Dynamic Non Fungible Tokens (NFTs)

If you’ve been even loosely following web3 for the past couple of years, you’ve probably heard of the Bored Ape Yacht Club—NFTs sold for hundreds of ETH (the largest Bored Ape Yacht Club transaction was made for a whopping [$3.4 million](https://moneymade.io/learn/article/most-expensive-bored-ape#:~:text=BAYC%20NFT%20%238817%20has%20broken,spinner%20hat%2C%20and%20party%20horn.)).



*Bored Ape Yacht Club #8817*

These JPEG images of pixelated characters are probably what comes to mind when you hear the once viral word “NFT.” However, the concept of a non-fungible token is much more diverse, and potentially much more powerful.

In this article we’ll look at the concept of *dynamic* NFTs—NFTs that can change their on-chain data in response to an on-chain (or even real-world) event. We’ll look at some current examples, some potential future uses, and how StarkNet enables dynamic NFTs with its cheap computation and storage costs.

## What are Dynamic NFTs

NFT stands for “non-fungible token,” a digital asset representing ownership of some unique item/information. Creators can tokenize their works by representing them with NFTs; tickets to events can be represented as NFTs; real-world assets such as real estate property can be bound to such tokens and become tradeable, even music and movies can be captured as NFTs.

But NFTs can still be considered “first-generation”—use cases that can be imagined for non-fungibles have yet to be exhaustively explored and evolutions of the standard are being created.

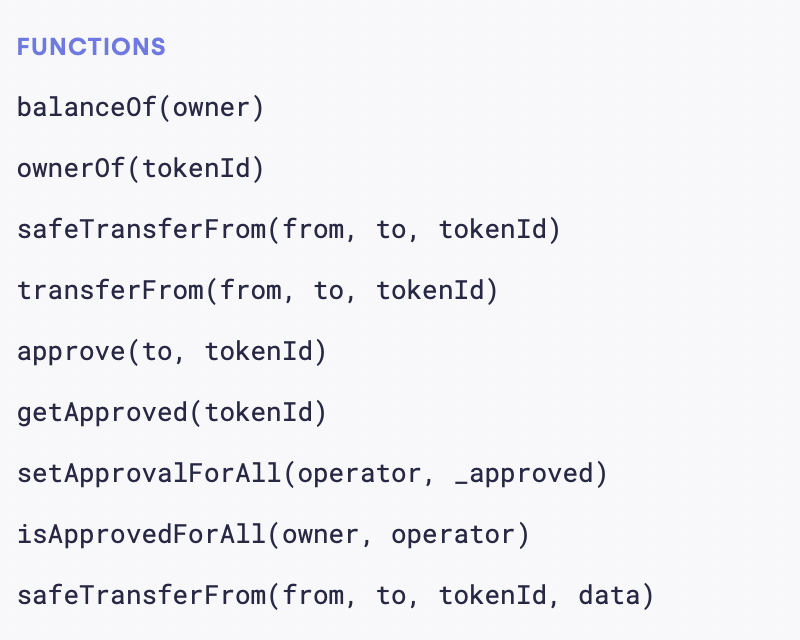
One example of the future of NFTs is **Dynamic NFTs—**NFTs which can change the value they represent. For example, NFTs that represent an image can have the image change colors and shape or even attain a completely new appearance. Audio NFTs can evolve their audio in reaction to on-chain events.

Since all the data related to such NFTs is stored on-chain, the amendments are transparent and can be permissioned as per requirements. We will get into the details of how such a project could work, but first, let’s look at NFTs from a technical perspective.

Unfortunately, most NFTs today can’t be dynamic. Let’s look at why.

## The Current State of NFTs

Usually, when we talk of NFTs, we speak of the [ERC721](https://eips.ethereum.org/EIPS/eip-721) standard, which was formally created in 2018. The ERC721 standard is a base interface for non-fungible tokens that declares a few of the functions that every smart contract implementing ERC721 must support (for someone with no Solidity background, it is kind of a blueprint that defines how to interact with a contract/class, specifying functions but not their implementations).



