Dynamo Protocol - Secure Decentralized Finance for Real World Assets Through Zero Knowledge Homomorphic Privacy Preserving Smart Contract and Rollup Blockchain

DynamoProtocol Team

contact@DynamoFinance.tech

GitHub: https://github.com/DynamoProtocol

Abstract

Dynamo Protocol provides innovative, fast, and efficient execution of decentralized financial transactions for real world assets on blockchain. Dynamo Protocol achieves efficient and fast execution through a rollup blockchain architecture. In addition, Dynamo Protocol provides users with privacy preserving smart contract capability through homomorphic encryption. Privacy preserving smart contracts removes one of the major concerns by financial institutions and enables the wide adoption of decentralized financial transations for real world assets.

Dynamo Protocol is composed of a rollup blockchain infrastructure and privacy preserving smart contract system that is fully compatible with the Ethereum Virtual Machine. The Dynamo Rollup Blockchain provides a theoretical transaction throughput of over 40,000 TPS, and thus fully capable of real world applications. The Dynamo Privacy Preserving Smart Contract system offers full EVM compatibility and portability from existing Ethereum ecosystem.

Figure 1: Dynamo = Rollup Blockchain + Privacy Preserving Smart Contracts for Real World Assets

Contents

List of Figures

1 Introduction

Real-world assets (RWAs) in blockchain represent digital tokens that mirror tangible and conventional financial assets, such as currencies, commodities, equities, and bonds. Real-world asset (RWA) tokenization emerges as a significant market opportunity within the blockchain industry, potentially encompassing a market size in the hundreds of trillions of dollars. In essence, virtually any valuable asset can undergo tokenization and be incorporated into blockchain networks.

As evidenced by the increasing number of projects, the tokenization of RWAs is a burgeoning market segment in the digital asset industry. These projects aim to tokenize a diverse array of assets, including cash, commodities, real estate, and other assets of value. The market size and data on real-world asset (RWA) tokenization suggest significant growth and potential within the blockchain industry.

1.1 Market Demand for Real World Asset Tokenization

As of December 2023, the market size for RWA tokenization is estimated to be in the hundreds of trillions of dollars. This estimation includes various asset classes such as currencies, commodities, equities, and bonds. The increasing adoption of blockchain technology and the demand for decentralized finance (DeFi) solutions contribute to the expansion of this market segment.

Several key data points illustrate the growth and trajectory of RWA tokenization:

Growing Market Opportunities: RWA tokenization presents one of the largest market opportunities in the blockchain industry. The ability to digitize and tokenize real-world assets opens up new avenues for investment and liquidity in traditionally illiquid markets.

Increased Investor Interest: Institutional and retail investors are showing growing interest in tokenized real-world assets due to their potential for diversification, transparency, and accessibility. This interest is driving the development of platforms and projects focused on RWA tokenization.

Diverse Asset Classes: RWA tokenization encompasses a wide range of asset classes beyond traditional financial instruments. These include real estate, art, intellectual property, and more. This diversity expands the potential market size and addresses various investor needs and preferences.

Technological Advancements: Advancements in blockchain technology, smart contracts, and decentralized finance protocols are facilitating the tokenization process and making it more efficient and secure. These technological developments further drive the growth of RWA tokenization.

Tokenized real-world assets (RWAs) are digital tokens created on blockchain technology, representing physical and traditional financial assets like cash, commodities, equities, bonds, credit, artwork, and intellectual property. The tokenization of RWAs signifies a significant shift in how these assets are accessed, exchanged, and managed, unlocking a myriad of new opportunities for both blockchain-powered financial services and various non-financial applications supported by cryptography and decentralized consensus mechanisms.

Asset tokenization is regarded as one of the most promising applications of blockchain technology, with its potential market size encompassing nearly all economic activities. The future of finance is expected to feature numerous blockchains supporting trillions of dollars of tokenized RWAs on a common substrate comprising blockchain and distributed ledger technology-based networks connected through a universal cross chain standard.

1.2 Process of Real World Asset Tokenization

The process of tokenizing real-world assets involves representing ownership rights of assets as on-chain tokens. Through this process, a digital representation of the underlying asset is created, allowing for on-chain management of the asset's ownership rights and bridging the gap between physical and digital assets.

