Tech Explainers

CRYPTO TOKEN CLASSIFICATION

I. EXECUTIVE SUMMARY

A societal "crypto mythos" has formed around crypto tokens. These little widgets are seemingly adaptable to any scenario ("tokenize X"), yet few in the legal profession understand what they are or how they work. This technology explainer demystifies crypto tokens for legal practitioners. Specifically, we'll explain how crypto tokens are created using smart contracts and the popular standards they follow based on their fungibility. We'll also describe how they're ascribed functionality, either intrinsically on the smart contract or extrinsically in how they interact with other systems. With a full understanding of what a crypto token is—and is not—legal practitioners will be able to understand how law and regulation fit into their clients' token projects.

II. DEFINING THE TOKEN: A SMART CONTRACT

Crypto tokens are utilized pervasively across various industries, from video games to fan engagement with artists to decentralized finance's governance models. Tokens will continue to be a fundamental asset for emerging industries, including the metaverse and proof of humanity projects. However, while individuals may purchase or utilize crypto tokens, many do not understand what tokens are, how they function, how they are built and their key features. This guide will break down these essential features of tokens, including how they are afforded functionality and assigned fungibility.[1]

Crypto tokens are built using smart contracts which run on top of a blockchain. Each token is a unit of value that represents an asset or utility for its token holder. Tokens can provide a variety of functionalities, such as a medium of currency, access to a digital right, ownership in a unique asset and engagement in voting or governance.

When working with clients regarding their crypto tokens, it is essential for legal practitioners to understand the token's fungibility and functionality, as both contribute to the regulation and classification of the token.

A. Fungibility: The Token's Standard

DEFINING TOKEN FUNGIBILITY

Developers commonly start by assigning fungibility to a token, not because it maintains elevated importance, but rather because, practically speaking, fungibility has a set of defined, common standards that are useful for blockchain networks. The fungibility of a token refers to its interchangeability with

1 *See, e.g.*, David J. Kappos, *et al.*, *Fuzzy Tokens: Thinking Carefully About Technical Classification Versus Legal Classification of Cryptoassets,* 36 Berkeley Tech. L.J. (2023).

other tokens of the same project. The question of whether a given token is fungible is answered on a spectrum, ranging from a completely fungible token, which is indistinguishable from another token of the same project, to a non-fungible token (NFT), which is unique, indivisible and often represents an underlying asset which is itself non-fungible. Semi-fungible tokens fall in the middle of this spectrum and maintain qualities of both fungible and non-fungible tokens.

CHOOSING TOKEN FUNGIBILITY

Tokens are commonly created on an EVM-compatible blockchain (Ethereum Virtual Machine blockchain) through the use of a smart contract.[2] A "smart contract" is simply some code,[3] and any smart contract starts with an entirely blank canvas; it can express any arbitrary program the developer chooses. This flexibility creates a problem, though: if each developer is programming tokens in a different way, major issues with interoperability would inevitably arise.[4] To solve this, the blockchain community has coalesced around a set of common standards for the code, thereby constituting smart contracts for tokens on the Ethereum blockchain.

There are three core standards[5] for a token developer to choose from: ERC-20, ERC-721 and ERC-1155, and the only difference between them is the extent of their fungibility.

THE MAJOR TOKEN STANDARDS	
ERC-20	Fungible token
ERC-721	Non-fungible token
ERC-1155	Semi-fungible token

FIGURE 1: A chart depicting the three most popular token standards.

ERC-20 is a fully fungible and transferrable token standard. ERC-721 is a fully non-fungible token standard. ERC-1155 is a semi-fungible token standard. Open-source software platforms, like OpenZeppelin,[6] provide one of the most popular implementations of these standards. After selecting a popular version of a token standard on OpenZeppelin, developers can plug and play.[7]

2 We present three standards based on the Ethereum blockchain (ERC) because Ethereum is the most widely used blockchain for developing tokens.

