TNA Core

A Standardization Framework for Bitcoin's Data Availability

Abstract

The introduction of scalability solutions such as BTC Layer-2s, along with new assets on the Bitcoin mainnet, has necessitated robust data availability (DA) solutions to maintain the integrity of assets and contracts across various networks. TNA, as a naming protocol for Bitcoin, is introducing a new approach to Data Availability (DA) for the Bitcoin network – the TNA Core, which aims to facilitate secure and efficient data interactions across diverse networks, thereby enhancing the scalability and operability of Bitcoin's evolving ecosystem.

1. Background

The evolution of Bitcoin's ecosystem has spurred the emergence of diverse Layer-2 (L2) solutions and novel assets such as Taproot Assets and Inscription Assets. While these advancements address the pressing demand for scalability, they concurrently introduce challenges related to data integrity and accessibility within an increasingly intricate and segmented environment.

In response, various Data Availability (DA) solutions have been proposed within the Bitcoin network. One approach involves employing sidechains, which leverage scripts from the Bitcoin mainnet and contracts from Layer-2 to facilitate functionalities like cross-chain interactions and staking. Rollups represent another approach, where computations and state changes are processed off-chain, with proofs or updated data (inscriptions) submitted to the mainnet for verification.

Some approaches entail the implementation of separate DA layers for data verification with commitments sent to blockchain. Another solution involves the use of Bitcoin scripts for logic gates and virtual machine opcodes, and data is structured into a Merkle Tree (MAST), ensuring transaction integrity and verifiability.

The diverse array of Layer-2 protocols and DA approaches, however, leads to a non-uniform ecosystem. This diversity underscores the need for standardized asset and contract data submission, verification, and proof across networks. Trust and security in multi-network data interactions, along with economic considerations, become critical when aiming for seamless asset circulation and smart contract functionality across different blockchain environments.

The Bitcoin ecosystem, with its multiple Layer-2 solutions and DA schemes, still requires a mechanism to allow assets to circulate between various ecosystems or chains efficiently. A set of standard practices for the submission, proof, and verification of multi-network asset and contract data can address this necessity, striking a balance between trustlessness, security, and economic feasibility. This standardization is where a naming protocol like TNA can play a significant role, not just by providing clear and consistent identifiers for assets across networks, but also by potentially serving as a linchpin for a more unified DA approach within Bitcoin's multi-layered architecture.

2. Introduction

TNA Core focuses on establishing a robust framework to standardize data verification across various layers of the Bitcoin network to enhance consistency across universes, indexers, and decentralized storage systems.

The proposed solution aims to synchronize asset statuses across networks efficiently, preventing issues like double spending or insufficient transactions. This is particularly relevant when multiple Layer-2 chains support transactions for the same Bitcoin mainnet assets. For instance, when two Layer-2 chains both support transactions for a specific token like $ORDI (BRC-20), the mechanism ensures that users cannot initiate transactions for $ORDI simultaneously on both Layer-2 chains, thus preventing potential conflicts.

To address the challenge of high costs associated with verifying global data, we have designed a data sampling verification mechanism. This approach balances efficiency and security in the verification process. Additionally, we provide decentralized data export for verification, allowing third-party networks to independently challenge the verification process.

Besides synchronizing asset statuses across networks, the solution further endeavors to enable Layer-2 applications to access a broader ecosystem of Layer 1 assets, creating more utilities for Layer 1 assets. This facilitates a paradigm where assets are issued on Layer 1 and transactions are executed on Layer-2, leveraging the advantages of decentralization and security on Layer 1, along with the faster transaction speed and lower fees on Layer-2.

TNA Core further enhances the capabilities of TNA names (“Tapnames”), allowing users to conduct transactions seamlessly with a single name across various networks. This blurs the boundaries between different chains, simplifying transactions and asset management for the users.

3. Overview

The technological routes and overall architectures of existing DA solutions vary. However, they all need to address key issues such as L2 transaction packaging, proof generation, storage, and L1 verification. Since Bitcoin can only offer limited data storage space in the script field of the transaction data structure and the witness data area, storage space is limited and costly for L2 transaction data. Therefore, it is necessary to design a decentralized storage architecture that can store large amounts of data, is scalable to meet the growing data demands, and can efficiently index data for retrieval.

