TrueNFT Specification

Ian McJohn, Sebastian Mendez

August 2022

Abstract

An on-chain storage management and custody system, designed initially for NFTs but supporting any applications where perpetual storage is desired via a single initial payment.

1 Problem Overview

1.1 NFT Landscape

The goal of this project is to address a specific concern that has existed in the NFT landscape for quite a while now, without any truly effective solutions. As long as NFTs have existed, they have typically consisted of some on-chain identifier token, proving ownership as well as some form of data identification. In most current implementations, this data identification has taken shape as a hash of the desired data, as well as a link to the data on either the traditional web, or a semi-decentralized filesystem such as IPFS. Because of this, NFTs inherently do not fulfill the promise of their design, as they are neither immutable nor permanent, as recent examples have shown.

The NFT community has largely accepted this without complaint for two primary reasons. First, as the lifespan of most NFTs is relatively short, with the industry only recently coming into popularity, most of the potential risky situations (ex. An artist running out of hosting money/moving on to other projects) that would cause an NFT to lose pinning on the traditional web or IPFS are many years down the road. Additionally, even if these concerns are present in the community, there has not been a comprehensive on-chain solution to store this data in a more effective manner. This is because all on-chain storage solutions require continuous payment in the same way that traditional hosting or IPFS does, thus providing negligible improvement for the significant technical hassle that integrating them brings, as they were not designed with NFTs in mind from the onset. Thus, a clear market niche clearly exists for a pay-once, store-forever style chain, where both NFT ownership and storage are integrated on the same blockchain, one that is purpose-built for the sole design of storage and ownership of the NFT ecosystem.

1.2 Current Technologies

In the storage space, one of the current state of the art chains is Sia. As a brief overview, the core project consists of a layer 1 which allows users to rent storage in a decentralized manner, as well as an additional layer 2 titled Skynet, which provides an efficient manner to make files on rented storage accessible by anyone, using unique identifiers called Skylinks which allow you to locate a file on the Sia storage chain. As this chain provides decentralized storage with internet-global location capabilities, it clearly seems like a powerful tool to implement the required features of NFTs on. However, it has minimal to no adoption in the NFT space, for two clear reasons. First, storage costs must be consistently kept up in the form of maintaining your contracts on the network, or your data will be removed. It is unclear whether the onus of contract maintenance should lie on the artist, piece owner, or custodial organization (ex. OpenSea), making the risk of an NFT dropping off the network intolerably high. Additionally, the Sia blockchain holds no native capabilities to handle verifiable ownership of the token portion of the NFT, meaning that it could only be used for storage and would have to be linked with a more commonly used NFT chain (ex. Ethereum, Solana), in order to provide those aspects of the system. Despite this, it clearly has great value as a decentralized store of data, something no current NFT chain has the ability to provide.

2 Implementation Overview

2.1 Initial Codebase

As can be inferred from the above problem statement, this codebase will be based off of a fork of the existing sia blockchain, with functionality added in order to make the chain more capable of supporting NFTs. Once these changes have been made to the storage and transaction layers of the base blockchain, an NFT-oriented layer 2 will be added, in order to allow for easy and structured access to the base layer of the overall designs.

2.2 Payment Structure

The primary goal of these changes is twofold: First, change the payment structure of the sia storage contracts from a standard host-style pay as you go model, to something with a more uniform distribution. Rather than paying to keep your NFT "pinned" to the network, you will instead pay to mint an NFT, or transfer one. Additionally, part of the minting fee of the NFT will be placed into lockup, and will be redeemable at any time (titled liquidation), at which point your contracts to host the NFT will be made null and void, with no hosts having the obligation to store the data anymore. This provides financial incentive to users to either remove their own data from the chain when the underlying NFT is no longer desired, as well as providing incentive to market-making third parties to purchase "bargin-bin" NFTs and liquidate them for their lockup value, thus minimizing the storage burden on the overall network. Both mint and transfer operations will distribute their respective fees to the storage hosts on the network in proportion to total data stored, providing a revenue stream to replace that of contracts under the previous sia ecosystem.

2.3 Chain-of-Custody Implementation

Secondarily, the transaction layer will also be modified to support proof of NFT ownership, to go along with the above features that complement the storage layer. This will be done in a fashion similar to the Colored-Coins whitepaper, such that the additional data field on the tNFT blockchain will be used, upon NFT minting time, to tie that NFT to its matching piece of data on the storage layer. Thus, it can be transferred, swapped, or exchanged like any other asset on the tNFT chain, while bearing significance that it corresponds to a piece of stored data. Once these changes have been met, the blockchain will have the full functionality required to support minting, exchange, and storage requirements of modern NFTs, at which point a layer 2 can be built according to these specifications to function as a wallet-style interface for users looking to manage these assets.

