/EIPs/issues/1400

Segregated responsibility in token standards

A token standard must accurately model the asset's underlying structure, its holders’ positions, and the rules governing its transfers. Given the complex nature of financial instruments, it is crucial to keep a siloed approach to tokenization, and additional functionalities that are complex enough to deserve their own standard should remain segregated and not be incorporated into the token standard to avoid unnecessary complications and limitations. For example, the World Wide Web Consortium (W3C) already provides elaborate standards for decentralized identifiers, thereby indicating that identity management is a distinct domain that needs its separate handling.

ERC-1400 compliance does not require the use of all its component standards. The code structure facilitates and even necessitates customization, particularly for the component standards to function harmoniously. This is evident in certain aspects of the code, such as the '_data' component.

Identity Management

Identity management is a critical aspect of security tokens, as regulations often require knowing who holds the tokens and ensuring that only eligible investors can transact. ERC-1400 does not prescribe a specific identity management solution but provides flexibility for integrating with various identity systems. In other words, the implementer has the freedom to determine how the data is obtained, and each token can be associated with different transfer agents.

One approach is to use an on-chain identity registry that associates Ethereum addresses with verified investor identities. The token contract can then be checked against this registry to ensure that only whitelisted investors can hold or transfer tokens. Another approach is to use off-chain identity verification and provide proof of identity through the additional data parameters in token transfers.

Utilizing the _data field on the canTransfer function to provide a proof that can be used to attest the transfer activity could allow the token standard to interact with a decentralized identifier (DID) or a verifiable credential-based identity system. The proof can be in the form of a cleartext certificate or a zero-knowledge proof (ZKP) issued by a trusted/recognized authority.

Transferability Rules and Compliance

The transferability of financial instruments involves more than just verifying the recipient's identity or the transferred quantity. It encompasses:

- **Regulatory Compliance:** Adherence to laws like MiFID (UE), Securities Exchange Act (US), Financial Services Act (UK), LSFin / LEFin / LIMF (Switzerland), etc.
- **Contractual Compliance:** Observance of shareholder agreements, bylaws or term sheet of the instrument.

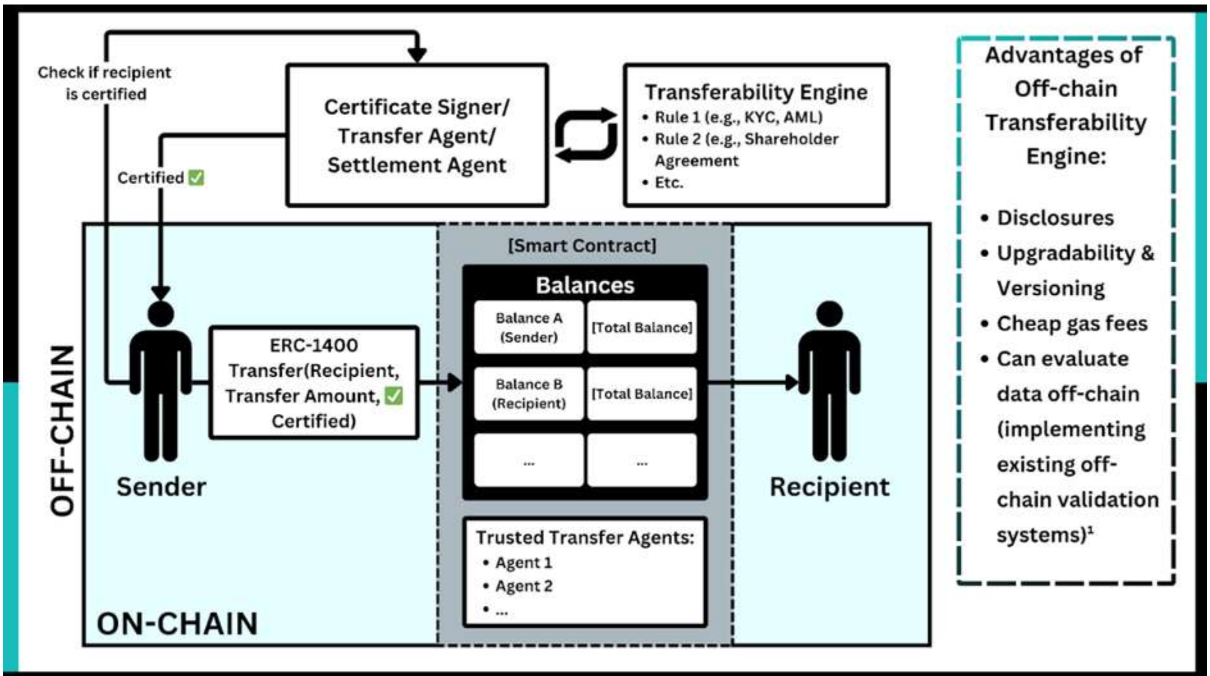
These rules can be complex, confidential, subject to change, and require versioning and asynchronous decision-making workflows, including human interventions. ERC-1594 addresses these requirements by introducing the flexible "canTransfer" checks and "transferWithData" functions, ensuring that tokens can be managed and transferred in full compliance with the law and contractual obligations. We discuss these functions in more detail in the following section.

On-chain vs Off-chain Transfer Restrictions

One key consideration in the design of ERC-1400 is the balance between on-chain and off-chain transfer restrictions. On-chain restrictions are programmatically enforced by the token's smart contract, ensuring that only compliant transfers can occur. However, some transfer restrictions may require off-chain data or input from external entities, such as regulators or KYC/AML providers.

ERC-1400 aims to address this by allowing the use of off-chain data in token transfers through the `transferWithData` and `transferFromWithData` functions. These functions accept additional data parameters that can be used to provide off-chain authorization or information required for compliance checks. The token contract can then use this data in conjunction with on-chain rules to determine the validity of a transfer.

Figure 4: ERC-1400 Compliance Off-chain Rules Evaluation


Check if recipient is certified
Certified ✅
Certificate Signer/ Transfer Agent/ Settlement Agent
Transferability Engine
• Rule 1 (e.g., KYC, AML)
• Rule 2 (e.g., Shareholder Agreement
• Etc.
Advantages of Off-chain Transferability Engine:
• Disclosures
• Upgradability & Versioning
• Cheap gas fees
• Can evaluate data off-chain (implementing existing off-chain validation systems)¹
OFF-CHAIN
[Smart Contract]
Balances
Balance A (Sender)
[Total Balance]
Balance B (Recipient)
[Total Balance]
...
...
ERC-1400 Transfer(Recipient, Transfer Amount, ✅ Certified)
Sender
Recipient
Trusted Transfer Agents:
• Agent 1
• Agent 2
• ...
ON-CHAIN

[1] Implementers can choose to evaluate data on-chain (implementing existing on-chain validation systems, such as ONCHAINID).

As illustrated in Figure (4), the token transfer process in ERC-1400 transactions involves an extra step where the token Sender requests a signed certificate from a Transfer Agent attesting that the desired transfer is permissible. This is achieved by calling the `transferWithData` function, which includes the `recipient` address, `value` (number of tokens), and a `data` parameter that holds the signed certificate.

To obtain the certificate, the Sender must first make a request to the Transfer Agent, providing details of the intended transfer. The Transfer Agent, which may be a centralized entity, contains a Transferability Engine that executes predefined transferability rules. These rules can check both on-chain state like token balances as well as incorporate off-chain data, such as verifying that the Sender and Recipient are not on a sanctions list, without the need for oracle services. If the transfer satisfies all rules, the Transfer Agent issues a signed certificate back to the Sender.

