LAOS: Vision for a Scalable, Bridgelessly Connected, Truly Non-Custodial, Dynamic NFT Protocol.

Alessandro Siniscalchi, Toni Mateos, PhD[*] and Alun Evans, PhD[†]
Co-Founders @ Freeverse & LAOS
{alessandro, toni, alun}@freeverse.io

November, 2023

Abstract

This whitepaper presents the vision and design for LAOS, a truly non-custodial dynamic NFT protocol that, leveraging its core features of bridgeless connectivity to all EVM chains, as well as second-order type of scalability, aims to establish itself as the consensus system that every other chain uses to scale their Digital Ownership transactions, instantly and securely, and without bridging. With an architecture planned for sharding-based scalability patterns, LAOS will enable currently mature ecosystems, such as Ethereum, to support applications that unlock the full power of digital ownership, migrating from the current mindset of scarcity and speculation, to one of abundance and captured User Generated Value. LAOS addresses the growing issue of custodianship, where companies hold users' NFT essential data to achieve asset mutability, thus assuming risks typically associated with securities, by providing a genuinely decentralized legislation-compliant solution.

1 Overview

LAOS aims to become the consensus system that every other chain uses to scale their Digital Ownership transactions, instantly and securely, without bridges, and in a truly non-custodial way, in contrast to highly centralized practices common today.

It introduces and leverages the powerful concept of bridgeless minting and evolution, whereby developers can benefit from the features of LAOS, from day 1, while remaining in the blockchain of their choice to do trading and use DeFi services as usual, with no bridges required.

LAOS enables DApp developers to build applications that unlock the full power of digital ownership, migrating from the common mindset of extreme *scarcity* and associated *speculative dynamics*, to one of *abundance* and captured *User Generated Value* (UGV).

LAOS is a fully open layer 1 blockchain, decentralized via usage of its own native utility token, built as a specialized Parachain in the Polkadot ecosystem, inheriting its security and advanced features. LAOS will replicate a part of Polkadot's architecture inside Polkadot itself, providing its own sharding-based scaling specialized in Digital Ownership.

Within Polkadot, it leverages its trustless connection to other Parachains specialized in smart contracts, such as Moonbeam and Astar, and data storage, such as Crust Network.

LAOS provides the foundations for companies and creators to protect themselves from legal issues regarding their assets being considered securities, by storing asset metadata on decentralized systems that can be verified on-chain. Committed to promoting awareness of the risks of centralized flows, LAOS collaborates with regulators to ensure blockchain digital ownership lives up to its promise, ensuring companies do not become custodians of the NFTs they create.

Section 2 provides an introduction to the current landscape of NFT technology, focusing on some of the main issues that have led to limited use cases, and highly centralized practices. Section 3 introduces the key aspects and use cases enabled by Digital Ownership within LAOS. Section 4 provides an in-depth walkthrough of the core technology and architecture of LAOS, as well as the main actors involved and their incentives. Section 5 describes the patterns that allow LAOS to implement bridgeless minting and evolution. Section 6 presents an analysis of attack vectors and their potential mitigation. Section 7 concludes with a

[*] PhD in String Theory, University of Barcelona & Imperial College of London.

[†] PhD in Medical Physics, University College London.

summary of the main aspects presented.

2 Current NFT Landscape

2.1 Scalability and Use-Cases

During 2017, blockchain-based Non-Fungible Tokens (NFT) became popular within the crypto community, especially after the initial success of Cryptokitties (CK) [1]. CK and its successors largely set the mindset of the first wave of NFT-based applications; it is therefore important to understand the technical limitations that these had to put up with.

Despite the fact that, compared to non-blockchain games, CK was an extremely limited game, with only a handful of actions available to users, which were often executed within days of separation, it managed to collapse the underlying Ethereum blockchain [2], [3], [4] when peaking around 15K daily active addresses, accounting for 25% of all Ethereum transactions. Fast forwarding 5 years, another game, Sunflower Farmers, reached a 40% quota of all gas consumed by Polygon, a scaling layer-2 on top of Ethereum, with roughly 5x more activity [5], [6].

These severe scalability limitations inevitably impacted on the use cases initially implemented by different industries. On the one hand, *extreme scarcity* became the most exploited pattern, since minting and trading require on-chain usage [7]. While prices of real-world products necessarily derive from their production and distribution costs, the price of the first wave of blockchain digital items was largely unrelated to such costs. Artificial digital scarcity was used to increase initial selling price, and led to purchasers *speculating* on the possibility of reselling the same item for a greater profit (see [8] for an interesting mathematical analysis on the expectation of profits during the first wave of NFTs).

2.2 Mutability and Centralization

Another pattern that arose due to lack of scalability was that of considering NFTs as either *static* collectibles or as fully centralized digital goods; indeed, in many cases, as both. This pattern was initiated with the rapid adoption of the ERC721 standard [9], which contains an *optional* method that returns a Universal Resource Identifier (URI) for every token:

$$\text{tokenURI: tokenId} \rightarrow \text{string} \tag{1}$$

The standard suggested the following use: ”This allows your smart contract to be interrogated [...] for details about the assets which your NFTs represent”.

While other methods in the ERC721 standard, such as *ownerOf*, allow the blockchain to be queried about *who owns* an asset, the optional *tokenURI* method aims at being queried about *what is owned.*

The vast majority of DApps ended up choosing one of the following approaches:

$$\text{tokenURI: tokenId} \rightarrow \text{IPFS address} \tag{2}$$

$$\text{tokenURI: tokenId} \rightarrow \text{private URL} \tag{3}$$

The first choice (2) is often made for static, immutable, NFTs that represent assets of high value; a representative example is Beeple's *Everydays—The First 5,000 Days* [10], sold at Christie's for $69.3M [11]. In this case, the answer to *what is owned* refers users to the Inter-Planetary Filesystem (IPFS [12]), which has the following two essential properties:

- it is a *content addressed* system, whereby the address of a digital file is the result of applying a certain Hash function H: IPFS address = H(content);
- it is decentralized in nature, due to its peer-to-peer protocol.

When using the IPFS pattern (2), the first property ensures that the blockchain returns an address that uniquely describes the content of the NFT. Together with the second property, the result is that *what is owned* can be answered in a fully trustless manner, without the need to query any trusted 3rd party.

