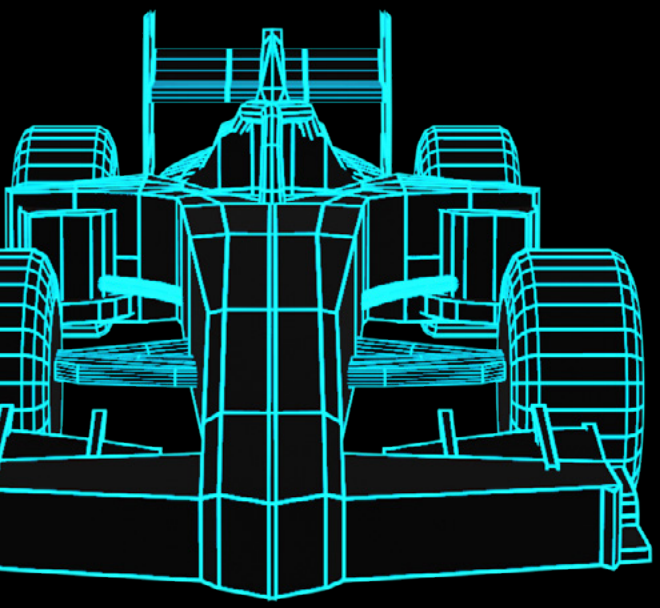
L I T E P A P E R

TITANIUM

NFT2.0
DIGITAL ASSETS & OWNERSHIP

Written By
Arnold Daniels & **Shawn Naderi**



The rise in popularity of NFTs has given a new dimension to the crypto space. It has people thinking about ownership of digital assets.

Typically an NFT is represented by a unique image. However, this image is public, not particularly high quality, and often not remarkable from an artistic point of view. So what gives an NFT its value?

To some extent, it is about exclusivity and cultural relevance. This is especially true for early NFT collections like CryptoPunks. But this doesn't hold up for more recently issued collections.

For those, the value comes from secondary perks that come with the NFT. This can be in the form of in-game items, unlockable content, or access to events.

While NFTs are a step in the right direction concerning decentralized ownership, at the moment their value comes from centralized, off-chain efforts. With NFT2.0, LTO is taking the next step by capturing the value of NFTs in a decentralized way.

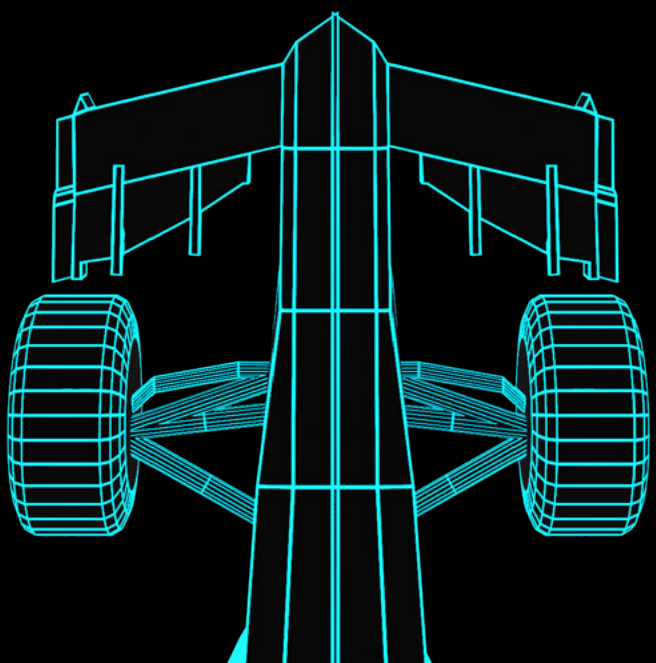
Introducing Ownables

Ownables are the key concept that NFT2.0 is built around. They are digital assets that live in your wallet.

To understand the power of Ownables, we first need to look at how NFTs and blockchain wallets work.

A blockchain wallet doesn't hold NFTs. It only holds key pairs which are used for cryptographic signatures. Each key pair is associated with a blockchain address. For an NFT, an on-chain record is kept connecting a unique number with the owner's address. Transferring or interacting with the NFT requires the owner to sign with its private key.