Tokenized assets offer several advantages over traditional assets, including enhanced liquidity, increased accessibility, transparent on-chain management, and reduced transactional friction. For financial assets, tokenization consolidates distribution, trading, clearing, settlement, and safekeeping processes into a single layer, resulting in a more streamlined on-chain financial system with reduced counterparty risk and more efficient capital mobilization.

The process of tokenizing real-world assets typically involves several steps:

Asset Selection: Identifying the real-world asset to be tokenized. Token Specifications: Determining the type of token (fungible or non-fungible), the token standard to be used (e.g., ERC20 or ERC721), and other fundamental aspects of the token.

Blockchain Selection: Choosing the appropriate public or private blockchain network on which to issue the tokens. Optimistic and Zero Knowledge based Rollup blockchains offer execution speed and throughput for RWA transactions similar to that of traditional financial networks.

Off-chain Connection: Most tokenized assets require high-quality off-chain data from secure and reliable sources such as oracles. Employing verification services like reserve proofs is essential for ensuring transparency regarding the assets backing the RWA tokens.

Issuance: Deploying smart contracts on the selected network, minting the tokens, and making them available for use.

1.3 Advantages of Real World Asset Tokenization

Tokenized real-world assets (RWAs) hold the potential to significantly alter the landscape of decentralized finance (DeFi). While DeFi has demonstrated the potential of on-chain finance as a superior technological layer for facilitating financial and economic activities, the vast majority of assets remain outside the blockchain ecosystem. Yet, these assets could benefit from the advantages offered by blockchain technology. This underscores the importance of tokenized RWAs in expanding the digital asset industry by enabling assets that are currently outside the blockchain ecosystem to be utilized within blockchain rails.

Enabling assets that are currently outside the digital asset ecosystem to become blockchain-enabled will create a financial system with improved liquidity conditions, greater transparency, reduced systemic risks, and conflict-of-interest-free infrastructure. This fosters a more equitable environment where a select few cannot exploit the system for personal gain.

The segment of tokenized real-world assets has seen significant growth within the DeFi ecosystem, with the total value locked in RWA reaching approximately $5 billion in December 2023, according to DefiLlama.

Real-world assets also hold the potential to enable the creation of novel financial products. For instance, MakerDAO, one of the largest DeFi protocols by total value locked, utilizes various real-world asset collateral to back the stablecoin DAI. This represents an innovative approach to creating new financial assets by leveraging both traditional and blockchain-based assets and technology.

Tokenized real-world assets offer several advantages, including:

Liquidity: By enabling globally accessible liquidity conditions on a unified substrate—the blockchain ecosystem with cross-chain activity supported by Chainlink CCIP—tokenized RWAs enhance market liquidity for traditionally illiquid assets.

Transparency: Tokenized assets are represented on-chain, ensuring transparency and auditable asset management. This decreases overall systemic risks, as the amount of leverage and risk in the entire system can be more accurately determined.

Accessibility: Tokenized RWAs can broaden the potential user base of certain asset types by enabling easier access through blockchain-based applications. This allows a broader set of users to utilize assets that would otherwise be unavailable to them through fractional ownership.

However, tokenized RWAs also present risks, primarily related to the custody of physical assets, which must be reliably managed, and the connection to the outside world. Additionally, there is a risk of smart contract bugs and vulnerabilities. Moreover, issuing an asset is not sufficient; there must also be good market liquidity or demand for it in order for it to thrive.

2 Decentralized Finance of Real World Assets - RWA DeFi

The integration of real-world assets (RWAs) by DeFi protocols can mitigate the cyclical nature of on-chain yields by leveraging the inverse relationship in demand between crypto-native assets and off-chain assets. The following chart illustrates this dynamic by comparing U.S. treasury rates against stablecoin yields:

From December 2020 to approximately April 2022, stablecoin yields remained relatively high while Treasury rates were low. However, from April 2022 to late November 2023, as the average rate for 1-month to 3-year U.S. Treasury Bills increased, on-chain yields declined and consistently remained below T-Bill rates.