The downside of pattern (2) is that it leads to a clash between asset mutability and scalability. Since any minor change in the asset's content would necessarily map it to a different IPFS address, an on-chain transaction would need to be used on every change of every asset. Thus, pattern (2) is mostly used for immutable NFTs.

The centralized pattern (3) has been the choice for many DApps, especially those that required some degree of asset mutability. It allows developers to minimize the use of the blockchain, by simply storing a link to external, privately-owned URLs, which host *what is owned*, the asset metadata. A representative example is Sorare [13], a popular web3-based sports fantasy game, where the data of the NFTs is stored in the company's private servers[1]

With the centralized pattern (3), mutability is simply achieved by changing the data provided by each company's private servers. This has several drawbacks. The most important consequence is a critical loss of the core principles of blockchain technology, as users must rely on private companies for an

[1] As a concrete example, at the time of writing, the blockchain method *tokenURI* for a Lionel Messi NFT [14] pointed to this Sorare's owned URL [15]

understanding of their own holdings, thereby undermining decentralization. Furthermore, these companies can change such data without leaving any trace, to the extreme of making it unavailable. Such events can even happen involuntarily, for example, via hacks, or because a company ceases operations, or is forced to. Private companies become the custodian of the attributes, and by extension, the value of the NFT, and assume the risks associated to it. We shall refer to this particular highly centralized practice as the *elephant in the room.*

2.3 Insecure Bridges

Finally, the aforementioned limitations have also forced DApp developers to make undesired compromises when selecting the blockchain to build on. Concerns around gas costs, availability of the underlying coin, user base of the existing DeFi ecosystem, or degree of decentralization and security are weighted against the needs of each specific application.

Bridges are often built to allow users to migrate assets across blockchains, which tends to be a complex UX process. But building bridges is hard. Almost every relevant bridge built to date requires some level of centralization, and many of them have been hacked [16], resulting in millions of stolen funds. Examples include widely used bridges between Ethereum and Solana [17], bridges required to play *Axie Infinity*, the most popular NFT-based game at the time [18], [19], as well as vulnerabilities of up to $850M that were live for some time in the Polygon to Ethereum bridge, but fixed before being exploited [20].

2.4 Growing Legal Concerns

As discussed, initial limitations of fully programmable-blockchain technology led to many of the early NFT use cases relying on some combination of extreme scarcity and speculation, making choices between an immutable approach, or highly centralized practices, and often forcing users to go through insecure bridges.

Legislation is indeed starting to pay closer attention to centralized NFT flows. A relevant example is the recent class-action lawsuit filed against the popular NFT-based application NBA Top Shot [21], to decide whether or not their NFTs should be considered unregistered securities [22]. Although this lawsuit does not deal with the *elephant in the room* directly, the judge did argue that users were forced to rely on the running of the platform by a few private actors, leading to *expectation of profit being "derived from the efforts of others"*.

Although legislation in the web3 space is a highly complex, rapidly evolving field, it may well be the case that similar NFT-as-securities lawsuits follow soon, given the aforementioned custodian role that many private companies are currently assuming.

3 Digital Ownership within LAOS

LAOS Assets (LA) are Non-Fungible Tokens that can be created and evolved on the LAOS blockchain, independent of the blockchain that manages their ownership. LA extend first-generation NFT technology by introducing the following features:

- Bridgeless Connectivity: all aspects around ownership of an LA, including trading, lending, access to DeFi, etc., can be managed in any blockchain that supports smart contracts, while minting and evolution is managed in LAOS, without the need to resort to any type of bridge (technical details in Section 5);

- Scalability. LA can be minted and evolved at scale, in fully decentralized flows;

- Full Decentralization. The attributes of all LA, now, and in every past state, can be certified on-chain, without resorting to centralized privately-owned servers (technical details in Section 4).

The following subsections provide a high-level overview of how these properties can be leveraged, focusing on the developers and users point of view. The low-level technical aspects are left for sections 4 and 5.

3.1 User Generated Value

Scalability and the trust generated by the fact that mutability has a fully verifiable history are the key ingredients to one of the main paradigm shifts that LAOS Assets enable.

LA enable digital industries to migrate from a mindset of *scarcity* and *speculation*, to one of *abundance* and *User Generated Value* (UGV). In the latter, assets can be distributed at scale, at zero or very small initial value (e.g., just for downloading and aplication or for signing-in), to a large base of users, hence eliminating all monetary entry barriers to owning a digital asset.

In this mindset of abundance, the initial market value of most assets is effectively zero. DApps using LA can provide the flexibility to evolve based on both off-chain activity (such as real-world events) and on-chain data. By engaging with apps, video games, online and social media activity, and the broader ecosystem, LA owners can improve their assets (a more powerful game item, an NFT that grants better rewards, an IP license granting further rights), making them more valuable to others, eventually converting their dedication, talent, and effort, into increased market value.

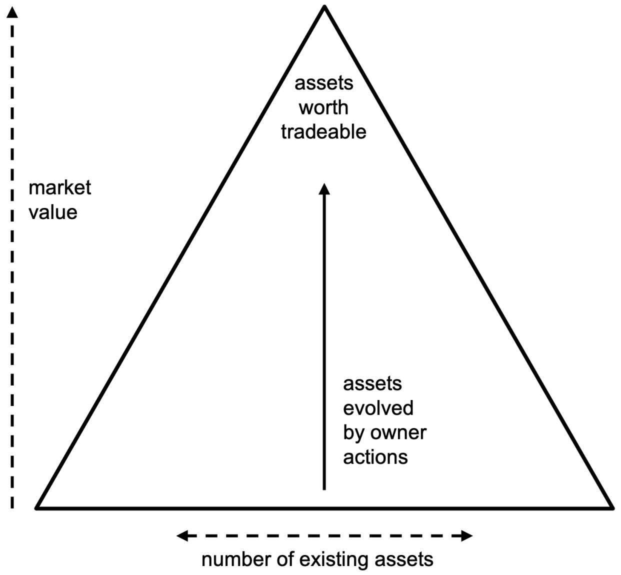

Figure 1: Typical distribution of market value (y-axis) vs. number of assets (width of the x-axis) in User Generated Value paradigms. A large number of assets with negligible market value co-exist alongside a smaller percentage that have accrued significant value through the efforts of their owners.

