Esaiyo Trace: Blockchain Auditing for Multi-Chain Digital Assets

S. Ryan Quick
Raymond St. Martin
Michal Bacia

Abstract: Esaiyo believes all objects — physical, conceptual, and digital — have stories to tell and rights to individual existence which is secure, provable, testable, and audited. Evidence of asset behaviors should be easily available to anyone, anywhere. Currently, when a digital asset is created, the history, attributes, and behavior of the asset remain intact only so long as that asset remains on the platform where it was created. For non-fungible tokens (NFTs) especially, the "nature" of the asset implies the ability to migrate between markets which can best match with the value of the asset[1]. Over time, NFTs will spend significant time in secondary markets, perhaps even more than on their original platforms[2].

While much attention has been paid to bridging assets between chains and platforms, there is little scrutiny regarding the metadata of the NFT — including history, digital rights, ownership, relationships to other objects and assets, legal presence and jurisdiction, and characteristics of the object. Esaiyo Trace (Trace) preserves both the asset transactional and intended[3] behavior as well as its characteristics and implications for the entire life of the NFT. Moving across chains need not leave valuable information behind (when "burning" NFTs on the original chain). The moves need not be "new beginnings" for an asset (when "minting" NFTs on the new chain). Over time these "re-starts" impact the intrinsic, archival and economic values of the asset itself. Esaiyo Trace maintains all aspects of the original asset as it migrates in a manner which preserves the primary capability of blockchain technology (Antonopoulos, 2017; Arsheep Bahga, 2017)[4].

1. Problem Statement

NFTs do not carry their existing blockchain history or the metadata associated with the owner and previous ownership when moving to another platform or chain. Current on-chain NFT platforms do not have any mechanisms to codify intended, attempted, failed, or unexpected behaviors which are not handled by the contracts themselves. Smart contracts can only report and handle behaviors which are anticipated by the contract itself. There are many situations which they cannot detect, let alone handle. Nevertheless, a complete picture of the as-

[1]For example, in the art world it is not uncommon for a single work to be presented and sold on a variety of markets including galleries, private sales, and auctions in its lifetime. Even for works which are commissioned and held over time encounter additional markets like estate settlements, wills, and gifts which directly impact their intrinsic and economic value.

[2]That is to say that for most digital assets secondary transactions will far outnumber the initial creation and ownership settlement events. This is true both for block time and for wall clock longevity for the asset.

[3]Intended behavior includes attempts, failures, unexpected results and other similar activities.

[4]Blockchain technology is founded on the notion that providing a single chain of information, one block at a time, secured with taper-proof "cement" delivers a next-generation capability for chain of custody. It is from this simple tenet that all of the web3 ecosystem builds upon. At the end of the day, it is this simple "one thing at a time" approach that enables trustless systems, decentralized exchange and governance, and the wide variety of other applications developed and developing (Antonopoilos, 2016).

set requires verifiable data of all these behaviors to present an accurate view of the asset itself.

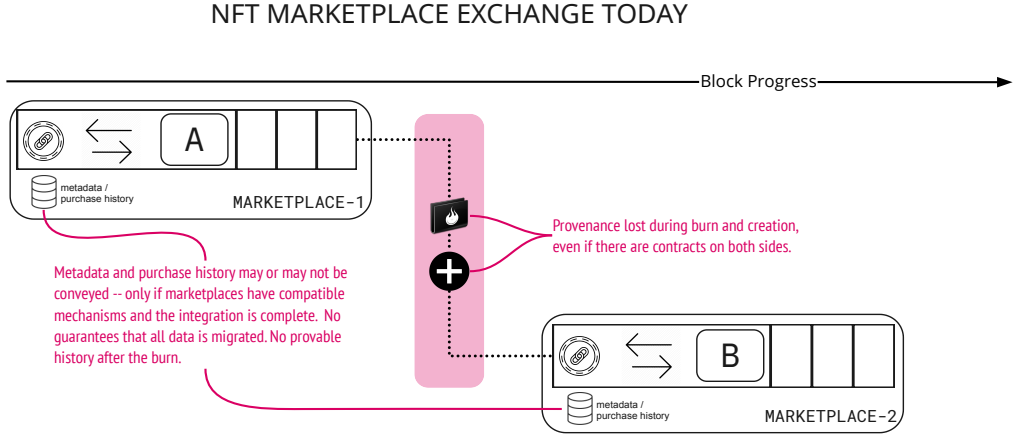

Figure 1: Cross-chain Asset Movement

Figure 1 illustrates a cross-platform token migration today. NFT-A is created on Marketplace-1 and includes a set of origination metadata, attributes, and transactional details. The move between Marketplace-1 and Marketplace-2 requires that the original asset be converted to the schema and format which Marketplace-2 understands. NFT-A must convert into a new entity, NFT-B — with its own associated metadata. The operation to perform this conversion is a destructive event for NFT-A however, with only the executing contract performing the actual work of the conversion[5]. Nowhere is there an extant link between Marketplace-1 and Marketplace-2 or any kind of binding whereby NFT-B can be understood as NFT-A. The two are distinct entities with distinct histories and characteristics (in fact, for most scenarios, NFT-A no longer exists and no longer has history at all).

The Trace system addresses these problems by preserving all data relating to an asset on chain. This data includes

- the transactional data as represented on the source chains for Marketplace-1 and Marketplace-2;
- the metadata and attributes associated with NFT-A and NFT-B;
- and any intended but otherwise uncodified behavior of the asset.