The integration of RWAs on-chain enables users to benefit from real-world interest rates when they offer more attractive yields than crypto-native sources, and vice versa when on-chain rates are more favorable. RWAs have the potential to optimize on-chain yields and create a more stable rate environment, reducing cyclicality in DeFi protocol usage, yields, and revenues.

Additionally, RWAs can enhance the stability, efficiency, and functionality of existing DeFi protocols. For instance, DeFi-native yield curve relying on tokenized bonds provides users with a new method to price risk on-chain and manage their DeFi portfolios. It also enables the protocol to raise funds through debt auctions similar to off-chain entities. Successful implementation and widespread adoption of RWAs among existing DeFi protocols can offer DeFi users exposure to more mature traditional financial products.

2.1 Challenges

While RWA tokenization and DeFi services of RWA bring about unprecedented efficiency and convenience to the financial industry and the overall economy, challenges exist to the wider adoption of RWA tokenization and DeFi at a larger scale and at a quicker speed.

One of the most prominent hindrance to the wider adoption of RWA tokenization and DeFi is privacy of transaction data. Blockchain based smart contract transactions are inherently open and available to public inspection. While this provides advantages such as transparency and accountability, it causes potential issues, such as front-running a large pending transaction by blockchain validators and inability to preserve privacy of transactions. Dynamo Protocol is designed to mitigate these challenges and enable wider adoption of RWA tokenization and DeFi.

3 Dynamo Protocol

Dynamo Protocol aims to provide solutions to the challenges faced by RWA DeFi by making available a privacy preserving smart contract system and rollup blockchain to the community.

Homomorphic encryption based privacy preserving smart contract system offers privacy of transaction data while preserving all the benefits of smart contract based financial transactions. Rollup blockchain offers high transaction speed and throughput suitable for real world financial transactions. [2]

3.1 Homomorphic Encryption and Privacy Preserving Smart Contract

Ensuring data privacy is a critical concern for smart contracts handling sensitive information. Dynamo Protocol adopts the ZeeStar system, a language and compiler that allows non-experts to create private smart contracts and perform operations on external data. The ZeeStar language enables developers to specify privacy constraints conveniently using zkay's privacy annotations. The ZeeStar compiler then guarantees the realization of these constraints by combining non-interactive zero-knowledge proofs and additively homomorphic encryption. ZeeStar is practical, as it prepares transactions for our contracts in at most 54.7 seconds, at an average cost of 339,000 gas.[2]

3.2 Rollup Blockchain

Practical usage of RWA DeFi requires user experiences similar to that of traditional financial transactions. To achieve such an end, Dynamo Protocol employs a rollup blockchain architecture compatible

with the EVM smart contract ecosystem. The Dynamo Rollup Blockchain is built on top of Arbitrum Rollup Stack, which has been proven in real world financial transactions. [1]

Armed with two key innovative features, privacy preserving smart contract based on homomorphic encryption and EVM equivalent rollup blockchain, Dynamo Protocol brings privacy and speed to real world asset transactions, and provides user experiences meeting real world expectations. [1]

4 Technical Implementation

Overview Dynamo Protocol consists of two major components:

1. Dynamo Privacy Smart Contact system - DPSC
2. Dynamo Rollup Blockchain - DRB

Dynamo Privacy Smart Contact system - DPSC is built on ZeeStar. DPSC consists of an expressive language to specify and a compiler to automatically enforce data privacy for smart contracts. DPSC not only supports homomorphic addition, but also multiplication for most combinations of owners. This allows expressing complex applications such as oblivious transfer. Furthermore, DPSC can mix homomorphic and non-homomorphic encryption schemes and is provably private.[2]

Dynamo Rollup Blockchain - DRB is a rollup based public blockchain, and can be deployed either as a layer 2 or layer 3 blockchain depending on how transactions are settled. DRB utilizes the Arbitrum technical stack customized and optimized for Real World Asset based financial transactions. Dynamo Rollup Blockchain utilizes the DYMO ERC20 token as form of payment for transaction fees. [1]

4.1 Dynamo Privacy Smart Contact with Homomorphic Encryption

A. Non-interactive Zero-knowledge Proofs A non-interactive zero-knowledge (NIZK) proof enables a prover to convince a verifier that she possesses a secret without disclosing the secret itself. Specifically, she can demonstrate knowledge of a secret witness w that satisfies a given predicate $\phi(w;x)$ for some public value x, without revealing any information about w other than the fact that $\phi(w;x)$ holds. Here, ϕ is referred to as the proof circuit, w is the private input, and x is the public input.