UGV paradigms tend to reflect patterns common to Free-to-Play models (Figure 1), whereby a large base of assets with near-zero market value co-exist with further layers of assets which users have evolved significantly with their actions. This pattern turns the sentiment of owning digital assets into an *active* experience, becoming a powerful engagement tool.

3.2 Certifiability

UGV and the use cases mentioned in sections 3.3-3.4 heavily rely on the history of the asset, on the confidence that NFTs have market value accrued through a fair, traceable, set of evolutions. It becomes paramount to be able to answer, in a fully trustless way, questions like *how did users turn this game item into something so powerful?*, or *how did the owners of this NFT evolve it from scratch to an item with level-10 rewards?*

This is manifestly impossible in centralized flows derived from pattern (3), since external privately-owned servers need to be trusted, exactly as in the pre-blockchain era.

LAOS provides the strongest forms of certifiability. On the one hand, as described in Section 4, all *data is fully available* via decentralized flows, including data about the current and past attributes of every asset.

On the other hand, LAOS enables any party to prove on-chain that an asset with tokenId id was in a certain state s at any time t in its history, via the provision of inclusion proofs $\Pi(id, s, t)$; these allow the blockchain to build *verify* methods which return true/false when queried:

$$\text{verify:} \qquad id, s, t, \Pi(id, s, t) \quad \to \quad \text{bool}\,. \qquad (4)$$

By virtue of LAOS being in the Polkadot ecosystem, it will be possible to use these strong forms of certifiability as part of smart contract logic built in other Parachains, such as Moonbeam [23] or Astar [24]. Example use cases would be contracts that allow owners of NFTs that have been evolved beyond a certain level to unlock certain rewards, or that execute methods of a DeFi contract in the same blockchain. An illustrative example of pseudo-code would be:

```
Data: Input values $id, s, t, \Pi(id, s, t)$
do initial logic;
if verify($id, s, t, \Pi(id, s, t)$) then
    do further logic based on $s, t$;
end
```

Perhaps the simplest, yet powerful, example of this usage would be to enable *certified purchases*. In this pattern, a buyer could sign a purchase transaction that is conditional on the asset attributes, e.g., "buy this asset if and only if it has *level* larger than 100".

3.3 Bridgeless Minting & Evolution

The scalable, strongly-certifiable, minting and evolution features of LAOS can be added on top of the advantages that leading blockchains already have in terms of mature ecosystems of users and DApps.

By means of example, we hereby discuss the use case of developers that want to build applications which benefit both from LAOS features and Ethereum's rich ecosystem of smart contracts and DApps. Any other blockchain that supports Turing-complete programming of smart contracts would work too.

Developers only need to deploy once an ERC721/1155 smart contract in Ethereum, tuned to use the Universal Location pattern as described in section 5. Next, they can mint and evolve assets in that contract by simply executing the corresponding transactions in the LAOS blockchain. Anyone can permissionlessly, in a fully decentralized manner, and without resorting to bridges, build new, or connect to existing, marketplaces that operate with these assets by means of standard ERC721/1155 calls on Ethereum. In particular, this contract's assets can be connected, e.g., to smart contracts in Ethereum that allow their rental or to fractionalize their ownership.

By doing so, applications can leverage paradigms where massive amounts of assets are created in a scalable, certifiable way, in LAOS, with only a small fraction of them having fairly accrued enough User Generated Value to be worth being traded in Ethereum.

3.4 Use cases

LAOS profoundly shifts traditional paradigms, heralding new prospects for creators, users, and industries at large. Through its bridgeless connectivity, it allows these new paradigms to emerge within the most mature ecosystems currently available, including Ethereum.

By actively engaging with LA, owners wield the power to shape and influence the attributes of their assets, thereby directly impacting their intrinsic value. This interactive participation fosters deeper connections between users and their assets, transcending the conventional notion of passive ownership.

The reimagining of digital ownership opens up a wide range of compelling use cases. With developers and their communities empowered by new tools and increased flexibility, it is impossible to predict the full extent of their creative potential. Nevertheless, here are a few examples that can be envisaged today.

Gaming. Using LAOS bridgeless minting and evolution, games can mint hundreds of millions of assets on Ethereum, at minimal cost, allowing gamers to trade, lend, and amplify their assets using Ethereum's vast applications.

Legendary Collectibles. Existing collections are given a new lease of life, with their creators using LAOS to extend and evolve their initially static image and metadata, e.g., creating seasonal campaigns. Marketplaces and explorers show the past and current states easily, ensuring the collection's year-round relevance and continued value.

Marketplaces. Leading NFT marketplaces attract more users by offering mass minting on Ethereum through no-code and API solutions. They absorb the minimal gas costs for all users meeting specific criteria and relays transactions to provide a gasless UX.

Interoperability. Games boost user acquisition and retention by letting players use assets from any blockchain. Users pay for imports via in-app purchases, after which the developer uses LAOS to permissionlessly extend metadata to match the game's style and introduces new attributes that are evolved within the game.

DApps. DApps protect themselves from legal issues regarding their assets being considered securities, by shifting their minting and evolution to LAOS. Now, rather than using private servers, the assets are stored securely on decentralized systems and can be verified on-chain.

User Generated Content. LA allows the flourishing community of content generators to accrue market value to their talent and dedication. In a game or online 3D crafting app example, users create a fully customized racing car, from engine tweaks, to paint job and looks, and use LAOS' decentralized asset identity to incentivize games to import them and provide further utility.

Game Distribution Platforms. Leading platforms partner with Copyright Offices to enforce asset copyright across all blockchains, using LAOS' Decentralized Identity.

4 Architecture

The architecture of LAOS is discussed in detail in Section 4.2, after an initial discussion of a key separation pattern of asset ownership and asset attributes used all across LAOS in Section 4.1. Section 4.3 concludes with the vision beyond the initial implementation.

4.1 The Ownership-Attributes Split

At the core of the technology is a hard separation between asset ownership and asset attributes, brought to the extreme whereby both sets of data can live and be managed by different blockchains, but carefully designed to be properly linked, and allow for on-chain traceability and certification.