The Esaiyo Engine employs a dedicated blockchain which is distinct from the origination and destination chains/platforms for a trusted ledger solution. This 3rd chain (Chain-C) allows for a trustable method to codify all aspects of the asset which are not handled natively or through its contract definitions, and those aspects which are subject to change over the asset lifetime. Chain-C and its data make up Object Provenance for the digital asset.

Providing both web2 (http2, gRPC) and web3 (smart contract, on-chain) APIs allows for any consumer to make use of Object Provenance data in ways which benefit their ecosystem and plat-

[5]In this case, we illustrate the most common method for NFT migration: "burn and create". To see an example of some of the metadata which actually changes between OpenSea/polygon and Binance Smart Chain see Figure 3, Figure 4, Figure 5 (Binance, 2022b; Boba, 2022; Cross-Chain, 2022; Visha, 2021).

form. Consumers may construct their own flows to provide additional verification, validation, agreement, and surety for the asset and its lifecycle.

2. Benefits

2.1. Provenance

Esaiyo Trace enables intent, execution, and state-auditing where all observed behaviors are on-chain. Operating alongside any existing web2, Layer 1, or Layer 2 solutions, the Esaiyo Engine creates a common verification and validation point by which platforms, owners, and audit entities looking into digital assets and NFTs specifically may leverage Esaiyo Engine provenance to aid in trust, validation, proof, and activity research. This enhances the service offerings from source platforms and owners, and improves the brands behind the NFTs and their collections. For example, the Esaiyo Trace system might be used to create a single pane of glass (SPOG) with information spanning multiple chains and digital asset platforms (see Figure 5). Object Provenance data provides much richer information than traditional block explorers by including associated metadata and verifiable relationships to other entities. Metadata connections provide a richer context for search and exploration between platforms and digital assets themselves. Connection and metadata contexts enhance the individuality (and value) of the asset.

2.2. Asset Metadata and Characteristics

Digital Assets in cryptocurrency and web3 systems are defined by the standard which enables their creation, existence, and trading on their Layer 1 or Layer 2 native platform. In almost all cases, the standard is one where key components of the asset — especially those which are not strictly necessary to define custody, economic value, and asset transfer — tend to be defined in metadata as attributes or characteristics. The method for obtaining this metadata varies from platform to platform, but is generally comprised of at least some off-chain elements. (In many cases interesting metadata is entirely off-chain) (10). For example, the OpenSea NFT platform includes on-chain metadata for ownership and buyer fee basis points (OpenSea, 2022), but the notion of what constitutes ownership in the first place is defined in multiple off-chain locations and includes legal contracts and user agreements, implied value transfer, and derived characteristics such as object longevity. No cryptocurrency or digital asset standard takes such attributes into account when managing NFTs[6].

Trace provides on-chain consistency and validation for these diasporic characteristics and combines them with the rich record of behavior required to capture and present the complete asset view. Without compromising the integrity of the blockchain or sacrificing performance, the Esaiyo Engine ensures that the value of the digital asset includes the entire history, attributes, and capabilities of that asset — regardless of its native home and platform.

[6] As of the time of this whitepaper, there are no standards which cover these details.

2.3. Multi-Chain By Design

Since the value of digital assets presumes platform movement, a complete picture assumes the logical extension of digital assets across the entirety of web3 blockchains, networks, and platforms. The Trace ecosystem delivers cross-platform and cross-blockchain capabilities from the outset. Our initial reference implementation for the concepts presented in this whitepaper bridges EVM-compatible and non-EVM platforms and bridges blockchains without restriction. The provenance and attributes of the Esaiyo managed asset are available throughout its lifecycle and captured independently throughout all movement and transactions regardless of the origin of those events.

Consider the case of an art NFT which is a limited-issue member of a collection. Once purchased, the ownership of the asset transfers to an account available to the minting platform. The notion of ownership and precisely what rights are conferred are held in the user agreement and legal restrictions assigned by the platform to both the asset and to the purchaser at the time of purchase. Neither the asset nor the purchase transaction capture these rights explicitly. Rather, there are off-chain locations where these rights are enumerated. The only association of these rights to the actual asset is managed also off-chain by the issuing platform.

In turn, when this asset is sold on another platform, the purchaser and asset assume a new set of rights as defined by the new platform. These rights may or may not align, augment, or conflict with those of the original asset. Consider the case where the original purchase is deemed null and void should the transfer leave the issuing platform — reverting the ownership back to the minting artist or contract. Who owns the asset once it is moved? Can an asset which is stipulated as null and void if it attempts to leave be said to exist in any other platform at all? This is sticky legal ground to be sure. For high-value assets, this can (and likely will) result in disputes entangling not just the artist and owners, but their platforms, blockchains, governments, property rights experts, etc. Esaiyo Trace ensures that these sorts of implied or external attributes (such as property rights and guarantees) remain associated with the digital asset in a manner which can prove priority and chain of behavior. While Trace may not answer these questions, it can provide the information and insight needed to assist the experts in doing so.