For instance, in a cyclic group G with generator g and $h \in G$, one can prove knowledge of the discrete logarithm z of h with respect to base g using the proof circuit $\phi(z;h)$, which is satisfied if and only if $g^z = h$.

Zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs) are a type of generic NIZK proof construction that supports any arithmetic circuit ϕ and offers constant-cost proof verification proportional to the size of ϕ (plus a typically negligible linear cost in the size of x). Due to their efficient verification costs, zk-SNARKs are commonly utilized on the Ethereum blockchain.[2]

B. Additively Homomorphic Encryption An additively homomorphic encryption scheme enables the addition of plaintexts corresponding to a pair of ciphertexts without requiring knowledge of private keys. Formally, let pk_α and sk_α be the public and private keys of a party α, respectively, and $Enc(x, pk_\alpha, r)$ represent the encryption of plaintext x under pk_α using randomness r. This scheme is additively homomorphic if there exists a function $\oplus$ on ciphertexts such that for all x, y, α, r, r_0:

$$Enc(x, pk_\alpha, r) \oplus Enc(y, pk_\alpha, r_0) = Enc(x+y, pk_\alpha, r_{00})$$

for some r_{00}, where $\oplus$ can be efficiently evaluated without knowledge of sk_α. It's important to note that both arguments to $\oplus$ must be encrypted under the same public key. Typically, additively homomorphic schemes also allow the homomorphic evaluation of subtraction using a function defined analogously.

For instance, the Paillier encryption scheme is additively homomorphic in Z_n (i.e., addition in Eq. (1) is modulo n) for an RSA modulus n, and exponential ElGamal encryption over a group G is additively homomorphic in $Z_{|G|}$, where $|G|$ is the order of G (see App. B).[2]

Privacy Annotations and Types. To facilitate precise and user-friendly specification of privacy constraints, ZeeStar utilizes privacy annotations inspired by zkay. These annotations track ownership of values within a privacy type system: Data types τ (such as integers and booleans) are extended to types of the form $\tau@\alpha$, where α determines the owner of the expression. The value of an expression can only be accessed by its owner. The owner α may be "all" (indicating the value is public) or an expression of type address. Expressions with owner "me" are referred to as self-owned, while those with owner $\alpha \notin \{me, all\}$ are considered foreign.

To prevent implicit information leaks, private expressions with owner α cannot be directly assigned to variables with a different owner $\alpha_0 \neq \alpha$. Instead, developers can use the $reveal(e,a)$ function to explicitly disclose a self-owned expression e to another owner a.

It's important to note that the privacy annotations entail minimal overhead compared to existing non-private smart contract languages such as Solidity. As discussed further, privacy is automatically enforced by ZeeStar's compiler, eliminating the need for developers to manually instantiate cryptographic primitives.[2]

Compilation. ZeeStar compiles the input contract into an executable Ethereum contract that enforces the specified privacy constraints.[2]

In the output contract, values with an owner $\alpha \neq$ "all" are encrypted under the public key of α using an additively homomorphic encryption scheme. Private expressions are precomputed locally (off-chain) by the sender and only published on the blockchain (on-chain) in encrypted form. Expressions revealed to all are additionally published in plaintext.[2]

In essence, any expression involving only public and self-owned variables is computed by the sender as follows: First, decrypt any private input variables. Then, evaluate the expression using the plaintext arguments. Finally, if the expression is private, encrypt the result using the owner's public key.[2]

Leveraging Homomorphic Encryption. As the encryption scheme used by ZeeStar is additively homomorphic, it also permits the evaluation of expressions. First, the sender re-encrypts the plaintext value val under the public key of to to obtain a ciphertext c. Then, the sender computes bal. In the proof circuit ϕ, ZeeStar ensures that c is computed correctly. Interestingly, the operation $\oplus$ is also evaluated within the proof circuit. While not necessary for privacy, this practice leads to reduced on-chain costs. Additionally, as we will discuss shortly, it allows for greater expressivity.[2]