Unlike traditional NFTs, where both sets are either stored on-chain in the same blockchain's contract, or where the attributes are stored in external, static, content-addressed systems (pattern (2)) or in external, privately-owned servers (pattern (3)), LAOS separation allows both sets of data to live in different decentralized *consensus systems*.

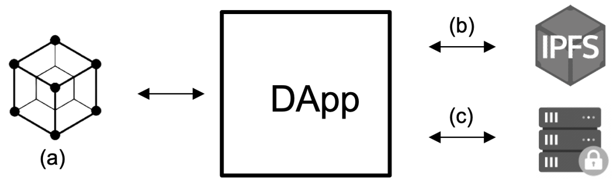

Figure 2: Typical structure of nodes/services required to run a DApp, such as a marketplace: (a) a consensus node, e.g., an Ethereum node; (b) an IPFS node, for NFTs implementing pattern (2); (c) access to privately-owned internet servers, for NFTs implementing pattern (3).

Figure 2 illustrates the traditional NFT node structure to which DApps normally connect. While DApp developers have the choice to operate their own consensus nodes, particularly for fully permissionless blockchains, it is commonplace to rely on third-party node providers that ensure consistent uptime and handle maintenance [25]. Likewise, developers often opt to use third-party IPFS providers to guarantee that content is pinned with high data availability.

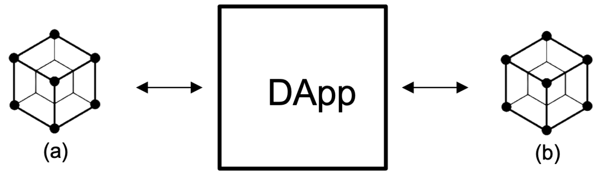

Figure 3: Structure of nodes when implementing the ownership-attributes split, with a DApp using two different consensus nodes, one for ownership (a) and one for attributes (b).

Figure 3 shows the structure for patterns based on LAOS' ownership-attributes split, whereby only consensus nodes are required. Figure 4 shows a convenient architecture where the DApp only interacts with one single node, which can be built permissionlessly: the node maintains a state that internally syncs with the two consensus system, and exposes a single API that abstracts all complexity, and mimics a single ERC721/1155 compliant system.

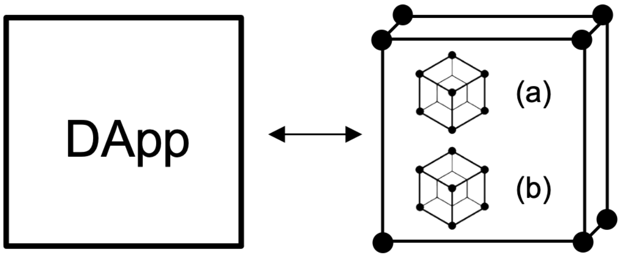

Figure 4: A convenient way to structure nodes in ownership-attributes split patterns, where a DApp interacts via the ERC721/1155 standard interface with one single node that abstracts all complexity.

The link between both consensus systems can be achieved by using recently developed standards like the *Universal Location* pattern, introduced as part of XCMv3 by Polkadot, and described in more detailed in section 5.1. From the point of view of the patterns described in section 2.2, this standard enables the DApps to reference assets that live in different consensus systems:

$$\text{tokenURI: tokenId} \quad \rightarrow \quad \text{Consensus Address} \qquad (5)$$

This fundamental aspect, as well as the design to link both systems, is reflected the LAOS architecture detailed in the next sections, and sits at the core of the bridgeless minting & evolution feature discussed in Section 5.

4.2 High-level architecture

4.2.1 Leveraging Polkadot's ecosystem and features

LAOS is architected as a specialized Parachain in Polkadot [26], [27], fully devoted to providing the ideal platform to fulfill the aforementioned requirements and vision. The architectural design choices exploit several benefits that are unique to Polkadot, among others:

- as a Parachain, it will automatically inherit the full security guarantees provided by Polkadot's Relay Chain from day 1;
- by internally being built on Substrate and using the GRANDPA[2] consensus algorithm, it will also share top features such as non-probabilistic fast block finality, and forkless upgrades of its protocol;
- it will make use of Polkadot's recently introduced Universal Location pattern, as part of XCMv3;

[2] GRANDPA stands for GHOST-based Recursive Ancestor Deriving Prefix Agreement, a finality gadget for consensus systems originally presented in [28].

- it will heavily leverage other Parachains in the ecosystem, making it easy to interact with EVM-compatible smart contracts, guarantee long-term data availability, access to DeFi, and benefitting from a combined maximum transactions per second (TPS) estimated between 100K and 1M;

- it will allow its internal design to leverage Polkadot's developed Relay Chain and trustless bridges technology to achieve, mid-term, a scalability comparable to second order relay chains.

As side benefits, Polkadot's available rich set of tools and open code bases will allow LAOS to avoid reinventing the wheel, use battle-tested software, and progressively adopt upgrades developed by other teams, while contributing to the ecosystem.

4.2.2 Components

LAOS will by built around the architecture depicted in Figure 5. We hereby detail its main system components.

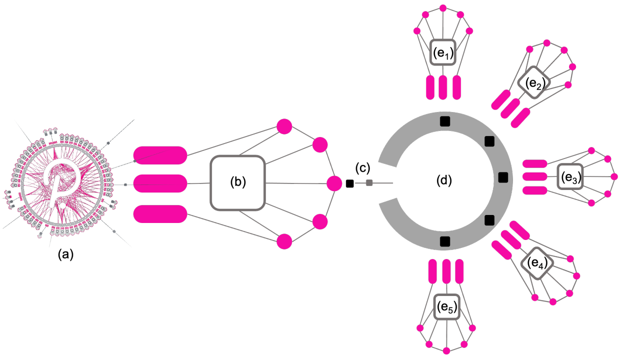

Figure 5: Vision for the core design of LAOS: the LAOS Relay Chain (d) is connected to the Ownership Parachain (b) via one single trustless bridge (c), while its set of validators provide security to the Evochains (e_i) via the standard Polkadot Relay Chain to Parachain sharding pattern.