3. Architecture and Implementation

It is well understood that for distributed consensus to operate properly and for blockchains to make progress, all applications running on those chains must represent finite states[7] (Hopcroft

[7]In order for block sealing to create verifiable transactions which can be used to ensure the "one thing at a time" nature for the next sealed block, all transactions involved in sealing blocks must arrive at the same results in all cases. This means for that any operation which must be considered for block validation, the outcomes of that operation must be predictable at all times and by all verification implementations. Thus, we must have finite states which are (in most cases) well-defined. While this sounds obvious, it is limiting to the types of data which may be validated and thereby can limit the types of transactions which can be placed on-chain. Metadata of unique assets — by its very nature of being unique and adding to the singularity of the asset to which it applies — can often run afoul of this finite state rule precisely because this uniqueness increases the likelihood that a complete view of an asset might appear random to a state machine. These additional calculations and verifications then can become extremely taxing (expensive) to the smart contract execution engine and eschewed. While there are consensus mechanisms which can tolerate these sorts of random states, it is not reasonable to expect that all blockchains and ledger solutions will adopt these same consensus algorithms. In order to bridge to all possible ledgers, then, we must create data structures

et al., 2007; Rich, 2008; Wikipedia, 2022). All smart contracts essentially represent state machine automata with absolutely predictable (and therefore verifiable) inputs and outputs. This also implies that maintaining any sort of variable, unpredictable, or random[8] information on-chain is difficult when using finite state automata.

The general design for Trace leverages a third blockchain (Chain-C) which records both the asset and its associated transactions as well as the attributes and characteristics of the asset (see the Esaiyo Engine chain in Figure 2). In many cases these attributes are available through the platform's on-chain interfaces, but there are also additional metadata and implied or derived characteristics which originate on the platform or through relationships between the asset, owner, buyers, or hosting platform. And unlike the on-chain asset, there is no means by which these characteristics can be verified to be permanent, and non-malleable. Once off-chain, the platform may make any modifications, additions, subtractions, etc as deemed necessary without regard to the on-chain provenance of the asset.

Chain-C serves as a common integration point between all platforms on which the digital asset might come to exist. By making attributes available to both web3 (on-chain, smart contracts) and web2 (HTTP2/gRPC) APIs, asset owners, hosting platforms, and audit providers can easily integrate the provenance of digital assets into their own ecosystems.

which are verifiable even when their data is not, or leverage other schemes and algorithms within information theory which can emulate discrete and finite outcomes in the block-moment. This is an exhaustive topic on its own. For the discussion here, it suffices to say that any solution which purports to support and even enhance individuality is fundamentally difficult in systems supported solely by finite state machines.

[8]From the point of view of the blockchain and its validators, any variable information which cannot be verified in the same manner by all consenting nodes cannot allow block-progress. Simply put, if the majority of the validators cannot agree on an outcome, then that transaction cannot be permitted in order to ensure block progress. In the case of bridge-crossing with metadata beyond the scope of the token standard, and without coordination and provable collaboration between validators on both networks, block progress must leave out the transaction. In cases where the block includes information which is accessible only by validators on one side of the bridge for example, it is impossible for the validators on the other side to obtain the information required to arrive at a verification decision. The solution to date, then, has been to largely ignore metadata and the community accepts these losses as a manner of course (Antonopoulos (2017); (Pierce, 1980)

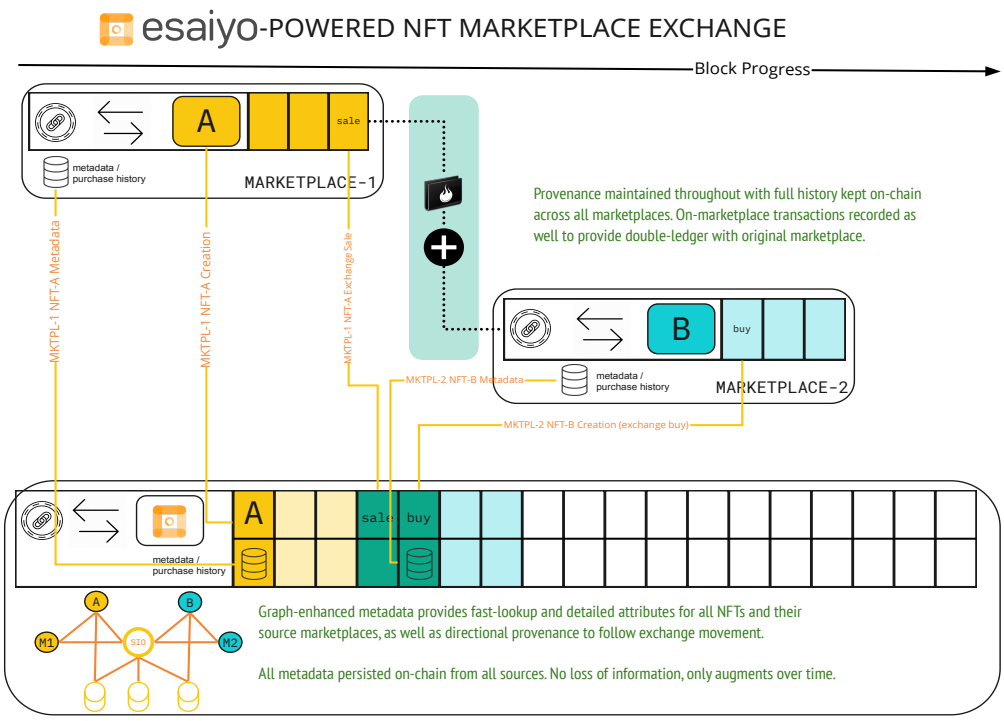

Figure 2: Esaiyo-powered Cross-Chain Movement

3.1. Architecture Approach

There are two critical issues which need to be solved for a successful implementation:

1. Asset behavior over time must be accurate, recorded, audited, and verified throughout the lifetime of the asset — regardless of the platform on which the asset is traded or held. (We refer to this as delivering a "trustable asset"[9]).
2. The detailed information and data which make up the complete view of the asset must be preserved.