Ownables flip this concept around. Everything concerning the NFT is stored in your wallet. Not only images or media but also the smart contract and event chain. We have also made it possible to embed whole applications, opening up a world of endless possibilities.



What makes LTO Network unique is that it's a hybrid blockchain with a public and private layer.

The private layer allows holding info privately, delegating consensus to the public layer. Ownables live on the private layer.

LTO Private Layer

The private layer isn't a blockchain. It's a communication layer where events can be shared peer-to-peer. Events on the private layer have a similar purpose as transactions on the blockchain. The key difference is how the events are organized.

Event chain

For a blockchain, there's a single chain consisting of blocks that hold transactions related to many different smart contracts. On the private layer, each smart contract has its own event chain. Clients share individual smart contracts, with the related events, peer-to-peer.

Each event is both hashed and signed. The hash is added to the next event, creating a hash chain. A blockchain isn't required to validate the chain. However, it would be trivial to roll back and rewrite history. To ensure that event chains are immutable, consensus is delegated to the LTO public chain.

Consensus

A blockchain consensus mechanism prevents rolling back the chain, which could be abused for double-spending. Mechanisms like Proof of Work or Proof of Stake rely on the chain being distributed across multiple nodes. These can't be used on the private layer, since an event chain may only exist in a single wallet.

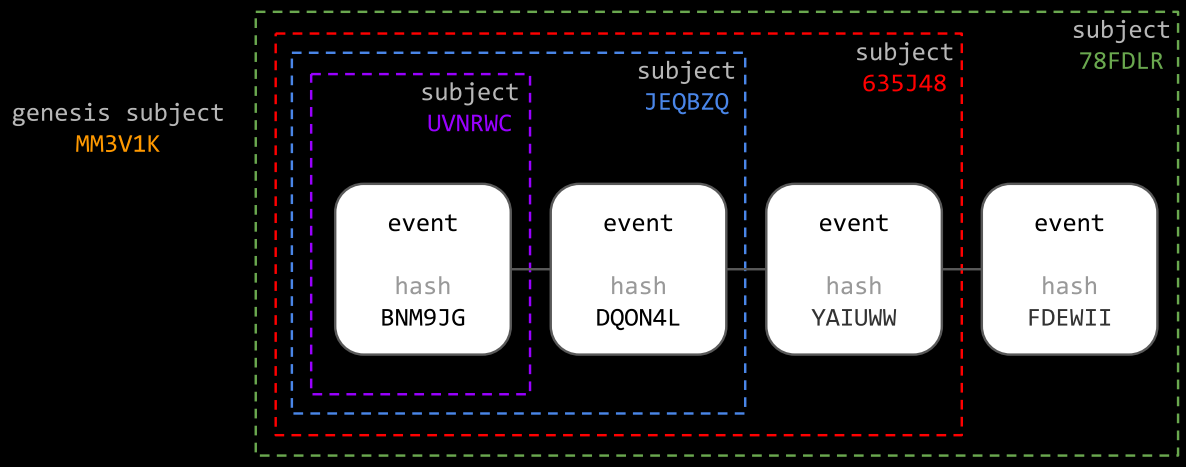
To solve this, the private layer writes each event hash to the LTO public chain. With the TITANIUM upgrade, we generate two values instead of one. In addition to the event hash, a subject is generated.

The subject is a unique value that reflects the event chain to the point where the new event is added.

Instead of just submitting the hash, a combination of the hash and subject is written to the public chain. The subject should be used only once. It's used to verify that the event chain is complete and not tampered with.

LTO Private Layer

Private layer



Public layer

✓	MM3V1K BNM9JG	UVNRWC DQON4L	JEQBZQ YAIUWW	635J48 FDEWII	78FDLR [not found]		
✗	MM3V1K BNM9JG	UVNRWC DQON4L	JEQBZQ YAIUWW	635J48 FDEWII	78FDLR WDZZMF		Incomplete event chain
✗	MM3V1K BNM9JG	UVNRWC DQON4L	JEQBZQ YAIUWW	635J48 9G3QXP	635J48 FDEWII	78FDLR [not found]	Roll-back detected

The Private Layer Consensus Method

Private Smart Contracts

With the TITANIUM upgrade, we're adding smart contracts to the private layer.