For various assets issued on the Bitcoin mainnet, including those based on UTXO and Witness (e.g., various BRC, ARC assets) as well as assets issued on Taproot (Tapnames, etc.), since their storage methods and data structures are very diverse, to achieve a unified Bitcoin mainnet asset verification and parsing, it is also necessary to construct a standard mainnet data asset protocol. This will allow these assets to be more conveniently applied on the Bitcoin mainnet and L2.

The main problem the protocol addresses is the existence check and authenticity verification of transaction data, achieving optimality in computational efficiency and storage space. The core components of the protocol consist of a Hierarchical Data Verification service (“HDV”) and BLOB-based storage.

The HDV protocol is composed of three fundamental processes: KZG commitment generation, Sub-Merkle Tree construction, and Block Number Allocation Table (“BNAT”) creation. KZG commitment generation involves aggregating the transaction data generated by L2 over a period, partitioning it by block number range, and generating fixed-size data blocks, referred to as “BLOBs”. Each BLOB undergoes a polynomial based KZG commitment algorithm to produce data verification information, leading to the creation of a KZG root. For each transaction in the data block, a corresponding Sub-Merkle tree is constructed to serve the rapid confirmation of the transaction's existence. To locate blocks and transactions more quickly within a data block, the upper and lower bounds of the block numbers contained in each block are generated along with the KZG root to form the BNAT. This allows for rapid location of the data block containing the desired transaction hash or block number, accelerating the verification process. Leveraging the advantages of rapid KZG commitment verification and low computational complexity, it allows the prover to efficiently compute and submit proofs. Furthermore, the generated proof data is more concise, saving transmission time and costs between prover and verifier.

For the unified verification and parsing of various assets issued on the Bitcoin mainnet, the protocol has built the Bitcoin Universal Asset Gateway (BUAG). BUAG extracts the original data storing these assets, retrieves its key metadata (such as asset ID, mapping target ID/Address, etc.), and generates a standard and rapidly searchable and verifiable KZG commitment and Merkle structure data. These data are then stored in the form of BLOBs in an off chain decentralized storage system and made available to users through RPC API and smart contract interfaces.