(a) Polkadot's Relay Chain. At the root of the LAOS consensus protocol is Polkadot's Relay Chain, with the set of Polkadot validators that provide security to the Parachain, ensuring that every transaction processed by a Parachain is valid. The Relay Chain also help Parachains transfer assets and connect functionalities in a fully trustless way via native Cross-Consensus Messaging (XCM), efficiently linking the light clients that Parachains have from other Parachains.

(b) LAOS Ownership Parachain. The LAOS Ownership Parachain will be a specialized chain, based on the Polkadot standard NPoS protocol, with the following functionalities:

- host and manage the LAOS native utility token;

- manage the ownership of all LA created directly in LAOS, with transactions implemented at the protocol level, and fees paid in the native tokens;

- manage runtime upgrades of LAOS' Relay Chain, and all Evolution Chains (Evochains);

- store state roots of the Evochains, exposing asset attributes certification methods, and reward LAOS Relay Chain validators upon reception of new correct roots;

- manage all trustless transfers of LAOS Assets and LAOS tokens between LAOS and the rest of Parachains via XCM;

- implement and coordinate governance and staking.

(c) GRANDPA-Based Trustless Bridge. LAOS will make use of the bridge technology developed by Polkadot ecosystem to trustlessly connect Substrate-based chains with GRANDPA finality [29], which is expected to connect, among others, Polkadot and Kusama via BridgeHub [30].

The security of this bridge design relies on each chain's set of validators running a light client of the other chain; this enables each endpoint to mathematically verify any statement received about the state of the other chain. In particular, new headers are accepted only after verification of the required majority of validators of the other chain.

The bridge will be used to teleport LAOS tokens among the Ownership, LAOS Relay Chain, and the Evochains, always with a 1:1 parity, to enable payments of gas fees for mint and evolve transactions, irrespective of whether they corresponds to assets that can be traded in the LAOS Parachain, or in other blockchains via remote, bridgeless minting and evolution.

Likewise, the bridge will be used to constantly record the state of the LAOS Relay Chain, and its Evochains, in the LAOS Ownership Parachain, allowing on-chain verification of all assets attributes, including every state they went through.

(d) LAOS Relay Chain. Connected to the Ownership Parachain via one single trustless bridge (c), and providing security to the Evochains (e_i) sits a specialized, limited-functionality, version of Polkadot's Relay Chain. Both chains at the end of the bridge run a light-client of each other, ensuring that their full states are mutually known in a permissionless way.

The LAOS Relay Chain runtime upgrades are orchestrated by the Ownership chain via XCM commands across the trustless bridge. Likewise, this Relay Chain will not have a native token; it's economy will use LAOS tokens reserve transferred from the the Ownership Parachain.

In turn the LAOS Relay Chain will provide security to its Parachains following exactly the same pattern as Polkadot's Relay Chain.

(e) LAOS Evochains. LAOS Assets are created and evolved in the Evochains. They are Parachains of the LAOS Relay Chain, supporting asset mint and evolution at protocol level.

All Evochains share the same runtime, and runtime upgrades are orchestrated by the LAOS Relay Chain, the runtime of which is in turn orchestrated by the Ownership Parachain. Similarly, Evochains do not have a native token. DApp developers that create and evolve LA pay transaction fees to Evochain collators in teleported LAOS tokens.

Each Evochain supports a large number of DApps, and new Evochains are spawned when approaching full capacity, as opposed to simply increasing gas price. When Polkadot Parathreads and on-demand patterns are fully consolidated, LAOS will consider integrating them as an additional complementary pattern.

Note that the trustless bridge ensures that the state of every Evochain remains in the Ownership Parachain, allowing for certification of every attribute of every LA, via standard Merkle proofs.

4.3 Scalability

The described architecture does not attempt to reproduce a *Polkadot-inside-Polkadot* 2nd-order-relay pattern, but rather, to constitute a specialized system, focusing on scaling homogeneously, supporting Evochains managed by exactly the same runtime.

As usage grows, LAOS stands to benefit directly from Polkadot's sharding technology, which enables the spawning of new Evochains, while maintaining their security through a single set of validators.

LAOS overall throughput, typically measured in transactions per second, is expected to align with Polkadot's own estimates as the combined throughput for all its Parachains.

5 Bridgeless Minting & Evolution

The architecture described in section 4 supports a pattern capable of enabling the minting and evolution of assets in LAOS, while their trading (and any other ownership-related concept) is managed in any other blockchain of choice, as long as it supports smart contracts; this applies to both EVM and non-EVM compatible cases. We stress that this does not require any type of bridge, neither trusted nor trustless.

We first introduce the concept of Universal Location in Section 5.1, and then describe a simplified version of the pattern which only allows bridgeless evolution (Section 5.2). We conclude discussing the full-fledged flow for bridgeless minting and evolution in Section 5.3.

5.1 Universal Location

One recent standard that LAOS will leverage is that of *Universal Location*. It was introduced by Polkadot as a natural evolution of the *MultiLocation* standard that was already in use within their ecosystem, as part of their Cross-Consensus Messaging (XCM) pattern [31].

The fact that the Relay Chain serves as consensus parent for all Parachains has enabled the design of trustless message-passing patterns among Parachains, for example, by relying on the light clients that collators of Parachains have from other Parachains.

MultiLocation assists this communication by identifying every single location that exists within the world of consensus systems that share a common parent, and example being all Parachains, as their finality derives from the consensus in the parent Relay Chain. A location in a consensus system is defined as an *isolatable state machine* held within global consensus.

For instance, the following:

$$\begin{aligned}&\text{MultiLocation} = \\ &\text{../Parachain(42)/AccountKey20}(0x12...cd)\end{aligned} \tag{6}$$

can be used by one Parachain to refer to a 20-byte account $(0x12...cd)$ that exists within a different Parachain (42), that shares the same parent consensus system: the Relay Chain (at '../').

By contrast, Polkadot's XCMv3 [32] introduced the concept of a *Universal Location* under which all systems which generate their own consensus exist, and by extension, all possible locations within consensus. The Universal Location is the only location that has no parent. In typical filesystem formatting, the Universal Location can be thought of as the *root*, often represented by the starting '/' symbol of a path.