Until now, events needed to be interpreted by an external application, like our workflow engine.

This allows more complex processes to occur on the private layer, the result of which can then be reflected and stored on the public layer.

Since there are no gas fees and no costs for the size of the contracts, smart contracts on the private layer may be larger and more complex than typical blockchain contracts.

NFT2.0 Smart Contract Details

- These smart contracts aren't run on a node but are executed directly in your wallet.
- Smart contracts are compiled to WebAssembly.
- They run in a sandboxed and deterministic runtime environment.

Any WebAssembly program can function as a smart contract with only a few constraints. The main constraint is that the contract must be deterministic. That means that the state of the contract can only be changed through events that have been added to the event chain. Information about ownership is part of the state of the contract.

Ownables

With the TITANIUM upgrade, we're adding smart contracts to the private layer.

Ownables can be used in our new NFT2.0 Wallet and can be traded peer-to-peer or in a marketplace using the LTO bridge.

Forging

Forging a new Ownable starts with the genesis event. This event must contain the smart contract. All subsequent events are processed by this contract. Information such as the public key that signed the event is passed to the contract. This information can be used to set properties like the issuer, which is automatically the initial owner.

Additional constructor arguments can be passed as part of the genesis transaction to set the initial parameters. This includes content like images, audio, and video.

Basic precompiled smart contracts are made available by LTO Network. They can be used directly from the wallet to forge a new Ownable. This doesn't require any programming skills. For custom or advanced features, users are required to write their own smart contract.

Transferring

Changing ownership requires two steps:

1) The new owner is declared through an event and added to the chain. Anchoring of the event on the public chain prevents double-spending.
2) The event chain, including the smart contract and all assets, are packaged up and transferred to another wallet.

Ownable packages can be transferred to another wallet in different ways:

- They can be sent over the private layer to another wallet.
- A bridge can be used as an intermediary, which holds a copy of the package.
- If users meet in person, the Ownable can be transferred between devices.

Content

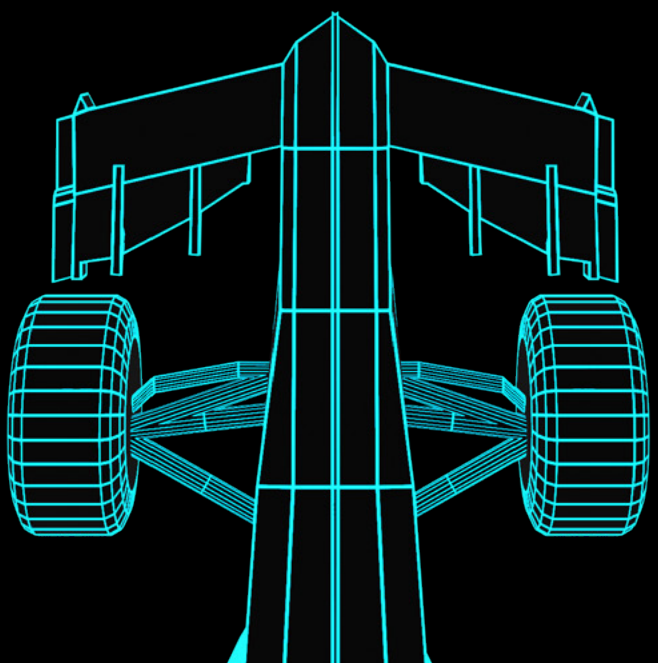
Ownables can contain public information, as well as content that's only available for the owner. However, it's a misconception to view an Ownable as a static package of files.

Events can contain binary data, which is processed by the smart contract to be stored as images, documents, videos, or audio files. The files are accessible by the contract through the WebAssembly System Interface. Content can be added and changed as the user interacts with the Ownable.

Since the data doesn't need to be stored on all nodes of a public blockchain, there are far fewer limitations concerning file size. An Ownable may contain the original master tracks of a song, 3D Models or the source files of a virtual world.

Unlockable content

NFTs typically only have public information that's available on-chain. Corresponding images are served from IPFS or a centralized web server.



To make NFTs more interesting, issuers are providing unlockable content that's only available for the owner. For public blockchains like Ethereum, it's impossible to manage that in a decentralized way. Unlockable content is completely centralized through a platform like OpenSea.