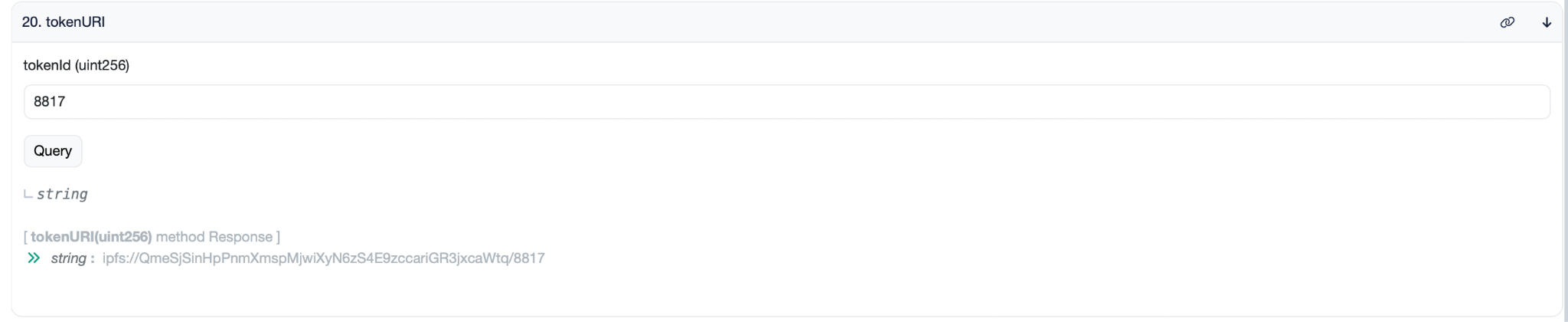
Set of functions implemented by ERC721

Whenever an NFT changes hands, the `transferFrom` function is called, which changes the owner of the NFT in the ledger. Similarly, when we check who owns a particular token, we call the `ownerOf` function.

Marketplaces like OpenSea rely on NFTs supporting this standard to display tokens on their websites and transfer NFTs between buyers and sellers.

However, look deeper at one of the BAYC NFTs: [Ape #8817](https://opensea.io/assets/ethereum/0xbc4ca0eda7647a8ab7c2061c2e118a18a936f13d/8817). If you visit the page on OpenSea, you will see the image associated with the 8817 Ape. However, where is this image stored? **It’s not on-chain**! Storing the whole image on the Ethereum mainnet would cost a small fortune in storage space, and considering that there are millions of NFTs on Ethereum have been deployed, it would not have been possible to store the images on-chain.

What these smart contracts do is store all of their NFT-related data (i.e., “metadata”) on some URL that is accessible from the Internet. The two main options for NFTs are keeping the metadata off-chain but on a decentralized storage solution such as IPFS ([here](https://ipfs.io/ipfs/QmeSjSinHpPnmXmspMjwiXyN6zS4E9zccariGR3jxcaWtq/8817) is an example of the metadata related to Ape #8817), or storing the data on a centralized source like traditional cloud storage.



You may check the metadata for any BAYC token from its [smart contract](https://etherscan.io/address/0xbc4ca0eda7647a8ab7c2061c2e118a18a936f13d#readContract) on mainnet

The first choice—IPFS—creates a fundamental constraint: the data becomes static. Since data, once stored on IPFS, cannot be modified, making any amendments to IPFS is impossible.

The second solution, however, allows modification of NFT-related metadata, but introduces the series problem of **centralization**, which goes against the ethos of blockchain.

Hence, in this constant conflict of increased accessibility vs increased centralization, we end up with NFTs that stay “the same” even for use cases that need dynamic changes, such as gaming, SBTs, creator tokens, etc.

NFTs that get the benefits of both of these choices—dynamic *and* decentralized—are needed. This is where dynamic NFTs come into play.