After constructing ϕ, ZeeStar inserts a proof verification statement into the output contract. When calling the transfer function, the sender must generate and provide a NIZK proof for the circuit ϕ as a function argument proof. The public arguments of ϕ are provided as arguments to verify. If verification fails, the transaction is rejected, and the contract state is reverted.[2]

4.2 Dynamo Rollup Blockchain

The Dynamo Rollup Blockchain is constructed on the Arbitrum technology stack, which addresses limitations commonly found in layer 1 smart contract systems. Arbitrum introduces a novel approach to overcome these limitations.[1]

Arbitrum contracts are highly cost-effective for verifiers to handle. When participants act in line with incentives, Arbitrum verifiers only need to verify a small number of digital signatures for each contract. Even in cases where parties deviate from their incentives, Arbitrum verifiers can efficiently resolve disputes regarding contract behavior without needing to inspect more than a single instruction execution by the contract.[1]

Additionally, Arbitrum enables contracts to execute privately, disclosing only hashed versions of contract states. Dynamo utilizes the Arbitrum technology stack as the foundation of its Rollup Blockchain. Moreover, the Dynamo Protocol customizes and optimizes the Arbitrum stack for real-world asset operations.[1]

Dynamo Rollup Blockchain is further optimized for RWA DeFi by providing customized middle-ware modules and precompiled smart contacts, including sub-block time data API. [1]

4.3 Dynamo Protocol - Privacy and High Performance for RWA DeFi

Armed with privacy preserving smart contract and rollup blockchain, Dynamo Protocol solves two of the major hurdles hindering the wide adoption of decentralized finance transactions for real world tokenized assets, data privacy and transaction speed and throughput.

Privacy preserving smart contracts are usually computationally intensive. This translates into high gas consumption and fees in blockchain ecosystem. Dynamo Rollup Blockchain solves these drawbacks with a highly efficient and low cost EVM compatible ledger.

Together with privacy preserving smart contract and rollup blockchain, Dynamo Protocol offers community a viable and robust solution to bring tokenized real world assets and related DeFi transactions to an industry scale.

5 Dynamo Protocol Economic System and Governance

Dynamo Rollup Blockchain utilizes the ERC20 token, DynamoProtocolToken DYMO, as a payment method for transaction fees. A portion or all of the transaction fees may be distributed to verifiers of transactions. A portion of the transactions fees may also be burned based on community decision.

Community decisions are based on votes by community members. Community members may vote in proportion of the number of DynamoProtocolToken owned for a proposal.

6 Conclusion

Dynamo Protocol is a high performance, low cost, and privacy preserving DeFi system optimized for tokenized real world assets. Through innovative homomorphic encryption and rollup blockchain technologies, Dynamo Protocol provides the necessary technological foundation for the wide adoption of tokenized real world assets and related decentralized financial transactions.

6.1 Future Work

We plan to further strengthen the Dynamo system by focusing on the following areas:

- Production grade privacy preserving smart contract templates
- Decentralized spot and derivatives exchanges tailored towards RWA
- Real world data integration services

6.2 Acknowledgements

We would like to acknowledge 1) ETH and ZeeStar for providing the foundation for Dynamo Privacy Preserving Smart Contract system. 2) Arbitrum for providing the foundation for Dynamo Rollup Blockchain.

6.3 Whitepaper Versions

- Dynamo v. 1.0 – Jan. 2024, initial release

6.4 Code Base

Codebase: `https://github.com/DynamoProtocol`

References

[1] Xiaoqi Chen S. Matthew Weinberg Edward W. Felten Harry Kalodner, Steven Goldfeder. Arbitrum: Scalable, private smart contracts. `https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-kalodner.pdf`, 2018.

[2] ROGER BAUMGARTNER MARTIN VECHEV SAMUEL STEFFEN, BENJAMIN BICHSEL. Zeestar: Private smart contracts by homomorphic encryption and zero-knowledge proofs. `https://www.sri.inf.ethz.ch/publications/steffen2022zeestar`, 2022.