The Sender then submits the desired transfer transaction by calling `transferWithData`, passing the signed certificate in the `data` field. The ERC-1400 smart contract verifies that the certificate is validly signed by a trusted Transfer Agent and only authorizes the token transfer if the certificate is valid. This allows complex logic and constraints to be evaluated off-chain by the Transfer Agent while keeping gas costs of the actual on-chain transfer low.

Executing transferability rules off-chain provides several advantages:

- The rules can be kept private rather than being exposed on-chain
- The rules can be easily upgraded and versioned over time
- Evaluation of the rules does not consume gas, enabling lower on-chain transaction costs
- Off-chain data sources can be incorporated without oracles

It's important to note that while current implementations utilize an off-chain Transfer Agent, the ERC-1400 standard is flexible enough to allow the Certificate Signer to potentially be implemented directly on-chain as well, such as by using an on-chain identity service.

Reason Codes for Success/Failure

ERC-1400 introduces the concept of reason codes for token transfer successes and failures. The `canTransfer` and `canTransferByPartition` functions return a status code indicating whether a transfer is possible and, if not, the reason for the failure. It provides transparency to users and helps them understand why a transfer may have been rejected.

The standard defines a set of status codes, such as `INVALID_RECEIVER,` `INVALID_SENDER,` and `INSUFFICIENT_BALANCE.` These codes can be extended with application-specific codes to provide more granular information about transfer restrictions and compliance requirements.

References to Other EIPs:

ERC-1400 builds upon and interacts with several other EIPs to achieve its goals. Some of the primary EIPs referenced in the standard include:

- **ERC-20:** The basic token standard that ERC-1400 extends to maintain compatibility with existing wallets, exchanges, and tools.
- **ERC-165** is a standard for detecting the interfaces implemented by a smart contract, which ERC-1400 uses to identify compliant tokens.
- **ERC-173** is a standard for contract ownership, which ERC-1400 uses to define the roles and permissions of token owners and controllers.
- **ERC-725** is an Ethereum standard for identity management, which can be used in conjunction with ERC-1400 to verify investor identities and maintain compliance.

By leveraging these existing standards, ERC-1400 ensures interoperability and compatibility with the broader Ethereum ecosystem while introducing the necessary features and extensions for security tokens.

Consistency & Constraints

One key consideration in developing the ERC-1400 umbrella of standards was ensuring consistency in using concepts like operators across the different standards. This consistency is crucial for the standards to interoperate effectively.

During the discussions surrounding EIP-1410, the importance of maintaining a consistent approach to operators was highlighted. Operators are entities authorized to perform specific actions on behalf of token holders, such as transferring tokens. By defining and implementing operators consistently across ERC-1410, ERC-1594, ERC-1643, and ERC-1644, the ERC-1400 framework aimed to create a cohesive and interoperable set of standards.
Ensuring interoperability among these standards was essential to facilitating the adoption and implementation of security tokens on the Ethereum blockchain. A consistent and harmonized approach to key

concepts like operators would enable developers and projects to integrate and build upon the ERC-1400 standards more easily, ultimately encouraging a more robust and efficient ecosystem for security tokens.

These key considerations highlight the complexity and nuances of designing a comprehensive security token standard. ERC-1400 aims to strike a balance between the need for regulatory compliance and the benefits of blockchain technology, such as transparency, efficiency, and automation. By providing a flexible framework that can accommodate various compliance requirements and integrate with existing identity and regulatory systems, ERC-1400 seeks to facilitate the adoption and growth of security tokens on the Ethereum blockchain.

Full Tokenization and Class Differentiation

Authorities may require that a shareholder register is either entirely tokenized or not at all. Companies' shares are typically divided into classes with varying rights, making it imperative to represent these differences accurately. ERC-1400, through ERC-1410's concept of "Partially Fungible Tokens" addresses this requirement by introducing the “Partition” construct to support the tokenization of multiple share classes within a single capital structure.

Accessibility of Underlying Fundamentals

For a token to be effectively utilized in both primary and secondary markets, its attributes must be easily accessible and classifiable. This includes financial structuring attributes (e.g., maturity, interest rates) and issuer industry attributes (e.g., ESG/green credentials). ERC-1400's document management capabilities (ERC-1643) ensure that relevant documents are securely and unequivocally attached to the smart contract, providing investors with the necessary transparency and trust in the tokenized asset.

ERC-1400: Technical Breakdown

As previously mentioned, ERC-1400's modular design, composed of several interrelated sub-standards, adheres to the core software engineering principle of separation of concerns, ensuring that each component addresses unique aspects of security token management to create a comprehensive, compliant, and flexible framework.

This section will cover the technical details and functionality of each of these component standards.

ERC-1410 - Partially Fungible Token Standard: This standard introduces the concept of token partitions, allowing for the creation of partially fungible tokens with different properties and transfer restrictions. It facilitates the representation of multiple classes of assets within the same capital structure, allowing for nuanced differentiation in rights and privileges.

ERC-1594 - Core Security Token Standard: This standard defines the core functionality of security tokens, including token transfers, minting, burning, and transfer restrictions.

ERC-1643 - Document Management Standard: This standard provides a way to manage and store documents related to the security token, such as legal agreements and investor information. It enables attaching documents directly to the smart contract, ensuring unambiguous and reliable access for investors.

ERC-1644 - Controller Token Operation Standard: This standard allows for administrative actions on tokens, crucial for compliance and regulatory interventions. It defines the role of a controller, who has the ability to force token transfers and perform other administrative actions.

ERC-1400

ERC-1400 outlines a set of requirements aimed at ensuring that securities issued on a blockchain network function in accordance with regulatory expectations while providing the necessary flexibility for digital asset management. These requirements play a crucial role in defining the processes and guidelines for issuing, transferring, and managing security tokens on-chain. The implications of these requirements are as follows:

- **Transfer Validation**

Requirement: Must have a standard interface to query if a transfer would be successful and return a reason for failure.

Implication: This ensures transparency and predictability in token transfers, allowing for pre-validation of transactions to comply with regulations and custom rules.

- **Forced Transfer**

Requirement: Must be able to perform forced transfer by an authorized party for legal action or fund recovery.

Implication: Allows token issuers or designated authorities to move tokens in exceptional circumstances, adhering to legal obligations or rectifying issues.

- **Standard Events for Issuance and Redemption**

Requirement: Must emit standard events for issuance and redemption.

For issuance and redemption of tokens, the below events must be emitted as defined by ERC- 1594:

Issued	0x0e9905d62635f049c2f4e11678ebf9dc3d1f8c4a653e290759b772e47ba00d00
Redeemed	0xb7d0d6b60740753e9f16692a2f479472a1385aec2420fa43225b02f2ffa1afe7

For issuance and redemption of tokens by partition, the below events must be emitted as defined by ERC- 1410:

IssuedByPartition	0x6032e2468b0f94dfa63c61d6c8a84842a99c049263eae408a52b945020b6578d
RedeemedByPartition	0xa4f62471c9bdf88115b97203943c74c59b655913ee5ee592706d84ef53fb6be2

For transfers TransferByPartition events (ERC1410) and ERC20 Transfer events must be emitted.

Implication: Facilitates tracking of token lifecycle events, making it easier for wallets, exchanges, and other services to update user balances and display relevant activities.

- **Metadata Attachment**

Requirement: Must be able to attach metadata to a subset of a token holder's balance such as special shareholder rights or data for transfer restrictions.

Implication: Enhances token utility by allowing differentiation between tokens held by the same entity, such as tokens with different voting rights or lockup periods.

- **Metadata Modification**

Requirement: Must be able to modify metadata at the time of transfer based on off-chain data, on-chain data, and the parameters of the transfer.