That is to say that the asset is more than a simple tradable token, but requires all of the associated metadata which adds to the richness of the object itself. This information is specific to the chain and/or platform on which the token trades at the moment, and there is no standard to which all assets are held. Since the attributes can and will change over time and platform as well

[9]The difference between *trusted* and *trustable* is an important distinction. Esaiyo Trace provides much of the information required to derive trust from an asset. For instance, in order to prove that the chain of ownership for an asset which has been purchased 5 times on 3 different platforms, we require a complete set of records of the asset, the descriptive metadata associated with the asset, and the transaction details as the asset bridged between owners and between platforms. Esaiyo Trace maintains this set of records and offers externally verifiable APIs to view and validate the records as accurate. This is *trustable* data. A *trusted* asset is one where the buyer and seller leverage *trustable* data to make decisions as to whether or not the asset is actually the one in question and can prove beyond reasonable doubt that is the case.

as the standards which are applied to the asset, any solution here must maintain the sum total of all attributes throughout the lifetime of the asset. We refer to this as the "complete asset view"[10].

Providing a trustable asset ensures that all aspects of ownership are maintained on the blockchain. There are a variety of chains and platforms functioning at all levels and the bridge relationships cannot be predicted. The NFT behaviors are not finite states which can be defined in finite state automata like smart contracts. The best solution is to create an audit chain-of-custody which captures all of the asset behavior. Verifying all behavior requires an auditor to validate the chain-of-custody. Esaiyo Trace uses Chain-C to provide these behavior records[11].

Capturing all attributes requires a flexible platform for data retrieval, search, and storage. Ensuring that attributes are verifiable and tamper-proof requires their addition to an auditable ledger. The Trace system leverages a fixed-governance blockchain to assure that the stored attributes are correct and un-modified from their original sources.

Combining verifiable and trustable auditing with asset richness then, Esaiyo Trace keeps the complete view of the asset on-chain — regardless of the source of the record and whether the asset is considered fungible or not[12]. In all cases, access to verify and audit these chains will be made available through both on-chain smart contacts and through external APIs.

3.1.1. Behavior Chain

The Behavior Chain captures all transactions for digital assets. These include traditional create, destroy, and economic value (buy/sell/trade) transactions, and also include intended behaviors such as bridge traversals, escrow, fractional asset division, and others yet to be invented. By creating a

[10]In much of the blockchain/crypto ecosystem, a token is considered complete. The data contained within the specification for a token however, tells only part of the story. Indeed in the ERC-1155 standard, part of the definition for descriptive metadata allows the reference of external URIs pointing to additional information about the token. Quoting the standard itself: "The ERC-1155 standard guarantees that event logs emitted by the smart contract will provide enough data to create an accurate record of all current token balances. A database or explorer may listen to events and be able to provide indexed and categorized searches of every ERC-1155 token in the contract." The bias is clear even in the standard that metadata is required and desirable but at the end of the day only token-balances matter to chain behavior (Witek Radomski, 2018).
This bias misses the fundamental nature of digital assets as independent objects with individualized behaviors and characteristics. While the standard acknowledges that there is more needed to describe tokens — especially assets which purport to be NFTs — the interface to off-chain resources does not satisfy trustability requirements over the lifetime of the asset. To be complete, the transactional token must be combined with all of its extent and derived metadata in a chained persistence mechanism which results in a verifiable chain-of-custody. Such combinations of data and non-reputable storage with validation APIs provides a "complete asset". This is the fundamental building block of the Esaiyo Trace system.

[11]This is currently an implementation choice to limit fees and ensure performance. There is nothing in the design which precludes either traditional on-chain transactions or Optimistic Rollups from satisfying these requirements as well(Boba, 2022; Optimism, 2022).

[12]Assuming that those chains are integrated with the audit platform, there is nothing precluding either permissioned or permissionless blockchains and networks. The choice to use permissioned blockchains for the initial deployment is strictly an implementation choice.
Fungibility and uniqueness can be themselves enhanced and proven through Trace as well. Object history helps to add individuality to the asset, and additional trades and economic valuations can enhance fungibility.

wholistic chain-of-custody for a particular digital asset delivers a reliable way of understanding its state over time.

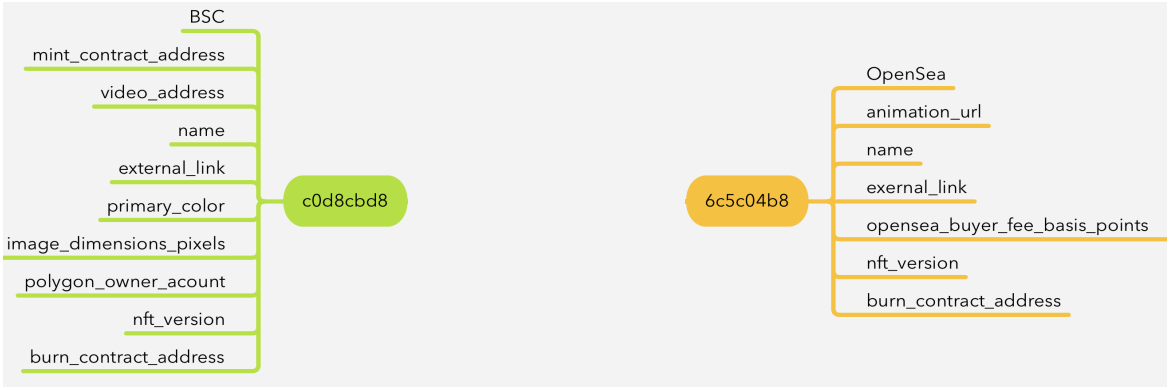

Figure 3 : NFT Metadata after platform migration (today)