Universal Location enables the creation of routes that connect smart contracts in an origin blockchain to assets or accounts in, e.g., Polkadot, through references such as:

$$\begin{aligned}&\text{MultiLocation} = \\ &/\text{Polkadot}/\text{Parachain}(42)/\text{AccountKey20}(0x12...cd)\end{aligned} \quad (7)$$

Finally, the *MultiAsset* part of the standard uses these concepts to standardize the referral to both fungible and non-fungible tokens.

5.2 Bridgeless Evolution

The simplest usage of LAOS for developers that want the ownership of assets to be managed by a different blockchain, e.g., Ethereum, is to simply mint assets in the origin blockchain, and use a LAOS MultiLocation string, using the Universal Location '/' as root, to specify the tokenURI, effectively using (7) to create a concrete implementation of pattern (5):

$$\text{tokenURI: tokenId} \quad \rightarrow \quad \text{LAOS MultiLocation} \quad (8)$$

As Figure 4 depicts, all that DApps need to build applications that use this pattern is to query a LAOS node to resolve tokenURIs and provide asset metadata.

This pattern basically links the asset metadata to an updatable slot governed by a consensus system, with transparency, traceability, data availability for both current and past states. And it does so in a flow which, ultimately, does not require of any trusted party.

In this pattern, however, one transaction per asset mint is still required in the origin blockchain, potentially running into scalability issues or incurring in high gas prices.

5.3 Bridgeless Minting & Evolution

The previous pattern can be extended to make a fully optimal use of the origin blockchain, to the extreme that one single transaction to the origin blockchain is required, on deploy time. After this initial transaction, *all minting and evolution can be offloaded to LAOS*.

We shall first discuss the nomenclature and the building blocks required to build this functionality, and then analyze the resulting flow.

This process requires two simple components: a pattern to build the *id* of assets, and a piece of smart contract logic that utilizes the meaning that can be extracted from such *ids*.

5.3.1 Encoding & Decoding Methods

The setup starts with the usage of a concrete pattern to generate *ids* for assets, enabling the encoding of a generic web3 address (*w3a*). Let us call *Enc* and *Dec* the encoding and decoding functions, correspondingly:

$$\begin{aligned} Enc:& \quad w3a, \Omega & \rightarrow id\,, \\ Dec:& \quad id & \rightarrow w3a\,, \end{aligned} \quad (9)$$

where Ω represents any other information that the *id* may contain, e.g., a consecutive index often used to distinguish assets as they are minted.

Enc is an injective function; in particular, two assets with different *id* must necessarily differ in either $w3a$ or Ω. Note, however, that there may be a large number of assets that share the same $w3a$, as made manifest by the *Dec* function, which projects out all information conveyed by Ω.

A simple example for an *Enc* function is provided by the binary concatenation of the binary representations of $w3a$ and Ω, which corresponds to a sibling *Dec* function that trims the bits assigned to $w3a$.

5.3.2 Smart Contract Logic

The ERC721/1155 smart contract logic required in the origin blockchain must contain a minor code extension in the implementation of the $ownerOf$ (and related) methods.

For the sake of simplicity, let us assume that the smart contract implements a simple method, $localStorage.wasEverTraded(id)$, that returns *true* if the asset has been traded *at least once*. On deploy of the smart contract, this method naturally returns $false$ for every *id*.

As customary, the contract writes to storage as assets are traded, allowing a method we shall call *localStorage.ownerOf* to return the new owners, and reflecting in positive returns of $wasEverTraded$. These methods can be used to extend $ownerOf$ as follows:

```
method ownerOf(id)
    if localStorage.wasEverTraded(id) then
        return localStorage.ownerOf(id);
    end
    else
        return Dec(id);
    end
```

5.3.3 Resulting Flow

The final ingredient required for bridgeless minting is implemented in LAOS, by simply using exactly the same encoder/decoder functions (9) to name assets in the Evochains, and declaring assets in these particular collections as *foreign assets*, signalling that they cannot be traded in the LAOS Ownership Chain.

The aforementioned smart contract logic, together with the LAOS MultiLocation pattern (8), allows one single deploy of an ERC721/1155 contract in an origin blockchain to link all assets to LAOS, while retaining the management of all ownership-related aspects in the origin chain. In effect, all assets ever assignable to all potential initial owners are reserved on deploy, and the LAOS consensus system is assigned the responsibility to *fill-in* the corresponding initially-empty registers as assets are minted, and modify them as they are evolved.

Again, figure 4 illustrates a convenient way to architecture nodes that would serve bridgeless minting and evolution; in this case, the ownership node would be one of the origin chain.

6 Security

LAOS core components are based on software and consensus protocols that are already in use, and analyzed extensively by other teams. In this sense, LAOS security analysis is different from projects proposing entirely new, still untested, protocols; in LAOS case, the analysis mostly consists of, for every component in its architecture, relying on previously published comprehensive studies of their attack vectors and potential mitigation.

6.1 Ownership Parachain

The Ownership Parachain will implement the default Parachain pattern that relies on the Polkadot Relay Chain to provide block finality [33], [34]. This protocol enables efficient heterogeneous sharding, and has been live in mainnet, in Kusama first, since October'19, and then in Polkadot, since March'20.

Parachains are free to decide their runtime and are encouraged to focus on specialization, in contrast to the one-runtime-fits-all homogeneous sharding being built in Ethereum 2.0. LAOS will diligently adhere to this pattern, specializing in all aspects relevant to Digital Ownership. The Ownership Chain will use the default Parachain pattern: Nominated Proof of Stake (NPoS) consensus algorithm, with GRANDPA providing finality over BABE block production. We refer the reader to the thorough theoretical analysis presented in [27] for full details.

It is essential to note that despite the freedom that Parachains have to design their inner details, every single transaction of every Parachain is eventually verified by the required majority of validators in the Relay Chain before being included in a Polkadot block.

Since the consolidation in Polkadot of Parachain's invalid transactions is prevented by the Relay Chain, the only significant attack vectors relevant to Parachains are related to censorship and data availability.