Implication: Ensures token attributes can be dynamically adjusted to reflect real-world agreements, compliance requirements, or other conditions affecting token properties.

- **Documentation Updates**

Requirement: Must support querying and subscribing to updates on any relevant documentation for the security.

Implication: Keeps token holders informed about significant information relating to their holdings, like changes in terms or rights associated with the tokens.

- **Optional Signed Data for Transfers**

Requirement: May require signed data to be passed into a transfer transaction in order to validate it on-chain.

Implication: Offers an optional layer of security and compliance, enabling token transfers to include additional verifications such as KYC/AML checks.

- **Asset Class Flexibility**

Requirement: Should not restrict the range of asset classes across jurisdictions which can be represented.

Implication: Promotes the adoption of the ERC-1400 standard across different markets and for various types of assets, enhancing interoperability and market reach.

- **ERC-20 Compatibility**

Requirement: Must be ERC-20 compatible.

Implication: Ensures basic interoperability with the broader Ethereum ecosystem, including exchanges and wallets, facilitating token transfers and visibility.

- **Optional ERC-777 Compatibility**

Requirement: May be ERC-777 compatible.

Implication: Introduces advanced features such as hooks for tokens to react to being sent or received, while still optional, it offers developers flexibility in token design.

These requirements collectively aim to create a robust framework for security tokens, ensuring they can operate effectively within the existing financial ecosystem and the DeFi space, providing a reliable and compliant means of representing and managing securities on EVM-compatible networks.

ERC-1410

ERC-1410, or the Partially Fungible Token Standard, is a vital ERC-1400 Security Token Standards framework component.

ERC-1410 Overview: ERC-1410 introduces the concept of token partitions, allowing for the creation of partially fungible tokens with different properties and transfer restrictions. It provides a standard interface for organizing an owner's tokens into a set of partitions, each represented by an identifying key and a balance.

Key Features:

- **Partition Management:** ERC-1410 enables tokens to be grouped into partitions, with each partition having its own unique identifier (bytes32) and associated metadata.
- **Granular Operations:** Tokens can be operated upon at a partition level, allowing for more granular control over token transfers and management.
- **Overall Supply and Balance Tracking:** The standard maintains data about the overall token supply and the total balances of token owners across all partitions.
- **Backward Compatibility:** ERC-1410 can be combined with ERC-20 or ERC-777 to provide an additional layer of transparency and control over token behavior across different partitions.

Motivation and Rationale: The main motivation behind ERC-1410 is the need to associate metadata with individual fungible tokens or groups of tokens. This metadata can be used for various purposes, such as:

- Implementing vesting or lockup logic for a portion of a token holder's balance
- Attaching restrictions or data to tokens, which can then be used to determine transfer restrictions
- Tracking token provenance by recording previous token owners
- Modeling assets that are fungible under certain circumstances but not others (e.g., in-game credits and deposited balances)

ERC-1410 aims to enhance transparency and allows for more complex token behaviors while maintaining overall fungibility by providing a standard way to identify and operate on token partitions.

Specification Details: The ERC-1410 specification defines several key functions and events for managing partially fungible tokens:

- `balanceOf` and `balanceOfByPartition`: Functions to query the total balance of a token holder across all partitions and within a specific partition, respectively.

- `partitionsOf`: Returns an array of all partitions associated with a token holder's address.
- `transferByPartition` and `operatorTransferByPartition`: Functions for transferring tokens within a specific partition, with the ability to include additional data for compliance checks or other purposes.
- `canTransferByPartition`: Allows checking whether a transfer within a partition is possible and returns status codes and reason codes for failed transfers.
- Operator management functions: Enables setting and revoking operators for all partitions or specific partitions of a token holder.
- Issuance and redemption functions: Allows issuing and redeeming tokens within specific partitions.

The standard also defines events for tracking transfers, operator authorizations, and issuance/redemption operations at the partition level.

Implications and Potential Use Cases: ERC-1410 provides a powerful tool for creating and managing security tokens with complex compliance requirements and transfer restrictions. By enabling the association of metadata with specific partitions, it allows for granular control over token behavior and enables use cases such as:

- Enforcing lockup periods or vesting schedules for certain portions of a token holder's balance
- Implementing jurisdiction-specific compliance rules and transfer restrictions
- Facilitating the creation of partially fungible assets, or assets with multiple classes of fungible tokens, such as tokenized real estate or collectibles with varying attributes

ERC-1594

ERC-1594, the Core Security Token Standard, is a crucial component of the ERC-1400 Security Token Standards framework. This standard focuses on providing a set of core functionalities and interfaces for security tokens to implement controls in order to be compliant with regulations, including off-chain data injection, transfer validity checks, and token issuance and redemption.

ERC-1594 Overview: The primary purpose of ERC-1594 is to define a standard interface for security tokens that allows for more complex interactions between off-chain and on-chain entities. It builds upon the ERC-20 token standard and can be easily extended to support the ERC-777 standard.

Key Features:

1. **Transfer Validity Checks:** ERC-1594 introduces the `canTransfer` and `canTransferFrom` functions, which allow checking the validity of a token transfer before executing it. These functions return status codes and reason codes, providing insight into why a transfer might fail.
2. **Off-Chain Data Injection:** The standard includes `transferWithData` and `transferFromWithData` functions, which allow arbitrary data to be passed alongside token transfers. This data can be used for various purposes, such as providing signed authorizations or additional transfer details.
3. **Token Issuance:** ERC-1594 defines an `issue` function for minting new tokens and increasing the total supply, along with an `isIssuable` function for checking whether the token contract allows further issuance.
4. **Token Redemption:** The `redeem` and `redeemFrom` functions enable token holders to burn their tokens and reduce the total supply, subject to the same conditions as token transfers.
5. **Event Emissions:** The standard mandates the emission of `Issued` and `Redeemed` events whenever tokens are minted or burned, respectively, providing transparency into the token's supply changes.

Motivation and Rationale: The transfer of security tokens often involves more complex rules and restrictions than utility tokens. These restrictions can be related to factors such as token metadata, sender and receiver identities, and overall token-level constraints.

ERC-1594 aims to enable security token contracts to implement more flexible and dynamic transfer restrictions by providing a standard way to check transfer validity and injecting off-chain data into transfer functions. The `canTransfer` and `canTransferFrom` functions allow off-chain entities to verify the validity of a transfer before executing it, potentially saving gas costs on failed transfers and improving the overall user experience.

Including token issuance and redemption functions in ERC-1594 helps formalize these processes and provides greater visibility into the token's total supply and its changes over time.

Implications and Considerations: ERC-1594 provides a flexible and extensible framework for implementing security tokens on the Ethereum blockchain. Incorporating off-chain data injection and transfer validity checks enables more complex and dynamic transfer restrictions to be implemented without relying solely on-chain whitelists or static rules.

However, it's important to note that the actual implementation of these transfer restrictions and the interpretation of the injected off-chain data are left to the individual token contracts. The standard provides the necessary interfaces and functions, but the specific rules and logic must be defined within the contract itself.

Including token issuance and redemption functions in ERC-1594 helps standardize these processes and improve transparency. However, it's crucial for token issuers to carefully consider the implications of enabling or disabling further token issuance, as this can have significant effects on the token's supply and value.

ERC-1643

ERC-1643, the Document Management Standard, is another component of the ERC-1400 Security Token Standards framework. This standard focuses on providing a way to associate and manage documentation related to security tokens within the token contract itself.