3.1.2. Behavior States

Preserving behaviors on-chain requires well-defined observed states to feed into the automata contracts supplying provenance for those behaviors. Attaching discrete definitions for events which are sometimes combinatorial, sometimes measured, and even sometimes implied, allows Trace to utilize existing blockchain solutions for provenance contracts. These well-defined state definitions are below[13]:

- Observed
  - These are behaviors which are directly measured by the publisher. An asset A purchased for price B with currency C is an observed event.
- Implied
  - These are behaviors which are implied by one or more measured (observed) behaviors. For example, using the purchase event observed above, we can create an implied event that owner D entered asset A onto the market for purchase.
- Derived
  - These are behaviors which are derived or determined from the outcomes of one or more behaviors. Generally derived behaviors combine events from multiple platforms which are not directly involved with one another. Using our example above, a derived behavior might be a change in the rights for Owner D to govern overall behavior on a blockchain as a result of its divestiture of asset A. Another example might be the change in rights of ownership to the property "asset A" due to the nature of its being sold for price B and currency C.

[13]While these states are our own, we owe a lot to our ontological and event methodology thought process to a laundry list of thinkers in the space. See the following to list a few of the greatest influence. (OCLC, 2022; Pierce, 1980; Quick & Kolster, 2014; W3C, 2022)

- Hidden
  - Hidden behaviors are those behaviors which exist, but are not observed in the originating platforms. For example, if the sale example resulted in a transfer of asset A to another platform, the "burn" event would be observed, but the notion of "platform migration" is not observed in a single operation. This is a hidden event.
- Intended
  - Intended behaviors represent the goals of an event or action, independent of their outcomes[14]. In our sale example, the intent of the contract handling the sale is to manage a conversion of currency for an asset and to pay for the execution of the contract. These intended events were all successful in our example. If this is the third attempt to make the sale, the first two failed attempts represent intended sales events which were finally unsuccessful -- but events nonetheless.

These states are not mutually exclusive, but can be combined and implied to create complex events and data as required.

Behavior Types *Asset Create* [observed] transactions record the creation of digital assets. This includes the initial minting of the asset and additional create operations as the asset moves between platforms and chains. For example, in Figure 3 the 6c5c04b8 asset on OpenSea/polygon (orange) eventually becomes the c0d8cbd8 asset on Binance Smart Chain (green) (Binance, 2022a). In the state of NFT migration today, these are two distinct and independent digital assets. There is no notion on-chain that these two assets are related to anything, let alone one another. In Figure 4 however, these create events are linked through Esaiyo Trace to provide a clear record that the assets described in each section are indeed the same asset (SIO 6b446a3e) even though neither platform carries its complete history.

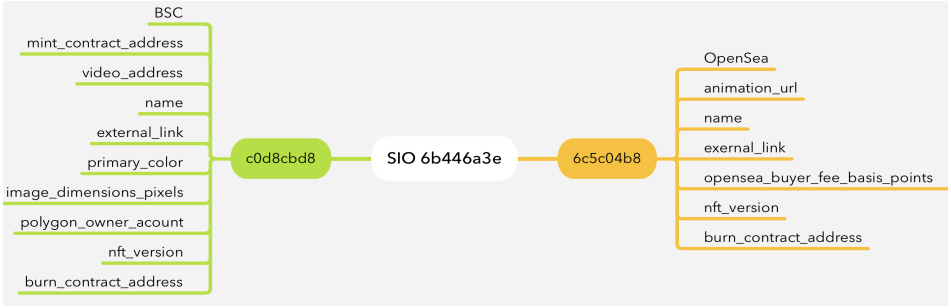

Figure 4 : NFT metadata with SIO and Esaiyo Engine

When the asset moves back to OpenSea/polygon, the problem compounds. There would there be no record or linking concept on the original migration, so OpenSea would not remember the prodigal asset returned (OpenSea, 2022). Esaiyo Trace changes all that. Not only is it possible

[14]Currently blockchains provide records of outcomes, but there is no record of what is intended at any moment. This is true for all transactions successful or not. We see the A-to-B transaction, but have no indication that there was an intent to create an A-to-B sale, how many times that attempt might have been made, whether there were different input parameters when the intent was made, etc.

to construct a chain of custody and path of ownership, but richer links provide additional insight into the asset – contributing to its uniqueness and historicity[15] (Figure 5).