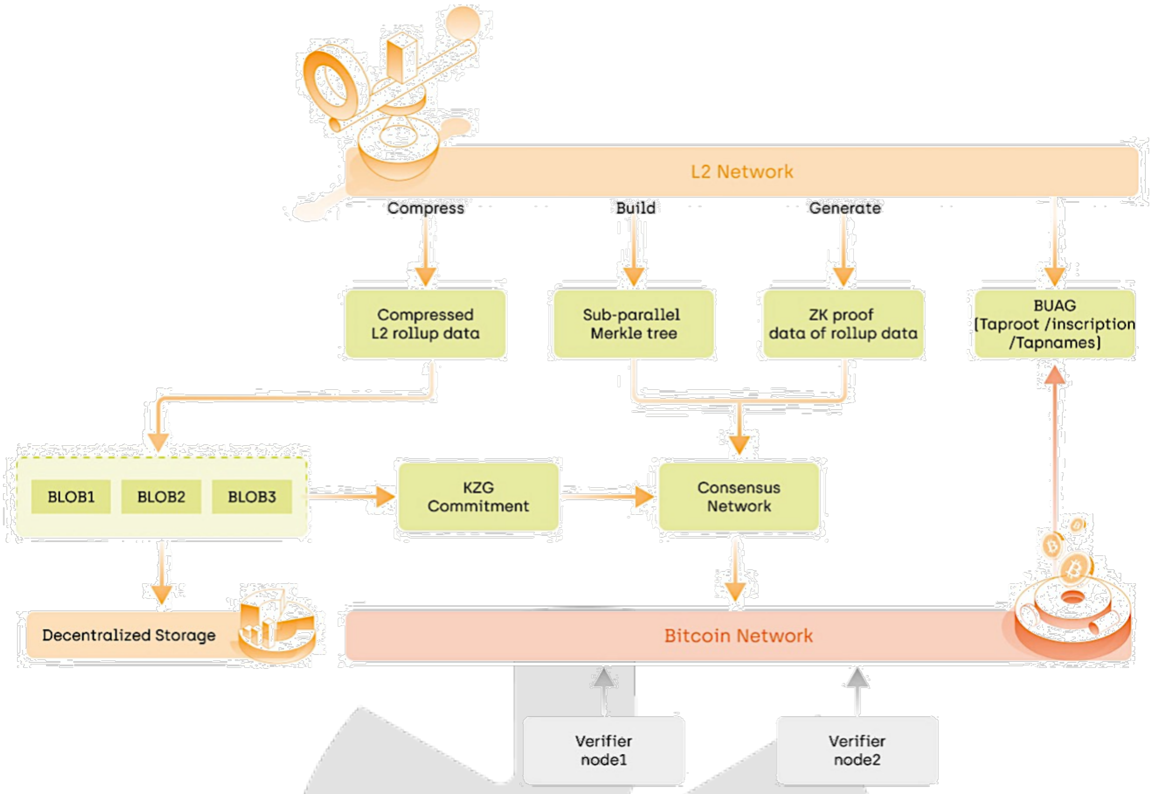

FIG. 1. OVERVIEW OF TNA CORE

3.1 BLOB-BASED STORAGE

BLOB-based storage is divided into two parts: verification summary data storage and original transaction data storage. Verification summary data storage includes KZG commitment and KZG root, Sub-Merkle Tree root hash, all of which are stored within the transaction data structure of the Bitcoin mainnet. Due to the large volume of original transaction data, which cannot be stored in the Bitcoin mainnet's transaction data structure, one method is to store it in the form of BLOBs in a dedicated decentralized storage system. Alternatively, the data is not additionally stored, and the L2 chain's block and transaction data are taken as the standard. The former method requires additional data transfer and storage costs, but the advantage is that the original transaction data can be directly read for verification. The latter method does not require additional data transfer and storage costs, but when original transaction data is needed, it necessitates a request to the corresponding chain's data service interface or synchronization with a full node, which can reduce verification efficiency.

3.2 Node Consensus Based on VRF and Schnorr

For node consensus, classical PBFT algorithms achieve consensus among nodes through a series of operations, including pre-prepare, prepare, and commit. The scalability of PBFT consensus algorithms decreases sharply as the number of nodes (i.e., verifiers) increases. This is because each node requires voting results from other nodes, resulting in communication overhead escalating to O(N^2). To adapt to and meet the scalability of the number of verifying nodes, we introduce Verifiable Random Functions (VRF) to select a smaller subset of verifying nodes from the initial node set in an unpredictable and unmanipulable manner, thereby reducing communication overhead between nodes. At the same time, we introduce the widely utilized Schnorr signature algorithm to aggregate signature verification results from the subset of verifying nodes to further reduce verification overhead and improve overall verification efficiency. Based on VRF and Schnorr, compared to classical PBFT consensus, the broadcast rounds of communication overhead are reduced from O(N^2) to O(Log(N)), further decreasing to O(Log(Ns)), where N represents the number of verifier nodes, while Ns represents the number of verifier node subsets after VRF filtering.