Ownables live on the LTO private layer. That means unlockable content is the default. All content is private and must explicitly be made public.

Content

Dynamic content

The smart contract has access to all files it has created. It can serve this to generate a UI, and also modify it based on new events.

When the Ownable is used in a game, the game can issue verifiable credentials for accomplishments like completing a level or collecting a number of stars. These credentials can be wrapped into an event and added to the event chain.

The smart contract will act as the verifier of the credential and use the information to level up the Ownable. It can modify an image within the Ownable to visualize leveling up.

Consumables

Another option to dynamically modify the content of an Ownable is to have it consume another ownable.

For instance, one Ownable represents a playable character in the metaverse. Another Ownable represents a specific skin. You can trade the skin as an NFT or store it as an Ownable. However, to make it available in-game it needs to be consumed.

To do this, the owner will first add an event to the Consumable, with a reference to the character Ownable. By doing so, the Consumable can no longer be used, transferred or traded.

The smart contract will read and process the content of the Consumable, making it part of the character Ownable.

Complete freedom

Developers have complete freedom over the content of the Ownable. The only constraint is that it needs to be deterministic. While we've given some examples of how this can be used, we believe that developers will come up with concepts that go beyond our imagination.

Cross-chain bridge

Each Ownable creates its own chain with information. Similar to smart contracts on the blockchain, the smart contract of the Ownable can use information that has not been put on-chain. Cross-chain logic is required to enable features like trading and staking.

Staking

Holding an Ownable might give you perks like rewards tokens. Since the Ownable only exists in your wallet, there's no way to automatically distribute rewards. Instead, it works similar to most yield farms.

By keeping track of the rewards that have been claimed, the smart contract can calculate the unclaimed reward. These tokens can be claimed by providing proof of ownership.

Trading

Ownables only exist in your wallet and not on a public blockchain. To trade them on marketplaces like OpenSea and Rarible they must be swapped for regular NFTs using a bridge.

To swap an Ownable for an NFT:

- The owner adds an event to the chain for bridging the Ownable.
- The owner sends the packaged Ownable to the bridge.
- The bridge mints or unlocks an NFT on any blockchain, like ETH, BSC, or SOL.
- The bridge adds an event to the chain, embedding the solidity event as proof.
- The event is sent back to the wallet of the owner, which can now start trading the NFT.

To revert the NFT to an Ownable:

- The new owner requests a copy of the Ownable from the bridge and signs the request with their private key.
- The bridge verifies that the key matches the holder of the NFT and sends a copy of the packaged Ownable.
- The owner locks or burns the NFT on the external blockchain.
- The owner adds an event to the chain, embedding the solidity event as proof.

Cross-chain bridge

Verifiable events

For bridging, the smart contract requires the owner or bridge to add proof to an event. This proof must be permanent and immutable.

Using stateful information from a solidity contract doesn't work, but a solidity event can be used as proof. On the LTO Network public chain, proof could be in the form of an association or claim transaction.

Using verifiable events prevents having to rely on the bridge as a trusted party.

Legal procedure

Additional logic can be required to swap the NFT for the Ownable, like a procedure to legally assign ownership. This opens the door for tokenizing real-world assets.

Proving you're the holder of an NFT can't be used to manage ownership when it comes to real-world items like real estate or copyright.

Bridging the tokenized asset allows you to trade it as an NFT. The new owner can take possession of the underlying asset by swapping the NFT to an Ownable, going through the proper legal procedure.

Minting as NFT

Instead of having the bridge mint an NFT, it's possible to first issue an NFT which can later be swapped for an Ownable. This has several advantages:

An NFT minting event can take place as normal.

Instead of burning an NFT to swap it for an Ownable, the NFT is locked. This means that the owner is known on-chain.

Staking rewards can be done purely based on the NFT.

The correct issuer and collection are displayed on marketplaces like OpenSea.

The disadvantage of this approach is that the user can't freely choose which blockchain to mint on and which bridge to use. Regardless, we expect that this method will be popular as it streamlines the use of Ownables for existing platforms.

NFT2.0 Wallet

With the TITANIUM upgrade, we'll also release a new wallet for running and managing Ownables.