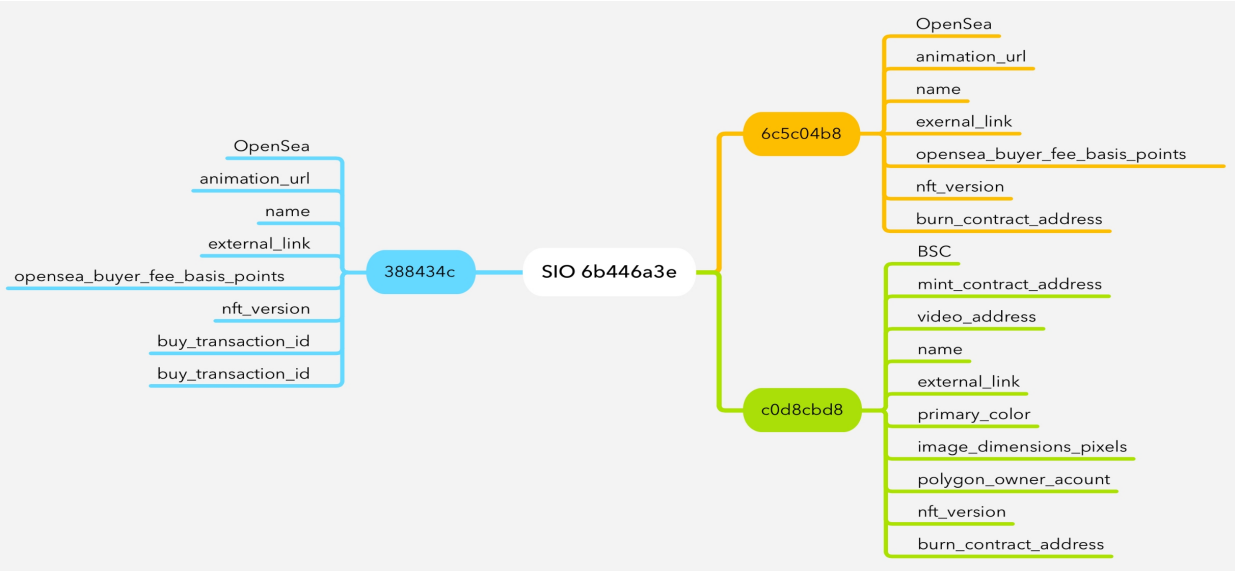

Figure 5: NFT metadata with multi-chain Esaiyo Engine

Asset Destroy [intended, derived] transactions are generally on-chain markers that an asset is no longer available for ownership or value transfer. In the Trace system, these mark the end of trading on one platform and point to created assets on other platforms and form a critical link in its custody chain.

Asset Modify [observed, derived, intended] transactions are those in which an asset may be altered from the original state. There are only a few instances of this behavior at the time of creating this whitepaper. For example, converting a single asset to fractional ownership would constitute a fundamental change in the nature of the asset itself. Generally, modify transactions change the digital constitution of an object and are not simply modifications to attributes. The immutability of the digital ledger keeps these modifications rare, if not impossible[16]. In all cases, however, these transactions are maintained in the Behavior Chain to keep the state of the asset intact.

Asset Obtain Attribute [observed] transactions are internal transaction markers for events within the Trace system where the associated attributes are retrieved from the owning platform. Those internal transaction markers are then stored in Esaiyo Trace as events alongside the asset's other attributes.

Asset Modify Attribute [intended, derived] transactions are internal transaction markers for events within Esaiyo Trace where the associated attributes are retrieved from the owning platform

[15]While we are not making a case for how value changes with regards to the historicity and attributes of a digital asset, certainly there is much to be learned here. Just as knowing the history and context of a sale involving items like home run baseballs can increase their value, so these same principles will come into play with digital assets once these sorts of verification and validation processes are widely available.

[16]Though not unheard of. Indeed, some might argue that "chain forks" represent precisely this modification of digital assets due to the nature of the supporting blockchain and ledger application changing "beneath" the asset itself. This deserves more discussion than we can apply here.

and those attributes already exist. The platform does not modify anything, but appends the values to the asset events so the revision history remains available.

Asset Destroy Attribute [hidden] transactions are internal transaction markers for events within the Trace system where the associated attributes are retrieved from the owning platform and those attributes are removed or destroyed. The platform does not modify anything, but appends the values to the asset events so the revision history remains available.

3.1.3. Attributes Chain

Essentially the items on the Attributes Chain are combinatorial hashes of the attributes themselves — which are then kept in an attributes store. Each attribute key-value pair is hashed individually, and the resulting data structure[17] is hashed to be stored as a behavior record. This behavior is termed *Asset Obtain Attribute* on the Behavior Chain. The records for the Attributes Chain form a chain of custody for metadata and characteristics of the asset which are not guaranteed to be on-chain in the source platform. The practice of hashing and roll-up hashing is common both in existing cryptocurrency applications and in off-chain financial applications – and is considered industry best practice for managing unlike comparison data in finite-state storage systems (like blockchain, relational databases, etc).

3.2. Engineering Approach

The reference implementation for Esaiyo Trace leverages both EVM and non-EVM source and destination asset chains and platforms[18]. The Behavior Chain and Attributes Chain are implemented using private Ethereum Clique (Szilágyi, 2017) as well as in the Cassandra NoSQL database (Cassandra, 2022). This dual-write functionality improves search-ability, retrieval latency, and allows the Trace system to make alternate choices for persistence as the platform grows. Both the Behavior and Attributes chains write to the same blockchain instance in the initial implementation but may be separated later as required. Such choices for the reference implementation should be considered as deployment-time engineering decisions and are representative, but not proscriptive, for other implementations.

State coordination and orchestration for Behaviors and Attributes are handled by standalone microservices using publish/subscribe messaging to observe states and coordinate actions by other services. State coordinators determine the proper responses to emulate RPC processes between web2 and web3 systems, though the actual implementation is always asynchronous. Messaging-based state coordination brings verifiable and trustable orchestration to the ecosystem and delivers a dynamic range of possible response paradigms (e.g. request/reply, worker queueing, complex event processing, fanout, etc.). Since the methodology, algorithm, and data chain are themselves verifiable, auditable,

[17] Esaiyo Trace uses the Apache Avro (Group, 2022) serialization libraries and protocols for persistence, marshaling, and validation.