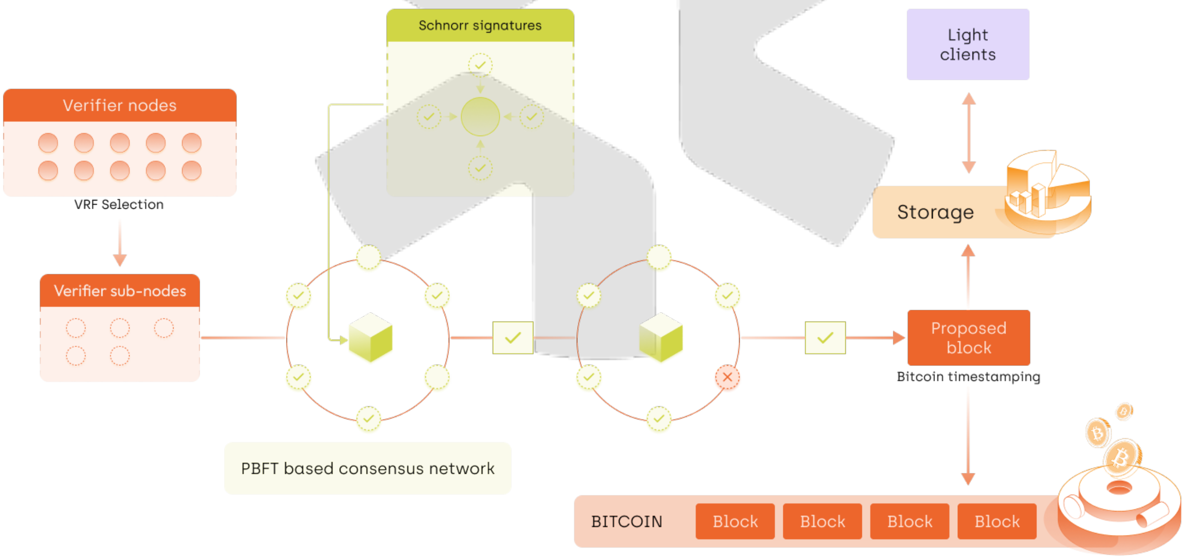

FIG. 2. VERIFICATION AND CONSENSUS FRAMEWORK

3.3 Sub-Parallel Merkle Tree

Merkle Trees are widely used in various blockchain ecosystems to expedite the verification process of transaction existence. Additionally, some projects combine Merkle Trees with Zero-

Knowledge proofs to collectively verify the existence and validity of transactions. Most of the original transaction data corresponding to Merkle Trees are stored in decentralized off-chain storage systems. During verification, nodes face the challenge of downloading all original transactions, resulting in significant data communication overhead for individual nodes. Some existing projects use Data Chunks and Data Sampling (DAS) techniques to balance the data transmission overhead of node verification and the correctness of global data verification.

Building upon this, we propose to integrate Merkle Trees with Data Chunks and Data Sampling (DAS) techniques, extend the structure of Merkle Trees, and propose the concept of Sub-Parallel Merkle Trees. This involves generating corresponding sub-Merkle trees for each data chunk/group, where each data chunk/group's verifier node only needs to rely on the sub-Merkle tree and the corresponding data chunk/group to complete data verification. Verifier nodes of different chunks can verify in parallel, resulting in a significant improvement in overall verification efficiency.