Censorship corresponds to Parachain collators not including one or more transactions based on criteria different from those specified by the protocol, e.g., based on the sender address. To prevent these attacks, a Parachain only needs to ensure that there are some neutral collators, but not necessarily a majority. Theoretically, the censorship problem is solved with having just one honest collator, as analyzed in [35].

Data availability attacks appear if the Parachain collators collude to withhold block data, or the corresponding Proof of Validity (PoV), to the validators in the Relay Chain. Polkadot's protocol mitigates this attack severely by using Erasure Coding [36], which adds redundancy on the block/PoV data made availability by a minimal subset of honest collators, and enables Relay Chain validators to reconstruct its entirety out of a small set of randomly queried data.

In summary, the standard Parachain-in-Polkadot level of security protects all features assigned to the Ownership Parachain, including transfers of LAOS assets and tokens among owners as well as to sibling Parachains, management of the LAOS token, transaction fees, treasury funding, storage of LAOS Relay Chain and Evochain states, and management of their runtimes.

6.2 Trustless Bridge

The design of the bridge that will connect the Ownership Chain and the LAOS Relay Chain is built around the same technology that will connect Polkadot and Kusama via BridgeHub [30].

This bridge design decouples the consensus mechanism and the state machine, requiring both networks to run *on-chain consensus clients*, also known as *on-chain light clients*, that only track consensus proofs of state transitions instead of the full state transitions. References [37] and [38] provide a good introduction to known attack vectors to light client based bridges in systems that either do not track the signatures of all validators, or do not implement accountability on validators that provide multiple signatures over different block proposals.

Instead, the bridge implementation is based on the on-chain light client used by Polkadot and Kusama [29], designed only for systems whose finality is provided by the GRANDPA finality gadget. Eventually, the bridge will become more efficient when

Polkadot integrates *on-chain accountable light clients*, which use signature verification via Zero-Knowledge schemes, and introduce on-chain accountability mechanisms. We refer the reader to [39] for a thorough analysis of attack vectors.

Regarding data withholding attack vectors, it is important to consider that any actor can relay block data and the corresponding transition proofs between the Ownership Chain and the LAOS Relay Chain. The attack can be straightforwardly mitigated by having at least one actor with incentives in the LAOS platform running a node of the LAOS Relay Chain, even if in read-only mode, capable of relaying the Evochain data as new blocks are produced.

Moreover, the LAOS Relay Chain validators have their own incentives to relay this data because they receive rewards in the form of LAOS tokens upon inclusion of new valid blocks in the Ownership Chain.

6.3 LAOS Relay Chain

The LAOS Relay Chain is an instance of Polkadot's Substrate-based Relay Chain, with GRANDPA finality. As such, its security depends on the standard game-theoretical incentive system that leads to more than two-thirds of their validators following the protocol honestly.

A comprehensive analysis of GRANDPA, including a listing of attack vectors and their potential mitigation, can be found in [28].

LAOS Relay Chain validators will be rewarded for including valid blocks from the Evochains, through transaction fees in LAOS tokens obtained from the Ownership Chain via the trustless bridge. Likewise, they will be rewarded when new block headers of the LAOS Relay Chain are correctly included in the Ownership Chain, in this case, in native LAOS tokens. This incentive system encourages LAOS Relay Chain validators to not only provide data to the bridge relayers but also potentially take on the role of bridge relayers themselves.

6.4 Evochains

Evochains are Parachains of the LAOS Relay Chain, supporting asset mint and evolution at protocol level. As such, the same security considerations discussed for the Ownership Parachains (Section 6.1) apply to the Evochains, replacing Polkadot's validators by LAOS Relay Chain validators.

7 Conclusions

We have presented the vision and design for a truly non-custodial dynamic NFT protocol, ready to scale any existing chain that supports smart contracts, and enable fairer, main-stream, models around digital ownership, such as User Generated Value patterns.

The platform fully embraces the Polkadot ethos of excelling in a specific domain while achieving exceptional performance and scalability. It leverages the rich feature set offered by other Parachains within the ecosystem, benefiting from the inherent trustless interoperability provided by Polkadot.

LAOS provides the foundations for a fully legislation compliant way of creating and managing NFTs, that directly tackles the *elephant in the room* issue, the pattern whereby many companies become the custodian of the attributes of the NFTs of their users in order to enable asset mutability, or save gas costs. LAOS will actively foster community engagement to raise awareness about existing practices and advocate for migrations towards non-custodial alternatives. It will encourage and support initiatives that promote the adoption of decentralized solutions, empowering users to maintain control over their assets.

With the key feature of bridgeless minting and evolution at its core, LAOS aims at becoming the home of Digital Ownership and Dynamic NFTs, the platform where digital goods are created and evolved for all blockchains.

7.1 Acknowledgments

We thank Gautam Dhameja, Maex Ament, Jordi Baylina, Gerard Colomer and Jordi Herrera for insightful technical discussions; the Web3 Studios team for their work on the tokenomics of LAOS; the team behind Polkadot's Builders Program for technical and strategic support; and the developers in the core Scrum team at Freeverse, Carla Trujillo, Zoran Jaksic, Alex Ripoll, and Luigi di Girolamo for their help in landscaping and analyzing the current state of the art in blockchain protocols.

References

[1] DapperLabs. CryptoKitties deployed smart contract. 2017. Ethereum Address: 0x06012c8cf97bead5deae237070f9587f8e7a266d.

[2] Vitalik Buterin. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform. 2014. https://ethereum.org/en/whitepaper/.

[3] Xin-Jian Jiang and Xiao Fan Liu. Cryptokitties transaction network analysis: The rise and fall of the first blockchain game mania. *Frontiers in Physics*, page 57, 2021.

[4] Kraken. What are CryptoKitties? 2017. https://www.kraken.com/learn/what-are-cryptokitties-nft.

[5] Sunflower Farmers Hits the Polygon Network Where it Hurts: Gas Fees. *PlayToEarn.online*, 2022. https://www.playtoearn.online/2022/01/10/sunflower-farmers-hits-the-polygon-network-where-it-hurts-gas-fees.

[6] Andrew Thurman. Polygon Under Accidental Attack From Swarm of Sunflower Farmers. *Coindesk*, 2022. https://www.coindesk.com/tech/2022/01/06/polygon-under-accidental-attack-from-swarm-of-sunflower-farmers.