[18] For the initial MVP, Trace leverages the Moralis (Moralis, 2022) toolkit for constructing cross-chain movement and asset creation, metadata acquisition, and other tasks. This is not to say that the solution assumes Moralis as a requirement, but the methods implemented in the framework do dictate additional implementation decisions at the time of this document.

and available to third parties, the state coordinators can function as "finite state lookup" systems for state automata on-chain (Hopcroft et al., 2007; Rich, 2008).

4. Considerations and Competition

There are many bridge technologies for moving assets between chains and platforms with more coming online all the time. We look at just a few examples in this section, with an eye towards common approaches and methods of exchange. Notably, the focus of these bridges is on token exchange and migration safety. And as a result, the general *modus operandi* for bridging between blockchains tends to follow these steps with varying degrees of automation (depending on the maturity and focus of the project). These steps, in general order, are:

1) Contribute and lock source tokens into a bridge contract located on the source blockchain;
2) Once source tokens are locked, new tokens are minted on the destination blockchain and are locked;
3) A variety of verification procedures are taken to validate intent, funds, commitment, minting, and other factors on the bridge itself. Generally these involve multiple signature validations and proofs to ensure that the transfer is legitimate, funded, and otherwise possible;
4) Once the new tokens are minted and the validation procedures successful, the lock is released on the source blockchain and source tokens are burned by the source contract;
5) Once burned, the destination tokens are unlocked and sent to the recipient wallet on the destination chain.

Sometimes the steps vary in their order, and some steps can be worked in lockstep depending on the functionality of the bridge, but the steps are generally the same. Blockchain security audits focus on demonstrating the reliability of the logic and how tamper-proof the protocols at work might be (Cross-Chain, 2022; ImmuneBytes, 2021).

What is missing is a coherent audit trail of all of the steps across both the source and destination. That is to say, that there is no clear picture of the intent, execution, and completion of steps 1-5 in any single, trustable location. Where there are events emitted from bridge contracts, those events are not backed on-chain and are often not available once the bridge operations complete. This audit gap is one of the key problems Trace solves.

4.1. Layer 1 Bridges

4.1.1. Binance Bridge v2

The latest bridge offering from Binance connects both custodial accounts as well as general wallets into the ecosystem. The process for asset transfer mirrors the steps laid out above, with the capability to move NFT tokens coming later. The Binance Bridge does not offer a view into the behavior of pre-transfer, on-bridge, and post-transfer operations either within custodial wallets or in external integrations. There is no record of aborted, missing, or otherwise incomplete activity (Binance, 2022a, 2022b).

4.1.2. OpenSea NFT Transfer

Perhaps the most well-known of the multi-chain NFT marketplaces, OpenSea does grant users the ability to transfer NFTs on-chain using wallet or custodial services (OpenSea, 2022). OpenSea allows migration of NFTs between supported chains (EVM and non-EVM) using custodial services. Neither method provides audit functionality for pre-transfer, on-bridge, and post-transfer operations. When the transfers use custodial, on-platform methods, OpenSea metadata and NFT characteristics are maintained across the transfer. There are no guarantees that attributes are preserved when leveraging other wallet or platforms to initiate and manage the transfer. There is no guarantee that NFT metadata which is maintained by OpenSea off-chain is tamper-proof or immutable, and there is no record as to whether or not (and when) alterations to that data are made.

4.1.3. Boba Network

Operating in the middle of Layer 1 and Layer 2 solutions[19], the Boba Network leverages an Optimistic Rollup[20] functionality from Optimism to provide native NFT movement between Optimism and Ethereum. Notable for its asymmetric approach to asset movement, Ethereum to Boba uses a locked token and rollup chain while Boba to Ethereum relies on cryptographic commitments (as in step 3 above). There is no audit log for intent, execution, or settlement. Validation relies on block verification mechanisms only and does not account for off-chain data (Boba, 2022; Optimism, 2022).

5. Summary

Verifiable behavior and the complete view for assets are the primary objectives for Esaiyo Trace. Bringing the same "look and feel" to unique digital assets that we have come to expect from their physical counterparts bridges the web3 world into the spaces we experience everyday.

Objects are defined and differentiated by time and space, origin, materials, experiences, sequences, groups, and interactions. This is true of physical and digital objects, as well as intangible concepts like expressions, ideas, and theories. Digital assets in the Esaiyo ecosystem can be unique creations on their own or digital representations of objects in the physical and metaphysical worlds. SIO[21] coding gives assets a unique existence of their own — distinct from any physical representation that exists. By keeping the complete view of the asset, its history, attributes, intentions, and

[19]While there is some debate, especially amongst infrastructure and ecosystem development tools, as to what constitutes "layers" for the web3 stack, in this document we conform to the standard generally accepted by the web3 user community. For a general and acceptable reference, see (Coinbase, 2022).

[20]Simply put, optimistic rollups are those where the consensus layer or mechanism from another (generally "parent") blockchain. This mechanism provides faster block sealing while trading off block-to-block security in most cases (a "rollup" of transactions). There are many methods and implementations, but for our Boba example, the Optimism network is used to demonstrate the idea here. (Optimism, 2022)

[21]The SIO code is shorthand for the "social identity of objects" identifier which is the central organizing method for all objects (real or virtual) under observation in Esaiyo Trace. It provides the foundation for events which span platforms, blockchains, implementations, uses, and any variety of other behaviors for an object. This whitepaper does not delve into the information and ontological theory behind the principles of the SIO. It suffices to point out that

behaviors on-chain, Esaiyo Trace enables novel approaches to value — beyond the simple economic models available in cryptocurrency paradigms. These values can include information, history, reputation, social, intrinsic, extrinsic, and relational measures. And all valuable aspects of the object can be measured and utilized simultaneously.