## Dynamic NFTs Made Possible by Cheap Computation and Storage on Starkware

Starknet is a validity rollup (commonly known as a zero-knowledge rollup) developed by [StarkWare](https://starkware.co/starknet/) that uses cryptographic systems called STARKs to massively decrease computation and storage cost. Because of these savings, Starknet finally makes the concept of dynamic NFTs—dynamic and decentralized—a possibility.

Let’s look at how it works with an example.

The [Carbon Protocol](https://twitter.com/Carbonable_io) is a project that is a hub for carbon neutrality, from funding regeneration projects to managing companies' net zero programs and reporting. Their strong commitment to decentralization and transparency led them to developing one of the first [proofs of concept](https://mintsquare.io/asset/starknet-testnet/0x056e77566cf9cff229a0fcf74e18b57c84bdcb23d099ad62a1a2a7918b1db6e0/3) for "dynamic" NFT such that the metadata for the NFT is stored completely on-chain. Below is an example token of the NFT contract that Carbonable deployed.



We can see the “value” stored in the token in the screenshot above as “square km” of area related to the token (this value has no real-life meaning yet since this is really more of a PoC). As an experiment, I transferred a major chunk of this “value” to another address and automatically the metadata related to the token updated. The image describing my token also got changed, and as we can see below the color of the text changed from golden to bronze. Note that this change has happened completely on-chain. This is the power of dynamic NFT !

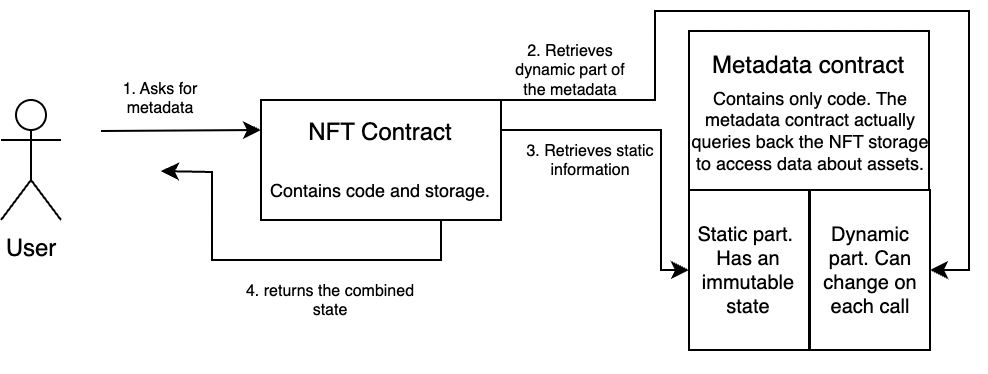


Unlike ERC721's, where the balance and ID of a specific token are used to understand the ownership of an NFT, for dynamic NFTs, the main information is the **value stored in a token**.

What this means is that whenever a user requests the value stored in the token, the value can change. This is implemented by declaring a simple contract (let's call it the "metadata" contract) that handles both the dynamic and the static part of the NFT metadata. For a more superficial understanding, we can think of the "static" part as a short string that stores some property of the NFT.

On the other hand, the "dynamic" part is different because to return some information (for simplicity, let's assume it also returns a string), it can run some code on the metadata contract. The code can be dynamic in the sense that the values it returns can change with different chain-specific parameters or the state of the contract itself. It would be possible to change the state of the contract upon external interaction.

Note that the metadata contract only has to be declared on Starknet; there is no need even to deploy it as such. Unlike Ethereum, Starknet distinguishes between a contract class and a contract instance. A contract class represents a contract's code (but with no state), while a contract instance represents a specific class instance with its own state. When you declare a contract class, the code is stored on-chain, but there is no state to upkeep. Since the metadata contract does not necessarily have to have a state, simply declaring it is enough to run the code inside the functions which are called by the NFT contract.