[7] Usman W Chohan. Non-fungible tokens: Blockchains, scarcity, and value. *Critical Blockchain Research Initiative (CBRI) Working Papers*, 2021.

[8] Christian Pinto-Gutiérrez, Sandra Gaitán, Diego Jaramillo, and Simón Velasquez. The NFT hype: what draws attention to non-fungible tokens? *Mathematics*, 10(3):335, 2022.

[9] Jacob Evans William Entriken, Dieter Shirley and Nastassia Sachs. ERC-721: Non-Fungible Token Standard. *Ethereum Improvement Proposals*, Jan, 2018. https://eips.ethereum.org/EIPS/eip-721.

[10] Beeple. EVERYDAYS: THE FIRST 5000 DAYS. 2021. Ethereum Address: 0x2a46f2ffd99e19a89476e2f62270e0a35bbf0756, Token: 40913.

[11] Logan Kugler. Non-fungible tokens and the future of art. *Communications of the ACM*, 64(9):19–20, 2021.

[12] Juan Benet. Ipfs-content addressed, versioned, p2p file system. 2014. arXiv:1407.3561.

[13] Sorare. Sorare smart contract on Ethereum. 2020. Ethereum Address: 0x629a673a8242c2ac4b7b8c5d8735fbeac21a6205.

[14] Lionel Messi 2021-22, 321/1000. Entry in OpenSea. https://opensea.io/assets/ethereum/0x629a673a8242c2ac4b7b8c5d8735fbeac21a6205/59955996392973292445454039167917827791697111503455234438533072172692682897997.

[15] Lionel Messi 2021-22, 321/1000. Data in privately-owned servers. https://api.sorare.com/api/v1/cards/5995599639297329244545403916791782779169711150345523443853307217269268289 7997.

[16] Sung-Shine Lee, Alexandr Murashkin, Martin Derka, and Jan Gorzny. SoK: Not Quite Water Under the Bridge: Review of Cross-Chain Bridge Hacks. 2022. arXiv:2210.16209.

[17] Dan Goodin. How $323m in crypto was stolen from a blockchain bridge called wormhole. 2022. https://arstechnica.com/information-technology/2022/02/how-323-million-in-crypto-was-stolen-from-a-blockchain-bridge-called-wormhole.

[18] Ryan Weeks. How a fake job offer took down the world's most popular crypto game. *The Block*, 2022. https://www.theblock.co/post/156038/how-a-fake-job-offer-took-down-the-worlds-most-popular-crypto-game.

[19] Morgan Chittum. Axie Infinity Developers Made Some Trade-offs That Enabled $625M Ronin Breach. *Blockworks*, 2022. https://blockworks.co/news/axie-infinity-developers-made-some-trade-offs-that-enabled-625m-ronin-breach.

[20] Immunefi. Polygon Double-Spend Bugfix Review — $2m Bounty. *Medium*, 2021. https://medium.com/immunefi/polygon-double-spend-bug-fix-postmortem-2m-bounty-5a1db09db7f1.

[21] NBA Top Shot. https://nbatopshot.com.

[22] Paul Paray. Dapper Labs NFT Ruling Dunks on Private Networks. *Coindesk*, 2023. https://www.coindesk.com/consensus-magazine/2023/02/28/dapper-labs-nft-ruling-dunks-on-private-networks.

[23] Moonbeam. Cross-Chain Connected Smart Contract Platform. https://docs.moonbeam.network.

[24] Astar Network. Multichain Smart Contracts. https://docs.astar.network.

[25] Felix Stöger, Anxin Zhou, Huayi Duan, and Adrian Perrig. Demystifying Web3 Centralization: The Case of Off-Chain NFT Hijacking. *Financial Cryptography and Data Security*, 2023.

[26] Gavin Wood. Polkadot: Vision for a Heterogeneous Multi-Chain Framework. 2016. https://polkadot.network/whitepaper/.

[27] Jeff Burdges, Alfonso Cevallos, Peter Czaban, Rob Habermeier, Syed Hosseini, Fabio Lama, Handan Kilinç Alper, Ximin Luo, Fatemeh Shirazi, Alistair Stewart, and Gavin Wood. Overview of Polkadot and its Design Considerations. 2020. arXiv:2005.13456.

[28] Alistair Stewart and Eleftherios Kokoris-Kogia. GRANDPA: a Byzantine Finality Gadget. *CoRR*, abs/2007.01560, 2020. arXiv:2007.01560.

[29] Parity. Parity Bridges Common. High-Level Bridge Documentation. https://github.com/paritytech/parity-bridges-common/blob/master/docs/high-level-overview.md.

[30] Joe Petrowski. Proposal for Common Good Parachains. 2022. https://polkadot.network/blog/proposal-for-common-good-parachains.

[31] Gavin Wood. XCM: The Cross-Consensus Message Format. 2021. https://polkadot.network/blog/xcm-the-cross-consensus-message-format.

[32] Polkadot. XCM v3 - Merged Pull Request. https://github.com/paritytech/polkadot/pull/4097.

[33] Parity Technologies. Polkadot Parachain Host Implementers' Guide. https://paritytech.github.io/polkadot/book/index.html.

[34] Polkadot. Availability & Validity Protocol. https://spec.polkadot.network/chapter-anv.

[35] Polkadot. Collators. https://wiki.polkadot.network/docs/learn-collator.

[36] Mustafa Al-Bassam, Alberto Sonnino, and Vitalik Buterin. Fraud and data availability proofs: Maximising light client security and scaling blockchains with dishonest majorities. 2019. arXiv:1809.09044.

[37] Seun Lanlege. Consensus Proofs. 2023. https://research.polytope.technology/consensus-proofs.

[38] Aidan Musnitsky Vincent Geddes. Snowfork's Analysis of Sync Committee Security. 2023. https://forum.polkadot.network/t/snowforks-analysis-of-sync-committee-security/2712.

[39] Oana Ciobotaru, Fatemeh Shirazi, Alistair Stewart, and Sergey Vasilyev. Accountable Light Client Systems for PoS Blockchains. *Cryptology ePrint Archive*, 2022. https://eprint.iacr.org/2022/1205.