On-chain storage of these assets and their legacies finally realizes one of the core promises of blockchain and web3 technology generally — to bring the equity, fairness, and trustability of blockchain platforms to bear on the value of objects — real or virtual — in a manner which is verifiable and accountable for everyone.

6. References

This section includes a selected bibliography supporting the whitepaper.

Antonopoilos, A. M. (2016). *The Internet of Money*. Merkle Bloom LLC.

Antonopoulos, A. M. (2017). *Mastering Bitcoin : programming the open blockchain* (Second edition. ed.). O'Reilly.

Arsheep Bahga, V. M. (2017). *Blockchain Applications: A Hands on Approach*.

Binance. (2022a, 2022-03-29). *Bridge V2*. Binance. Retrieved 12 May 2022 03:22 UTC from `https://docs.binance.org/smart-chain/guides/bridge-v2.html#swap-tokens-from-binance-chain-to-a-different-network`

Binance. (2022b, 2022-03-29). *Introducing Binance Bridge 2.0*. Binance. Retrieved 12 May 2022 03:22 UTC from `https://www.binance.com/en/blog/ecosystem/introducing-binance-bridge-20-4214998246849036 26`

Boba. (2022, 2022-05-09). *Boba Developer Docs: Welcome to Boba*. Boba. Retrieved 12 May 2022 03:22 UTC from `https://docs.boba.network`

Cassandra. (2022). *Cassandra Architecture Overview*. Retrieved 05/19/2022 from `https://cassandra.apache.org/doc/latest/cassandra/architecture/overview.html`

Coinbase. (2022). *A Simple Guide to the Web3 Stack*. Retrieved 05/19/2022 from `https://blog.coinbase.com/a-simple-guide-to-the-web3-stack-785240e557f0`

Cross-Chain, B. (2022, 2022-02-10). *Cross-Chain Bridge LitePaper: Connecting Blockchains for Full Interoperability*. Cross-Chain Bridge. Retrieved 12 May 2022 03:22 UTC from `https://docs.crosschainbridge.org`

---

this is one of the foundational technologies for the Esaiyo Trace system on which much of the platform capabilities rests.

Group, A. (2022). *Apache Avro™ 1.11.0 Documentation*. Apache Group. Retrieved 02/21/2022 from https://avro.apache.org/docs/current/#compare

Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2007). *Introduction to automata theory, languages, and computation* (3rd ed.). Pearson/Addison Wesley. Table of contents http://www.loc.gov/catdir/toc/ecip0613/2006014263.html

ImmuneBytes. (2021, 2021-02-22). *Crypto Tokens & How to Audit Them*. ImmuneBytes. Retrieved 12 May 2022 03:22 UTC from https://immunebytes.com/crypto-tokens-how-to-audit-them/

Moralis. (2022). *Moralis API Documentation*. Retrieved 05/19/2022 from https://docs.moralis.io/introduction/readme

OCLC. (2022). *Linked Data*. https://www.oclc.org/research/areas/data-science/linkeddata/linked-data-outputs.html

OpenSea. (2022). *OpenSea API*. Retrieved 05/19/2022 from https://docs.opensea.io/reference/api-overview

Optimism. (2022). *How Optimism Works*. Retrieved 05/19/2022 from https://community.optimism.io/docs/how-optimism-works/#

Pierce, J. R. (1980). *An introduction to information theory : symbols, signals & noise* (2nd, rev. ed.). Dover Publications. Publisher description http://www.loc.gov/catdir/description/dover033/80066678.html Table of contents http://www.loc.gov/catdir/toc/dover031/80066678.html

Quick, S. R., & Kolster, A. (2014). *Comprehending Things: Ontology and Semantics for Event Handling IOT* Wolfram Data Summit, Washington, DC.

Rich, E. (2008). *Automata, computability and complexity : theory and applications*. Pearson Prentice Hall.

Szilágyi, P. (2017). *EIP-225: Clique proof-of-authority consensus protocol*. Retrieved 05/19/2022 from https://eips.ethereum.org/EIPS/eip-225#:~:text=Clique%20is%20a%20proof-of-authority%20consensus%20protocol.%20It%20shadows,effort.%20Motivation%20Ethereum's%20first%20official%20testnet%20was%20Morden.

Visha, C. (2021, 10 Oct 2021 15:30 UTC). *The Top Bridges for Interoperability with Ethereum*. Crypto Briefing. Retrieved 12 May 2022 03:22 UTC from https://cryptobriefing.com/the-top-bridges-interoperability-with-ethereum/

W3C. (2022). *Semantic Web: Linked Data*. https://www.w3.org/standards/semanticweb/data

Wikipedia, c. (2022, 8 February 2022 14:11 UTC). *Finite-state machine*. Wikipedia, The Free Encyclopedia. Retrieved 11 May 2022 03:22 UTC from https://en.wikipedia.org/w/index.php?title=Finite-state_machine&oldid=1070640030

Witek Radomski, A. C. P. C. J. T. E. B. R. S. (2018, 2018-06-17). *EIP-1155 Multi-token Standard*. Ethereum Foundation. Retrieved 11 May 2022 03:22 UTC from https://eips.ethereum.org/EIPS/eip-1155