Unlike typical blockchain wallets, the LTO Network NFT2.0 Wallet holds the complete event chain, smart contract, and all contents.

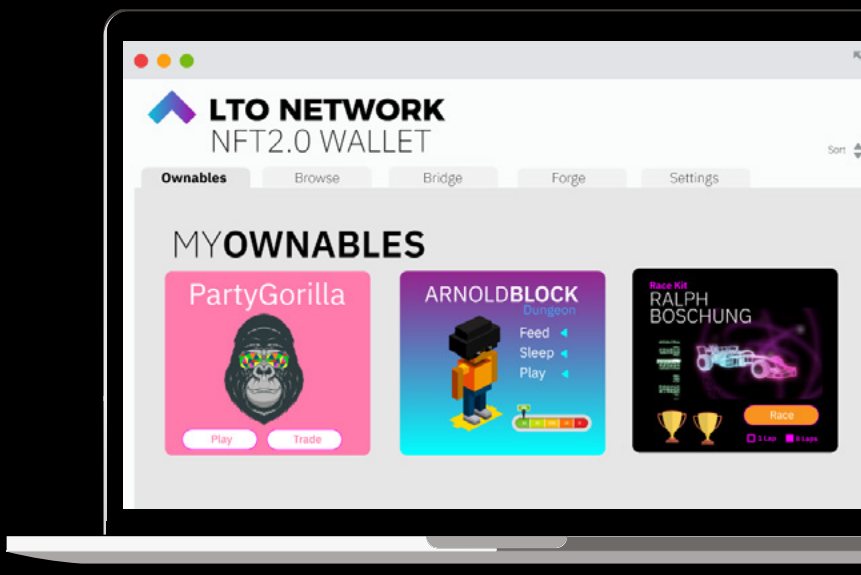
Executing a smart contract doesn't happen on a node, it happens within the wallet.

Widgets

By default, the wallet will display the name and thumbnail for an Ownable. However, it is possible for the smart contract to define a widget that is displayed instead. This widget can show information about the Ownable, such as unclaimed awards, and can display custom buttons, and animations.

For the widget, the smart contract needs to render a UI of 400 by 400 pixels. Everything within this block is sandboxed. Neither the smart contract nor the widget is able to access information about other ownables or the wallet.

To protect privacy, it's impossible for a widget to send HTTP requests to communicate with an external API.



NFT2.0 Wallet

Embedded applications

Standard functionalities like exporting, deleting, and bridging an Ownable are built into the NFT2.0 Wallet and are always available. For additional functionality, the Ownable can contain an embedded application.

Applications are run in a browser-based environment, thus can be created using HTML, CSS, and JavaScript. The application can interact with the smart contract and access content like images and videos. It doesn't have the same constraints as the smart contract. It doesn't need to be deterministic and can communicate with external APIs.

These applications can be as simple as a viewer for the content of the ownable, or as complex as a complete game.

Interaction with NFTs is done through applications that are centrally hosted and managed. There's no guarantee that these will exist forever. Embedded applications are part of the Ownable and thus under the owner's control.

Using the bridge

The NFT2.0 Wallet will show a list of NFTs you own that can be swapped for Ownables. If the NFT is traded on a marketplace, it will show the current price as well as other relevant information.

Swapping Ownables for NFTs (and vice versa) using the bridge, doesn't require any technical knowledge and can be done with a click of a button.

The NFT2.0 Wallet will support publishing directly to OpenSea, Rarible, or Binance, making it easier to list an Ownable-based NFT.

Ownables Marketplace

Ownable projects and existing Ownables that are for sale can be showcased to other users directly in the NFT2.0 Wallet.

The wallet will link to a marketplace or website, where the Ownable can be purchased.