ERC-1643 Overview: The primary purpose of ERC-1643 is to define a standard interface for attaching, querying, and updating documents associated with a security token contract. These documents can include legal agreements, offering documents, investor disclosures, or other relevant materials that token holders and stakeholders may need to access.

Key Features:

- **Document Attachment:** ERC-1643 allows documents to be associated with a token contract using a short name (represented as a bytes32 hash), a URI pointing to the document, and an optional hash of the document contents.
- **Document Querying:** The standard provides functions to retrieve individual documents by their name (`getDocument`) and a list of all documents attached to the contract (`getAllDocuments`).
- **Document Management:** ERC-1643 includes functions for adding or updating documents (`setDocument`) and removing documents (`removeDocument`) from the contract.

- **Event Emissions:** The standard requires that events (`DocumentUpdated` and `DocumentRemoved`) be emitted whenever a document is added, updated, or removed from the contract. It allows off-chain systems to track changes to the associated documentation.

Motivation and Rationale: Security tokens often come with various legal rights and obligations for both token holders and issuers. These rights and obligations are typically outlined in legal agreements, offering documents and other materials that must be easily accessible to all relevant parties.

By providing a standardized way to associate and manage these documents directly within the token contract, ERC-1643 aims to streamline the process of issuing and managing security tokens. It allows wallets, exchanges, and other ecosystem participants to provide a consistent and transparent view of the associated documentation. It enables token holders to receive updates on changes to these documents in a standardized manner.

Implications and Considerations: ERC-1643 provides a convenient and standardized way to manage documentation associated with security tokens. However, it's important to note that the standard itself does not enforce any restrictions on who can attach, update, or remove documents. Access control for these functions would need to be implemented within the token contract based on the specific requirements of the issuer and applicable regulations.

Additionally, while ERC-1643 allows for the association of document content hashes with the on-chain records, the actual storage of the documents themselves is expected to be handled off-chain. The standard provides flexibility regarding the URI schemes that can be used to reference these off-chain documents, allowing for various storage solutions such as IPFS for publicly shared documents or other proprietary storage solutions and access control mechanisms to be employed.

It's also worth considering the gas costs associated with storing and updating document references on-chain. While the use of short names (bytes32) and hashes helps to minimize storage requirements, contract owners and issuers should be mindful of the costs involved in managing a large number of documents.

ERC-1644

ERC-1644, the Controller Token Operation Standard, is another key ERC-1400 Security Token Standards framework component. This standard focuses on providing a transparent and standardized way for authorized controllers to perform forced transfers and redemptions of security tokens.

ERC-1644 Overview: The main purpose of ERC-1644 is to allow token issuers or their designated controllers (such as regulators or transfer agents) to retain the ability to force transfer tokens between addresses or redeem tokens from a token holder's address. These actions may be necessary to comply with legal requirements, resolve disputes, or recover funds in case of lost private keys.

Key Features:

1. **Forced Transfers:** ERC-1644 introduces the `controllerTransfer` function, allowing an authorized controller to transfer tokens between any two token holders while respecting balance restrictions and other potential transfer restrictions.
2. **Forced Redemptions:** The `controllerRedeem` function enables an authorized controller to redeem tokens from a token holder's address, subject to balance and other potential restrictions.
3. **Transparency:** The standard requires that all controller transfers and redemptions emit corresponding events (`ControllerTransfer` and `ControllerRedemption`) to ensure transparency and auditability of these actions.
4. **Controllability Flag:** ERC-1644 includes an `isControllable` function that indicates whether the token contract allows controller transfers and redemptions. If this function returns `false`, the token contract must always return `false` in the future and revert any attempts to perform controller transfers or redemptions.

Motivation and Rationale: Security tokens are subject to various legal and regulatory requirements that may necessitate the ability of authorized entities to force the transfer or redeem tokens in specific situations. These situations can include:

- Resolving legal disputes or enforcing court orders
- Recovering funds in case of lost or compromised private keys
- Complying with other jurisdictional or regulatory requirements

By providing a standardized way to perform these actions, ERC-1644 aims to streamline the management of security tokens and ensure transparency in the process. The standard also allows token contracts to explicitly declare whether they support controller actions, giving token holders and other stakeholders clarity on the token's controllability.

Implications and Considerations: While ERC-1644 provides a mechanism for complying with legal and regulatory requirements, it's important to consider the potential centralization risks associated with controller actions. Authorized entities' ability to force transfer or redeem tokens may conflict with the decentralized nature of blockchain technology.

To mitigate these concerns, ERC-1644 emphasizes transparency by requiring the emission of events for all controller actions and clearly indicating a token's controllability through the `isControllable` function. It allows token holders and other stakeholders to make informed decisions and assess the risks associated with a particular security token.

It's also worth noting that the specifics of controller authorization and the criteria for performing forced transfers or redemptions are not defined within the ERC-1644 standard itself. These aspects need to be addressed through additional legal and regulatory frameworks and the specific implementation of the token contract.