The proof of concept deployed by the Carbonable team is completely decentralized since the SVG stored by the image has all the code deployed on the contract (see code [here](https://github.com/Carbonable/carbon-protocol/blob/a9cd9efb11bda033f3f0a340e8ff5006793a4247/src/metadata/utils.cairo#L71)). This allows anyone to keep track of the changes made to the SVG. As soon as the token is transferred from one user to another, the token's value can change, showing its ability to adapt.

## Use Cases for Dynamic NFTs

This example deployed for testing purposes only opens the door to a vast array of possibilities, such as:

**Images and Animations.** As the example above shows, amendable images can be stored on-chain. However, similar to that, animations can also be stored on-chain and can be modified with user interaction. Several teams are working on this right now, and this possibility can be used by the cryptogaming sphere massively. For example, in the proof of concept project by Carbonable (that we mentioned earlier) the color of the text changes with the value stored in the token. So, if a user, say, has a token with value > 1000, the color of the text is golden and silver if the value is in the range of 100-1000.

**Music and audio**. When we listen to music or audio files, these files are often in a .mp3 or .wav format. This format of an audio file is "finite" in that there is no quick way to change a beat or only a small part of the clip easily. Audio formats (like WAV or MP3) store actual sound data, while MIDI (Musical Instrument Digital Interface) files contain instructions for synthesizers or other devices to generate sounds.

[Raphael Doukhan](https://www.linkedin.com/in/raphaeldoukhan/), the co-founder of Rosis Lab, worked on a proof of concept for generative music by storing a MIDI file as a JSON object on Starknet and allowed the file to be modified by external interaction. Some parts of the music file were constant (e.g., music structure, melody theme, etc.), and others were amendable (e.g., BPM, tension profile, keys, etc.). This proof of concept was a prime example of generative music where an artist's work "[is not a finite object anymore but just lives beyond the creator and beyond the creation](https://twitter.com/Starknet/status/1658019475084517378)." For example, the beat frequency of the audio file can be changed as users interact with the file and hence the music would attain a unique characteristic upon new interactions.

**Gaming, fantasy sports and collectibles**. On-chain games could harness the power of the dynamicity of NFTs to make in-game changes to the environment, objects, and characters based on different gameplay.

For example, [Realms](https://twitter.com/LootRealms), a multiplayer on-chain game on Starknet also is exploring the use case of dynamic NFTs to have on-chain character skins (and even music) that changes with the game. In fantasy sports, character features could be upgraded/downgraded based on the player's real-life performance (which could be retrieved from oracles). An example of this is Lamelo Ball's ChainLink-powered NFT. After each game, the NBA player's stats are brought on-chain using an oracle, and the fans can collect the player's "talent" as tokens. There are 4 different categories of Lamelo’s NFTs: Red Mars, Blue Neptune, Silver Moon and Gold Sun. They are all tied to different statistics of the NBA player. For instance, Red Mars is tied to the points scored and Silver Moon is tied to assists made. Hence, each time LaMelo scores points or makes an assist the value inside the respective NFT increases.



Lamelo Ball’s NFT

This idea can be incorporated in games so that skins of the characters become “cooler” as the player’s level increases, or the player’s achievements are shown on the NFT as medals and trophies.

**Generative Art**. Art comes in many forms, some of which have been discussed above. However, the notion of art is wide open. [OG:CR](https://ogcrystals.com/) (Organic Growth: Crystal Reef) is a dynamic NFT project that bridges digital and physical art. Users can collect NFTs of crystals on-chain, and the NFTs evolve with every transfer between the users. The user's wallet's history (with a hint of randomness) decides the changes made to the crystals. The end idea is to convert the digital crystals into a physical art piece that can be displayed in museums worldwide.

## Dynamic NFTs are an Evolution

NFTs, though still early in their development, offer a myriad of applications. The emergence of Dynamic NFTs represents a further evolution in the uptake of this technology. Storing a considerable amount of information efficiently and modifying it consistently is made possible by chains with low-cost computation like Starknet. Starknet is witnessing significant ongoing development, and now is an opportune moment to join the developers of the future and consider building projects with exciting new use cases on Starknet.