3 Technical Implementation Strategy

3.1 Core Concept: Storage Pools

3.1.1 Pools Overview

The core change required in this storage network that Sia does not provide is a means to consistently pay hosts without requiring additional intervention on the part of the owners of the data. Thus, rather than having the owners hold contracts with specific storage hosts, we will instead have them pay into a storage pool at the time of minting and transferring of NFTs. This liquidity will then be used to pay out the holders of storage contracts in a rate agreed upon by the network, thus encouraging hosts to be proactive in establishing hosting contracts without any intervention from the renter. Additionally, the number of contracts for a specific chunk of data stored will be limited, to require hosts to offer a wide variety of data in order to receive significant payments.

3.1.2 Pools Implementation

By modifying the core transaction types in the Sia network, specific transactions can be designated as minting, transfer, and liquidation transactions. These will only be accepted as miners if they fill the conditions described in the below custody section, thus ensuring any valid NFT will provide the requried liquidity to the storage pool. Additionally, file contracts will be modified to support only hosting data with merkle roots corresponding to valid NFTs, and will be funded from the central storage pool rather than a specific renter. This can similarly be implemented by consensus-level modifications, where miners only mine the new type of contract.

3.2 Custody (storage pool inflow)

3.3 Overview

The NFT chain-of-custody layer a standard (similar to ERC-721) level of on-chain asset custody, while making these artifacts first-class citizens on the blockchain, allowing for faster, cheaper transactions. Additionally, this chain-of-custody layer supports the balanced economic system encouraging users to remove excess data from the network, while guaranteeing that storage pools remain appropriately funded.

3.4 Minting

The minting transaction type takes in the merkle root of new data that is to be stored as a custodial artifact on the network, without uploading said data. Additionally, it takes the required amount of cryptocurrency to support the liquidation fee, as well as an appropriate funding level for the storage pool. The liquidation fee is burned (to be later recovered potentially), and the storage funding is transferred to the appropriate pool. Asssuming all fees are paid correctly, the consensus layer will include this transaction in a block, and the new NFT storage artifact will be an output of the transaction.

3.5 Transfers

Transfers operate similarly to the minting structure, with one slight difference. They take in the funding levels of a transfer fee as well as the previous output for an NFT. The funding is transferred to the storage pool, and the NFT input is spent to the appropriate address of the new owner.

3.6 Liquidation

Liquidations take in a previous NFT transaction as an input, but provide no NFT output. As they are marking an NFT for non-storage, there is no artifact to retain custody of following the transaction. Instead, their sole output is a coin minting equivalent to the initially burned liquidation fee, thus encouraging the use of liquidations to discard unused or obsolete artifacts.

3.7 Host Payments (storage pool outflow)

3.8 Overview

The purpose of the newly defined payment structure is to provide a way for hosts to receive continuous payment without requiring the owners of storage artifacts to maintain any on-chain presence (ex. on sia, holding and maintaining contracts). Thus, the hosts are encouraged to create and hold contracts without requiring any user involvement.

3.9 Implementation Details

In order for this implementation to remain functional, there are two key components. First, contracts are modified from the existing sia structure to only support storing sectors that have been appropriately funded by being linked to a custodial NFT artifact (which contains the merkle root of any data it has sponsored to store). Second, all valid contracts are permitted to take funding from central storage pool in a manner similar to the balanced SiaFunds pool strategy, in which they mint coins in marked transactions that treat the associated mint as a subtraction from the pool. Finally, in order to avoid abuse of the central storage pool's resources, for example by holding many contracts for a piece of data you hold only once, each NFT is limited to having only a set number of contracts at any given time for that particular asset. This allows for encouragement of storage of diverse data, as a storage host must hold contracts for many artifacts in order to properly utilize their full disk space.

3.10 Economic Balancing - Storage Guardrails

Initially, the storage proposal used fixed and percentage-determined values in order to set prices for minting and transferring, the two operations that would fill the storage pool. However, this was not flexible enough to account for a large price swing in the underlying token, as the storage hosts would still have to pay their costs denominated in USD. Thus, the storage guardrails proposal was added to the spec, and is expected to be implemented post Proof-Of-Concept. In storage guardrails, minting and transfer fees for an NFT are not fixed, but a host can refuse to store artifacts that did not hold to a specific level of pool contribution at the time that this guardrail was put in place. Thus, if the price of the token drops with respect to the US dollar by $10x$, hosts can indicate (by announcement broadcast), that they are only willing to store artifacts for newly minted NFTs that contribute ten times more in minting and transaction fees to the storage pool than ones previously added to the chain before this economic event. While this still acts as a somewhat slow response to broader price shifts, it significantly decreases the odds that hosting will become fiscally infeasible for the majority of the network.