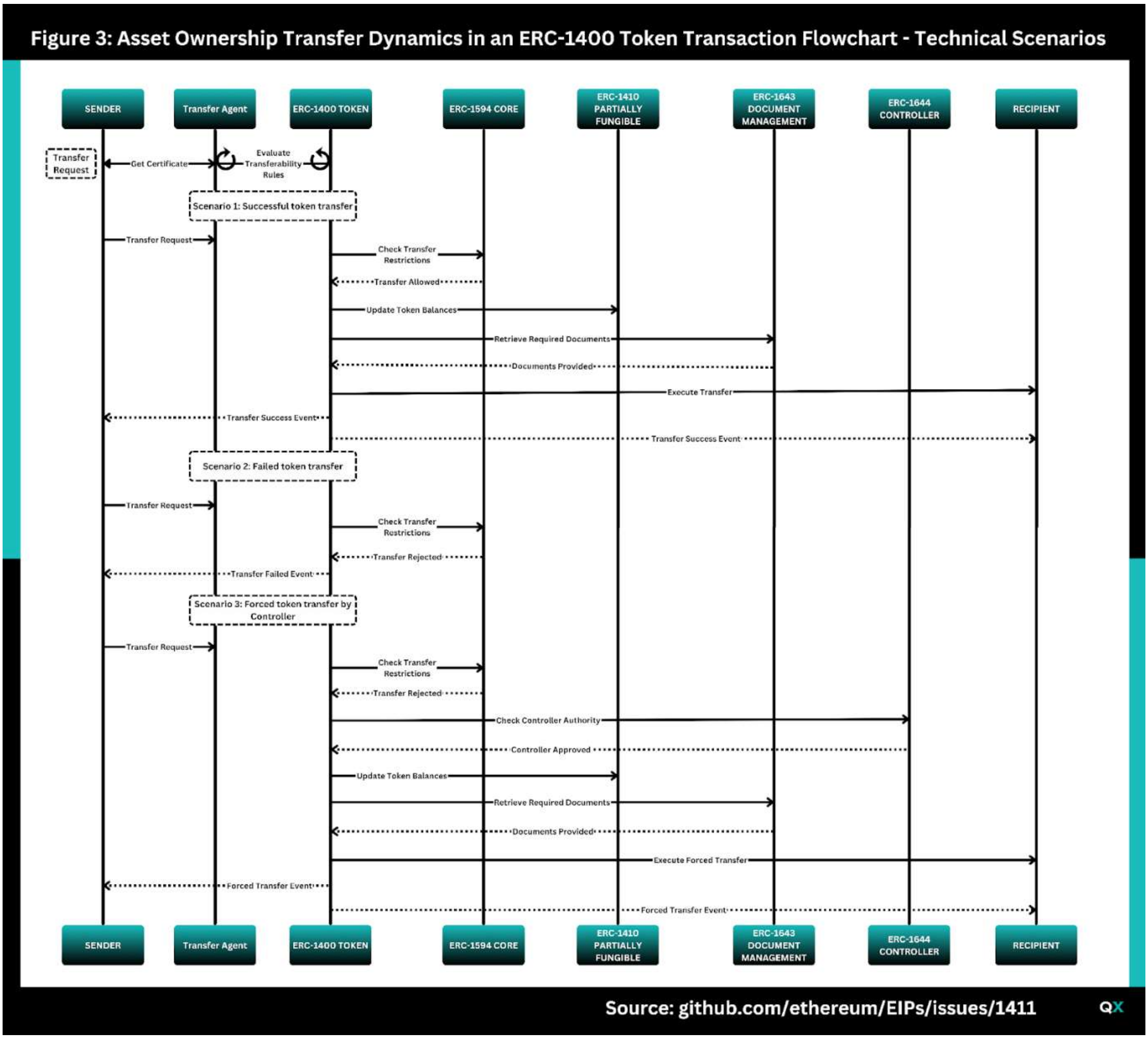
Transacting in the ERC-1400 Framework

This section describes the process of transferring asset ownership using ERC-1400 security tokens, highlighting different technical scenarios and the interactions between various components and entities involved.

The diagram below demonstrates the flows and interactions required for asset ownership transfers using ERC-1400 security tokens, covering successful transfers, failed transfers, and forced transfers by a controller. It highlights the roles of various components in ensuring compliance, managing identity, and handling document management throughout the token transfer process.

Figure 5: Asset Ownership Transfer Flowchart

Figure 3: Asset Ownership Transfer Dynamics in an ERC-1400 Token Transaction Flowchart - Technical Scenarios



Source: github.com/ethereum/EIPs/issues/1411

Scenario 1: Successful Token Transfer

1. **Transfer Request:** The sender initiates a transfer request.
2. **Get Certificate:** The sender requests a certificate from the Transfer Agent to validate the transfer.
3. **Evaluate Transferability Rules:** The Transfer Agent evaluates the transferability rules to ensure compliance.
4. **Certificate Issued:** If the rules are met, the Transfer Agent issues a certificate.
5. **Transfer Request:** The sender submits the transfer request along with the certificate.
6. **Check Transfer Restrictions:** ERC-1400 Token checks the transfer restrictions using ERC-1594 Core functionalities.
7. **Transfer Allowed:** If all checks pass, the transfer is allowed.
8. **Update Token Balances:** Token balances are updated to reflect the transfer.
9. **Retrieve Required Documents:** ERC-1594 Core retrieves any required documents from ERC-1643 Document Management.
10. **Documents Provided:** Necessary documents are provided.
11. **Execute Transfer:** The token transfer is executed.
12. **Transfer Success Event:** A transfer success event is generated, and the recipient receives the tokens.

Scenario 2: Failed Token Transfer

1. **Transfer Request:** The sender initiates a transfer request.
2. **Get Certificate:** The sender requests a certificate from the Transfer Agent.
3. **Evaluate Transferability Rules:** The Transfer Agent evaluates the transferability rules.
4. **Transfer Rejected:** If the rules are not met, the Transfer Agent rejects the transfer.
5. **Transfer Failed Event:** A transfer failed event is generated, indicating the transfer was not successful.

Scenario 3: Forced Token Transfer by Controller

1. **Transfer Request:** The sender initiates a transfer request.
2. **Check Transfer Restrictions:** ERC-1400 Token checks the transfer restrictions using ERC-1594 Core functionalities.
3. **Transfer Rejected:** The transfer is initially rejected based on standard checks.

4. **Check Controller Authority:** ERC-1400 Token checks if a controller has the authority to enforce the transfer.
5. **Controller Approved:** If the controller's authority is verified, the controller approves the transfer.
6. **Update Token Balances:** Token balances are updated accordingly.
7. **Retrieve Required Documents:** ERC-1594 Core retrieves any required documents from ERC-1643 Document Management.
8. **Documents Provided:** Necessary documents are provided.
9. **Execute Forced Transfer:** The forced transfer is executed by ERC-1644 Controller.
10. **Forced Transfer Event**: A forced transfer event is generated, and the recipient receives the tokens.

ERC-1400 Adoption and Ongoing Developments

Since its proposal by Polymath in 2018, the ERC-1400 token standard has gained traction and adoption across the industry. In recent years, there has not been a dedicated group focusing on the continued development and promotion of this standard. However, key industry stakeholders have recognized the importance of ERC-1400 and discussions are underway about the potential establishment of an ERC-1400 association. This initiative would aim to rejuvenate interest and support for the standard, guiding it through the final steps of formal recognition. The association would also plan to play a crucial role in expanding education, awareness, and development capacity within the ERC-1400 ecosystem. Such efforts are deemed essential for ensuring that this standard can fully realize its potential in streamlining and securing digital asset transactions, thereby offering tangible benefits to regulators, market participants, and financial services companies alike as well as contributing to the maturation of the digital assets and real world asset tokenization space.

Understanding ERC-1400 Adoption

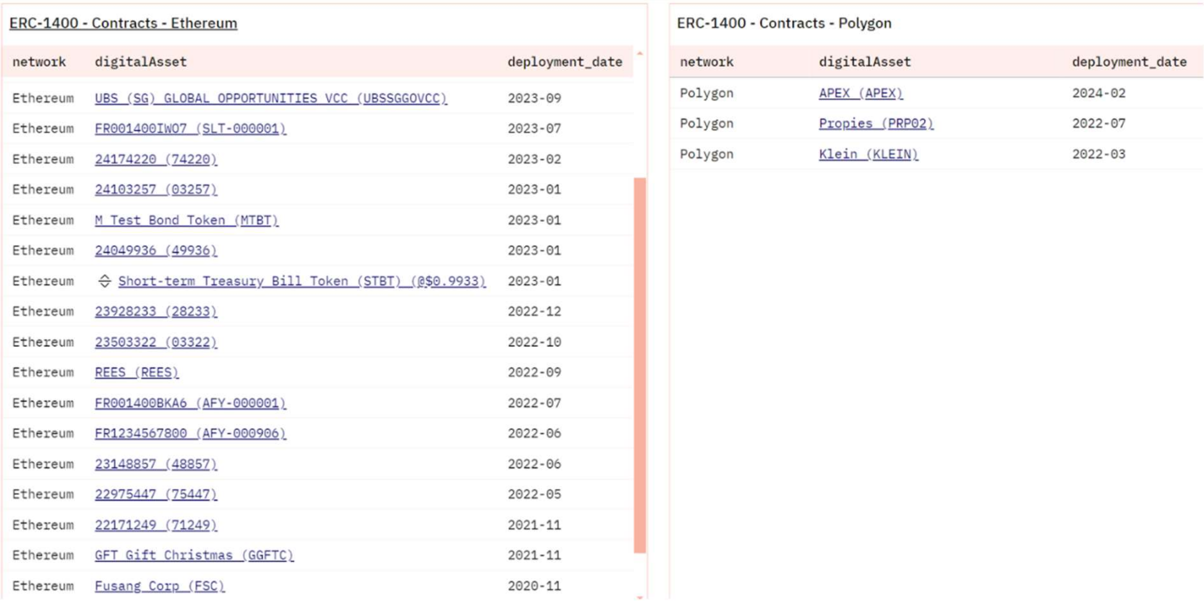
ERC-1400 has been adopted by multiple firms because it is an independent protocol which is not tied to any specific vendor or vendor ecosystem. Several companies have chosen to use ERC-1400 for their security token offerings, rather than other proposed standards that may be used by single infrastructure providers.