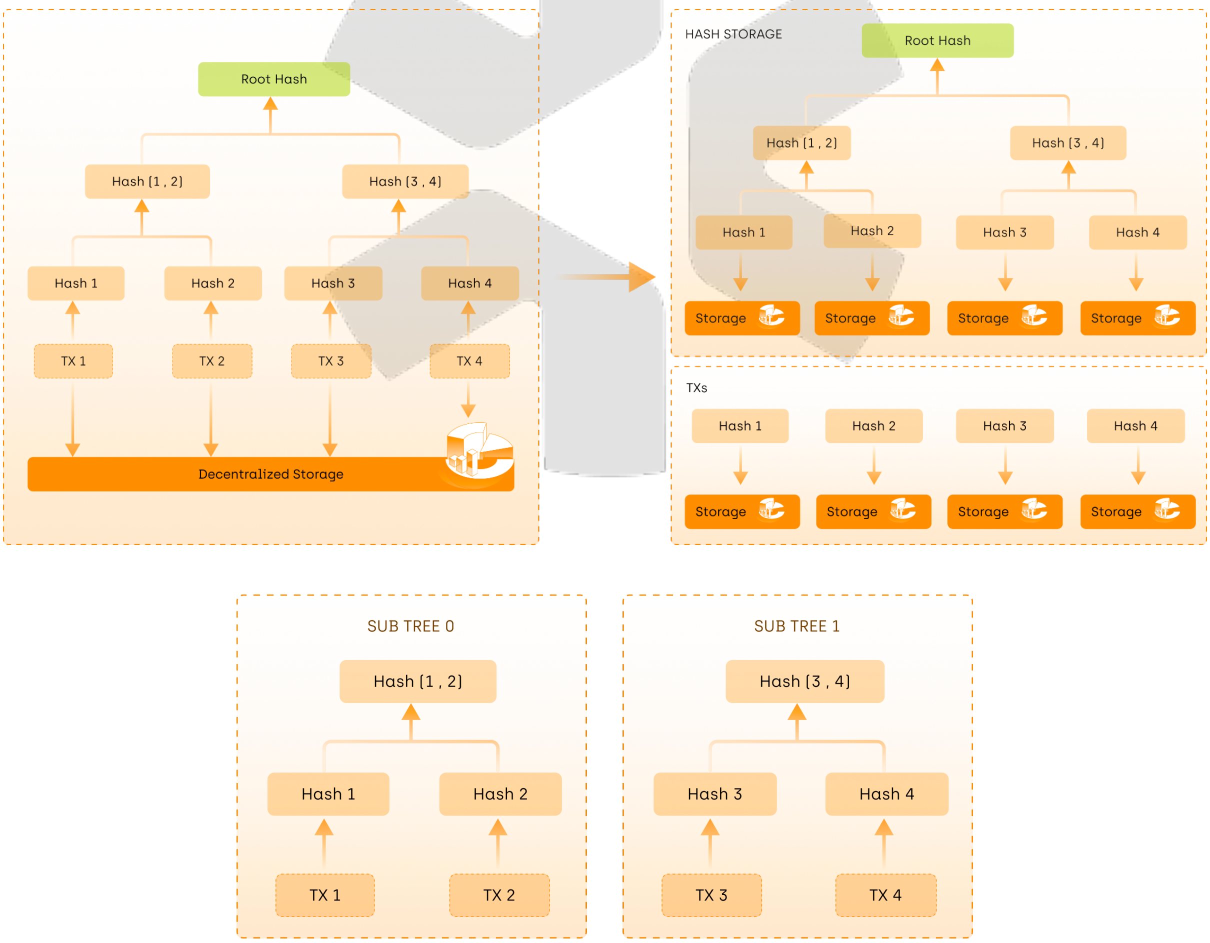

FIG. 3. SUB-PARALLEL MERKLE TREE

4. Conclusion

In summary, TNA Core presents a comprehensive framework to standardize data verification across Bitcoin networks. By enhancing trust, security, and interoperability, the solution contributes to the broader goals of scalability and operability within the Bitcoin ecosystem. We acknowledge the importance of ongoing refinement and collaboration within the blockchain community to address potential limitations and further advance the capabilities and sustainability of Bitcoin's multi-layered architecture. Ultimately, by facilitating efficiency trustworthy data interactions, our framework seeks to promote the continued growth and resilience of Bitcoin's evolving ecosystem.

References:

1. Ethereum Contributors. *EIP-4844*. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-4844.md
2. Ethereum Contributors. *EIP-2718*. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-2718.md
3. Feist, D. (2020). KZG polynomial commitment. https://dankradfeist.de/ethereum/2020/06/16/kate-polynomial-commitments.html
4. Paradigm Research. (2022). Data availability sampling: From basics to open problems. https://www.paradigm.xyz/2022/08/das
5. Micali, S., Rabin, M., & Vadhan, S. Verifiable random functions. https://people.csail.mit.edu/silvio/Selected%20Scientific%20Papers/Pseudo%20Randomness/Verifiable_Random_Functions.pdf
6. Arcology Team. Parallel Merkle Tree. https://website.arcology.vercel.app/docs/parallel-merkle-tree
7. Nubit. (2024). Bitcoin-Native Data Availability Layer with Instant Finality.
8. Gustavo F., etc. (2020). AutAvailChain: Automatic and Secure Data Availability through Blockchain. https://ieeexplore.ieee.org/document/9322396