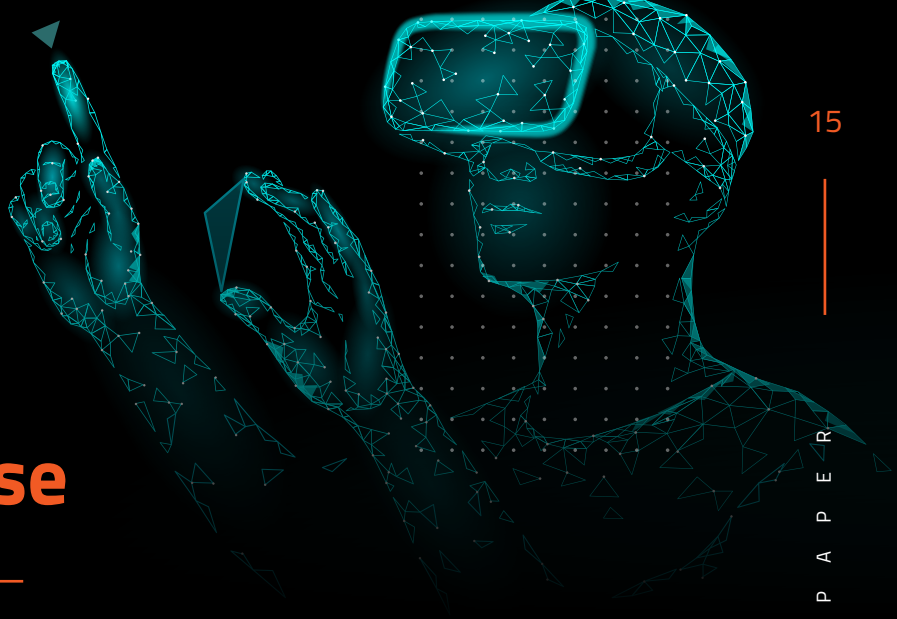
Ownables can be displayed free of charge. However, the order in which projects are shown is decided in a decentralized way and requires burning LTO tokens.

When performing a burn transaction, the project ID is given as an attachment. Each LTO token burned this way is initially worth one point. The value decreases over 10,000 blocks (one week) in a linear fashion.

$$\text{points} = \text{burned LTO} \times \left(1 - \frac{\text{current block - tx block}}{10{,}000}\right)$$

A project owner can move a project up in the list by burning more LTO tokens. Points of different burn transactions will be added when determining the display order.

Wallets can subscribe to a blacklist of inappropriate or scam projects. The blacklist will be managed by the LTO Network team in collaboration with the LTO community.



Open Metaverse

With NFT2.0, LTO Network is contributing to a permissionless open metaverse, where anyone can create a virtual world by running a server.

Currently, metaverses depend on managed repositories, which require a permissioned model.

Holders of a regular NFTs never truly possesses the virtual item.

All of the assets live in the repository.

When entering a virtual world, the repository then checks if the user has the right to wear or use the item.

The downside is that if the asset is ever removed from the repository, e.g. through a DAO vote, the NFT is rendered useless.

Ownables are different. The holder truly possesses the item, as the Ownable contains all files and information required to render

Open Metaverse

Verification

It's up to the maintainer of each world to decide whether or not to accept an asset introduced by a user. Of course, they should always verify, through a cryptographic challenge, that the user is the current owner.

However, this doesn't prevent plagiarism and unwanted assets from entering the metaverse. Gatekeeping is required. In a permissionless open metaverse, it's up to the maintainer of a world to determine which Ownables are accepted and which will be rejected.

To do so, it can use a web of trust. This is a decentralized trust model that doesn't rely on a hierarchy. Instead, trust is obtained through endorsements of peers.

Additionally, parties may create curated lists. These parties can be added to the trust network and help determine which Ownables should be accepted.

Cross-world assets

Ultimately, the goal is for a person to buy or create an item in one world and take it with them to the next. Items can simply be taken out of the world and stored as an Ownable in a wallet until the owner is ready to place it back in the metaverse.

In some cases, the item will be on the avatar. But in other cases, like virtual real estate, the item will be part of the world. In that case, it makes sense that there's only one copy across all worlds. This can be accomplished through the smart contract, or by using an association to lock the asset.

Beyond Titanium

With the Cobalt and TITANIUM upgrades, LTO Network is taking giant strides in providing solutions for a decentralized future.

After TITANIUM, we'll focus on privacy-aware decentralized applications. This enables peer-to-peer communication between embedded applications, allowing for multi-player games, eliminating the need for a central server. This will also allow services like the bridge to be fully decentralized.

We have great plans for the future of LTO Network and the TITANIUM upgrade brings an immense amount of depth that allows new types of content and applications to be built. We cannot wait to see what you do with it!

TITANIUM