Private Networks

A notable challenge in gauging the adoption of the ERC-1400 standard across the industry lies in the fact that our research indicates that a substantial number of its deployments occur within private permissioned blockchain networks. Unlike public permissionless blockchain networks, where transactions and smart contract deployments are transparent and can be monitored by anyone, private networks restrict access to authorized entities only. This limitation inherently makes it difficult to track the full extent of ERC-1400's implementation and usage. Many organizations opt for private networks to leverage the benefits of blockchain technology, such as security and immutability, while maintaining control over their data and adhering to regulatory requirements. Consequently, while there is evidence of the standard's adoption through publicly available information and confirmed known implementations, the true scale of its deployment outside of public networks remains less visible, underscoring a gap in the overall visibility of the standard's adoption and its impact within the broader real world asset tokenization and digital assets space.

Public Networks

As of 18 May 2024, our analysis of the existing ERC-1400 deployments reveals the creation of 44 assets across two public permissionless networks: Ethereum and Polygon.



ERC-1400 - Contracts - Ethereum

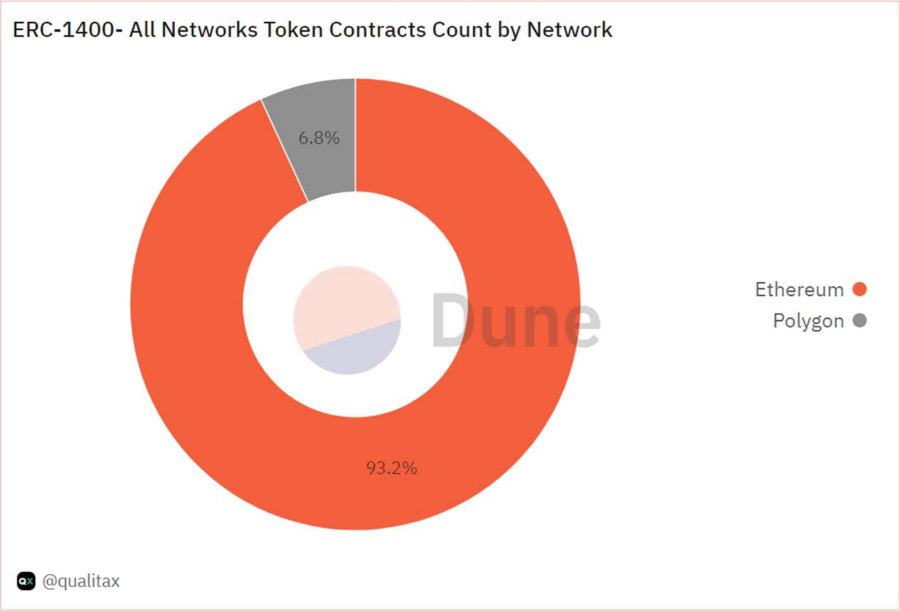
network	digitalAsset	deployment_date
Ethereum	UBS (SG) GLOBAL OPPORTUNITIES VCC (UBSSGGOVCC)	2023-09
Ethereum	FR001400IW07 (SLT-000001)	2023-07
Ethereum	24174220 (74220)	2023-02
Ethereum	24103257 (03257)	2023-01
Ethereum	M Test Bond Token (MTBT)	2023-01
Ethereum	24049936 (49936)	2023-01
Ethereum	⇳ Short-term Treasury Bill Token (STBT) (@$0.9933)	2023-01
Ethereum	23928233 (28233)	2022-12
Ethereum	23503322 (03322)	2022-10
Ethereum	REES (REES)	2022-09
Ethereum	FR001400BKA6 (AFY-000001)	2022-07
Ethereum	FR1234567800 (AFY-000906)	2022-06
Ethereum	23148857 (48857)	2022-06
Ethereum	22975447 (75447)	2022-05
Ethereum	22171249 (71249)	2021-11
Ethereum	GFT Gift Christmas (GGFTC)	2021-11
Ethereum	Fusang Corp (FSC)	2020-11

ERC-1400 - Contracts - Polygon

network	digitalAsset	deployment_date
Polygon	APEX (APEX)	2024-02
Polygon	Propies (PRP02)	2022-07
Polygon	Klein (KLEIN)	2022-03

ERC-1400 Contracts Snapshot - Source: https://dune.com/qualitax/erc-1400-analytics/

Currently, the percentage distribution is as follow - Ethereum 93.2%, Polygon 6.8%.



Source: https://dune.com/qualitax/erc-1400-analytics/

ERC-1400 Deployment Costs

When choosing a blockchain network for deploying ERC-1400 contracts, it is important to consider the potential benefits and trade-offs associated with each option.

Private permissioned networks often have fixed, minimal, or even zero gas fees, as the economic incentive for miners or validators is not required in the same way as in public networks. This results in more predictable and lower deployment costs.

However, deployment costs on public permissionless networks are highly variable and depend on the network's current gas prices, which fluctuate based on demand. This can result in significant cost differences depending on network congestion. Also, the features and flexibility of ERC-1400 mean that even on the same network, deployments may have widely different costs. Use cases with more complex implementations, as well as larger contracts, would cost more to deploy and to operate due to higher gas usage.

In private equity for example, there could be complex shareholder agreements to implement and it would not be recommended to put them on-chain due to cost and confidentiality. In that case, transactions fees would be cheaper as transferability rules are evaluated off-chain.

ERC-1400 Contemporaries

To develop a broad view of the landscape ERC-1400 occupies, it is important to acknowledge and explore other token standards that have emerged alongside ERC-1400, each aiming to address specific challenges and requirements associated with (though not always directly addressing) tokenized securities.

ERC-3643

ERC-3643 is a token standard for security tokens on the Ethereum blockchain and EVM-compatible networks. It provides a framework for issuing, transferring, and managing security tokens while ensuring compliance with regulatory requirements. For a comprehensive review of the ERC-3643 standard, refer to the report "Demystifying ERC-3643: A Deep Dive Into Compliant RWA Tokenization" published in March 2024 and accessible here: https://www.qualitax.io/erc3643.

ERC-3643 emphasizes compliance through embedded transfer rules in tokens and rigorous identity verification using ERC-734 and ERC-735 standards and via identity solutions such as ONCHAINID (ONCHAINID is an open source, decentralized identity system for compliant digital assets).

The ERC-3643 standard achieved 'Final' status in its Ethereum Improvement Proposal (EIP) in December 2023, becoming the first standard specifically tailored for compliant tokenization to reach this stage.

ERC-1404

ERC-1404 is a security token standard developed to facilitate regulatory compliance for tokenized securities. It is built upon the ERC-20 standard, adding two additional functions: detectTransferRestriction and messageForTransferRestriction. The detectTransferRestriction function allows issuers to enforce transfer restrictions, such as checking if the recipient is whitelisted or if the sender's tokens are frozen. The messageForTransferRestriction function provides a human-readable explanation of the restriction codes returned by detectTransferRestriction.

While both ERC-1400 and ERC-1404 are designed for security tokens, ERC-1400 offers a more comprehensive framework with features for partitioned transfers and extensive compliance mechanisms. In contrast, ERC-1404 provides a simpler approach, focusing on basic transfer restrictions and ease of use for issuers needing straightforward compliance features.

ERC-7092

ERC-7092 defines a standard for tokenized financial bonds on the Ethereum blockchain and EVM-compatible networks. The standard incorporates essential bond properties, such as issue date, maturity date, coupon rate, principal, and currency, and includes features for callable, puttable, and convertible bonds.

ERC-7092 aims to simplify the process of issuing, trading, and redeeming bonds by providing a unified framework for their characteristics and functionalities. While this standard has reached final status, there are currently no production-level implementations.

ERC-2222

ERC-2222 augments the ERC-20 token framework by incorporating a Funds Distribution Standard, facilitating the proportional distribution of funds to token holders. This standard is particularly relevant for tokens representing dividend-bearing assets, automating the distribution process to reflect the true ownership of the underlying assets.

Proposed as an EIP in 2019, ERC-2222 continues to be refined through collaborative efforts on GitHub. The standard contributes to the ecosystem’s ongoing initiatives to formalize the tokenization of RWAs, aiming to address both regulatory compliance and investor assurance concerns within the blockchain-based financial sector.

ERC-4626

ERC-4626 is a token standard introduced for the Ethereum blockchain, aimed at providing a uniform API for tokenized vaults that accrue yield on a single underlying asset. Termed as the Tokenized Vault Standard, it seeks to streamline the technical aspects of such financial instruments.

This standard expands upon the ERC-20 framework, offering a structure through which users can derive earnings from their investments. It can also be combined with ERC-3643 to create vaults for ERC-3643 tokens. The standard serves as a foundation for developers, offering a consistent framework for constructing contracts that handle yield-bearing assets. It aims to reduce the complexity of developing these applications, facilitating easier integration and broadening the accessibility to yield-generating opportunities.

ERC-1155

ERC-1155 is a token standard in the EVM-based ecosystem that allows for creating fungible and non-fungible tokens within a single smart contract. It is a multi-token standard that combines the functionality of previous standards like ERC-20 and ERC-721. ERC-1155 enables the efficient transfer of fungible and non-fungible tokens within a single contract, thus reducing the transaction costs and the complexity of deploying and managing multiple contracts for each new token needed by the system.

It has several unique features, such as support for an infinite number of tokens, semi-fungible tokens, safe transfer functions, batch transfer and approvals, and metadata storage capabilities.

ERC-1155 has use cases in various applications, such as gaming, creator monetization, digital art and collectibles, and tokenized real-world assets. All major NFT marketplaces, including OpenSea and Rarible, have adopted ERC-1155. However, it does not take steps beyond ERC-721 to address security token considerations.

The differentiating factor between ERC-1155 and ERC-1400 is that although ERC-1155 allows creation of multiple ERC-20 tokens from a single contract, it does not support the ability to impose different transfer restrictions for each token type as in the case of Token Partitions introduced by ERC-1400.

ERC-6960

ERC-6960 introduces a dual-layer classification system where the main ID represents the primary asset type, and sub IDs represent unique attributes or variations of the main asset. It has been designed to address limitations of previous standards like ERC-20 (fungible tokens), ERC-721 (non-fungible tokens), and ERC-1155 (multi-token standard). One of its goals is to facilitate fractional ownership which a useful feature for tokenizing real-world assets (RWAs) like real estate, commodities, and securities, allowing these assets to be divided into smaller, tradable units. Primarily ERC-6960 is aimed at asset tokenization, not specifically designed with regulatory compliance features. While ERC-1400 has been designed with regulatory compliance in mind, includes features like partitioned transfers, compliance checks, and whitelist/blacklist functionalities to meet legal requirements for security tokens.

The Dual Layer Token standard was proposed to the Ethereum community in April 2023, with ongoing development and discourse on GitHub as it progresses through the EIP process.

CMTAT

The Capital Markets and Technology Association (CMTA) standard token for securities (CMTAT) is a digital token framework that enables the creation of "ledger-based securities" in compliance with Swiss law. CMTAT is designed to enhance regulatory compliance within the Ethereum ecosystem by building on the ERC-20 standard and introducing mechanisms for identity verification, anti-money laundering (AML) protocols, and Know Your Customer (KYC) compliance directly within the token transfer framework. The standard is licensed under the permissive Mozilla Public License 2.0 (MPL 2.0).

CMTAT aims to streamline the compliance process for tokenized assets, making it an essential tool for issuers and investors dealing with securities, real estate, and other regulated financial products.

This standard is particularly significant for projects seeking to navigate the complex regulatory landscape of tokenized assets, providing a clear pathway for compliance with local and international regulations. As of May 2024, CMTAT continues to be refined and actively developed by the CMTA.

Selected Case Studies

In this section, we present two real-world case studies that showcase the practical applications and current implementations of ERC-1400.

Case Study 1: Sustain Harvests' Club Deal Marketplace

Sustain Harvests is a Club Deal on an ethics-focused mission to help ESG-driven small and medium-size companies strive by raising funds (€500k to €10M) and harmonizing the relationships between company owners, investors and partners. Approached through a model to anticipate the "hyperspace of danger" (Kervern, Diamond, Tainter), AML risks are countered in the context of investor and investment protection to regenerate the economy and biodiversity.

To foster efficiency, transparency and operational cost reduction along the way, as well as cope with the growing demand for seamless and personalized experiences, the investment phase and full investment asset management lifecycle have been digitized into an online marketplace. This platform was built over 5.5 years and is comprehensive, flexible, robust, secure, data-privacy protective, and scalable. The entire onboarding processes are in place: KYC / AML verifications, subscription contracts & signature, payment & reconciliation, etc. Digitization is maximized thanks to tokenization, which enables for optimal automation and cost reduction possibilities.

For every capital funding operation, Sustain Harvests creates an ad hoc investment vehicle (SPV) and designs the best-suited fundraising instrument (equity shares, convertible bonds…), which is then proposed through a private sale on the marketplace to both love-money investors and traditional qualified investors in the form of a digital security token - or Real-World Asset (RWA). Regulatory compliance being critical, extensive KYC and AML procedures are executed using API-accessed automated services of a reputable KYC-provider.

Sustain Harvests' mission goes beyond the issuance and primary market of fundraising assets: tokenization enables the provision of real-time cap table tracking, the platform automatically calculates dividends or coupons, and most importantly it helps derisk the investments. Indeed, investors looking to liquidate their assets over secondary trading can access a bulletin board to find OTC buyers, and secondary trades are considerably simplified by not only automating the KYC / AML compliance checks, but also the compliance with contracts in place such as a shareholders' agreement giving preemptive rights to certain individuals over trades. To enable this, the platform integrates a transferability rule engine capable of handling any rules, from simple KYC checks or integration of transfer agents, to complex and confidential shareholders' agreement clauses. This is all possible specifically thanks to the use of ERC-1400.

Commendable ESG projects are now being boosted by Sustain Harvests' services. One such project, NAT5, is a blockchain-powered marketplace that simplifies and makes the biodiversity market more transparent and accessible to buyers and sellers to push forward nature-based projects of high - non green-washing - quality.

Overview	Sustain Harvests is a club deal which uses tokenization to simplify and reduce the cost of investment and asset lifecycle management, and uses ERC-1400 specifically to accept all real-life compliance constraints, thereby empowering secondary trading possibilities however complex the transferability rules may be.
Target Customers	ESG-oriented small and medium-size companies, and both love-money and qualified investors
Technology	ERC-1400 Talium Assets tokenization marketplace Onfido KYC provider Polygon
Benefits	Simpler access to operations for all investors. Simpler asset lifecycle management. Lower operational costs. Derisking through secondary trading possibilities. Any transferability rule, simple or complex. Versioned rules, modifiable on the fly. Confidential rules remain confidential.
Product Type	Digital asset issuance marketplace
Launch Date	September 2023 (launch of Sustain Harvests' whitelabel instance of the marketplace, developed since 2018)

Production Phase	September 2023 (the whitelabel marketplace is in production since 2019)
Notable partnerships	Talium, with its Talium Assets whitelabel tokenization marketplace
Milestones/ Achievements	6- and 7-digit tokenization and fundraise operations with confirmed drastic simplification of onboarding processes, acceptance by stakeholders and significant streamlining and operational cost reduction.
Next steps	Same, multiplied. Ongoing preparations

Commentary and Key Takeaways

Using ERC-1400 is necessary when compliance requirements go beyond simple KYC whitelisting checks and the use of transfer agents. As soon as shareholders' agreement clauses are involved, handling transferability off-chain (e.g., using a transferability rule engine such as the one provided by the Talium Assets tokenization platform) is necessary to enable confidential rules, asynchronous ones and modify them dynamically at nearly no cost without compromising the smart contract’s integrity. ERC-1400 is the ERC format suited for these real-life scenarios.

Case Study 2: Matrixdock Short-term Treasury Bill (STBT) Token

Matrixdock, a digital assets platform owned by Matrixport, has developed a tokenized U.S. Treasury Bill (T-Bill) offering called STBT (Short-term Treasury Bill Token), which is built on the ERC-1400 token standard. STBT allows stablecoin holders to gain exposure to risk-free U.S. Treasury yields on-chain while ensuring compliance, transparency, and cross-chain capabilities. The token is fully backed by short-term U.S. Treasury securities and reverse repurchase agreements.

STBT is 1:1 pegged with USD, meaning that holding 1 STBT is equivalent to holding 1 USD net-asset value (NAV) of short-term U.S. treasuries, plus receiving daily interest, via a rebasing mechanism on-chain.

Overview	A USD-pegged, on-chain investment vehicle for accredited investors, backed by short-term US Treasury securities with 6-month maturities and reverse repurchase agreements.
Target Customers	Accredited investors seeking returns from US treasury securities.
Technology	• ERC-1400 security token standard • Integration of Chainlink Proof of Reserve (PoR) for on-chain verification of asset collateralization. • Integration of Chainlink Cross-Chain Interoperability Protocol (CCIP) for secure cross-chain transfers of STBT.
Benefits	• 'Risk-free' returns from US treasury securities • Daily interest rebased on-chain • A 1:1 peg to USD
Product Type	Tokenized short-term US Treasury securities with 6-month maturities and reverse repurchase agreements.
Launch Date	February 2023
Production Phase	Active
Notable partnerships	Chainlink
Milestones/ Achievements	• "Ecosystem Excellence" Award at the 4th Annual Assets and Digitized Securities (TADS) Awards. • Total Distributed Interest: $4.6M (as per 18th May 2024)
Next steps	• Regularly publish assessment reports that account for the validity of STBT's underpinning assets and are intended to serve as an additional layer of validation.

	- Launch more RWA tokens.

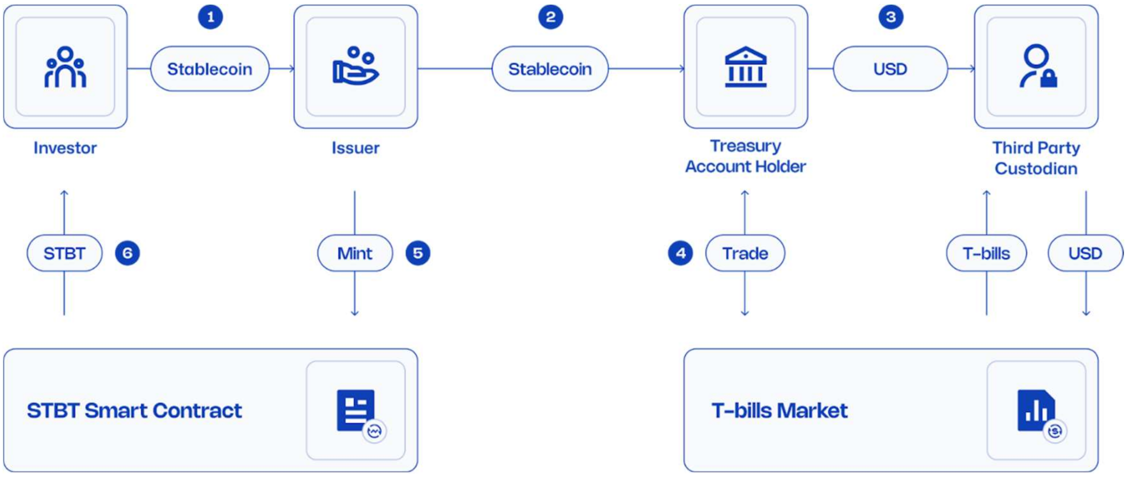
Commentary and Key Takeaways:

Matrixdock's STBT showcases how the ERC-1400 token standard can be utilized to create secure, compliant, and transparent tokenized RWAs. By leveraging ERC-1400, Matrixdock has access to a range of features that enhance the overall offering and user experience.

The ability to implement ERC-1400 features like whitelisting, transfer restrictions, and controller operations ensures compliance and suits the regulated nature of the tokenized offering. In addition, the ERC-1643 sub-standard, which is part of ERC-1400, allows STBT to associate relevant documentation with the token contract. This feature enhances transparency by providing on-chain access to legal agreements and asset collateralization records.

As a permissioned token, STBT requires whitelisting of contracts to facilitate seamless integration and operation within DeFi ecosystems such as Curve. This has led to some initial hurdles, such as the need for Matrixdock to whitelist Curve contracts to allow the withdrawal of admin fees and conversion to 3CRV.

However, once these issues were resolved, STBT was successfully paired with 3CRV in a stableswap pool on Curve, providing liquidity and yield opportunities for STBT holders. The pool utilizes a modified 3CRV metapool implementation to support STBT's positive rebasing mechanism. STBT's integration with Curve highlights the potential for tokenized real-world assets to be incorporated into DeFi platforms, offering investors enhanced liquidity and access to new yield opportunities.



STBT Mint Flow - Source: https://stbt.matrixdock.com/

MEDIA CREDITS

Frontpage:

Photo by Hert Niks on Unsplash

Source: https://unsplash.com/photos/city-skyline-across-body-of-water-during-daytime-ZT6XDAGOtEw

Description: City skyline across body of water during daytime.

Page 14:

Photo by Simone Hutsch on Unsplash

Source: https://unsplash.com/photos/white-painted-building-in-landscape-photography-roD-khbnLLI

Description: White painted building in landscape photography.

Page 23:

Photo by Tânia Mousinho on Unsplash

Source: https://unsplash.com/photos/white-concrete-building-under-white-sky-during-daytime--ZAzOJoExl8

Description: White concrete building under white sky during daytime.

Page 38:

Photo by Simone Hutsch on Unsplash

Source: https://unsplash.com/photos/gray-and-white-digital-wallpaper-g9-R1PprzVw

Description: Gray and white digital wallpaper.

Page 42:

Photo by Clique Images on Unsplash

Source: https://unsplash.com/photos/high-rise-gray-concrete-building-during-daytime-QrpQ2M0Mb1U

Description: High-rise gray concrete building during daytime.

Page 47:

Photo by Kyle Bushnell on Unsplash

Source: https://unsplash.com/photos/green-and-black-glass-windows-HbcFhbdYfR0

Description: Green and black glass windows.

QX

www.qualitax.io

contact@qualitax.io

QualitaX.io is owned and operated by Consianimis Consulting Ltd.

A private limited company registered in England and Wales under registration number 09006129.

Registered address: 167-169 Great
Portland Street, 5th Floor, London,
England, W1W 5PF, UK.