3 Yet, a smart contract is most certainly not a contract! It is unfortunate that developers used that term, and it became ubiquitous before lawyers got involved. The term had no relation to offers, acceptance, meeting of the minds, etc., when it was coined.

4 Crypto tokens operate on blockchain networks, which rely upon a common set of standards to ensure seamless transactions and compatibility on all platforms. Yet, if each token was coded differently without a set of standards, it would lead to interoperability issues. Thus, common standards are necessary for the smooth processing of all transactions.

5 This paper does not cover every token standard or standards on blockchains other than Ethereum that are not EVM-compatible. However, by focusing on the dominant standards and Ethereum blockchain, we highlight the necessary information for a solid grasp of crypto tokens.

6 OpenZeppelin is an online, open-source platform which allows developers to build token smart contracts and select token standards and further functionalities via the OpenZeppelin Contracts Wizard.

7 *Contracts Wizard*, OPENZEPPELIN, https://docs.openzeppelin.com/contracts/4.x/wizard (last visited June 15, 2023).

TOKEN STANDARDS REPRESENTED ON A LOOKUP TABLE

All the bells and whistles we associate with token functionality, discussed in detail below, are outside these three token standards; in selecting a token standard, a developer is only selecting where on the spectrum of fungibility one's token should fall. Put another way, these three token standards focus solely on how the token is indexed on a "lookup table."[8]

For fungible tokens using the ERC-20 token standard, since all such tokens of a particular project are fully interchangeable with each other, the applicable lookup table tracks the quantity of such tokens held at any particular wallet address. The lookup table is visualized below. Wallet addresses are signified starting with 0x.

FUNGIBLE TOKEN (ERC-20)	
0xSasha	20
0xDan	1
0xDave	50
0xCarys	20

FIGURE 2: Lookup table for ERC-20.

For non-fungible tokens, the ERC-721 lookup table tracks a unique token index number, which serves to identify the token. As visualized below, each token index number differs from the next. The table also tracks the wallet address that holds this unique token.

NON-FUNGIBLE TOKEN (NFT) (ERC-721)	
1	**0xCarys**
2	**0xDan**
3	**0xSasha**
4	**0xDan**

FIGURE 3: Lookup table for ERC-721.

8 We use the term "lookup table" as an abstraction to signify a programming object that functions as such.

Semi-fungible tokens exhibit the characteristics of both non-fungible and fungible tokens through hybrid fungibility. Consider a video game, in which there are three items offered to its players.[9] The first item is called a "bronze coin," and there are one million copies of the bronze coin that are all interchangeable with each other. Thus, the bronze coin represents one series. The second item is called a "silver coin," and there are 100 copies that are interchangeable with each other. The silver coin represents a second series. The last item is called a "platinum coin," and there is only one of them, which represents its own series. Within each series, all bronze coins are interchangeable with each other, as are all silver coins. However, in comparing each series, they are all completely different. The silver coin is not interchangeable with the bronze coin, meaning that each series is non-fungible on a series level. Thus, for semi-fungible tokens, there are two layers to the ERC-1155 table. The first layer tracks the unique token index number, and the second layer tracks the quantity of tokens held at a unique wallet address. In this sense, ERC-1155 is an amalgamation of the ERC-20 and ERC-721 standards.

SEMI-FUNGIBLE TOKEN (ERC-1155)		
1	**0xCarys**	1
	0xDan	50
2	**0xDan**	2
	0xDave	50

FIGURE 4: Lookup table for ERC-1155.

B. Token Functionality

Once one of the three token standards is selected, the level of fungibility is set, but the token does not yet have any other features or functionalities. To reiterate, the default implementation of these standards has no other utility ascribed to it. Features and functionalities can be added to the token in one of two ways: intrinsically, through adding to the code of the token smart contract, or extrinsically, often through arrangements that are external to that token smart contract, and that are sometimes even external to the relevant blockchain itself.

9 Items in video games are a quintessential example of the use of semi-fungible tokens. The gaming space, particularly the purchase of unique in-game items, is well-suited for semi-fungible tokens.

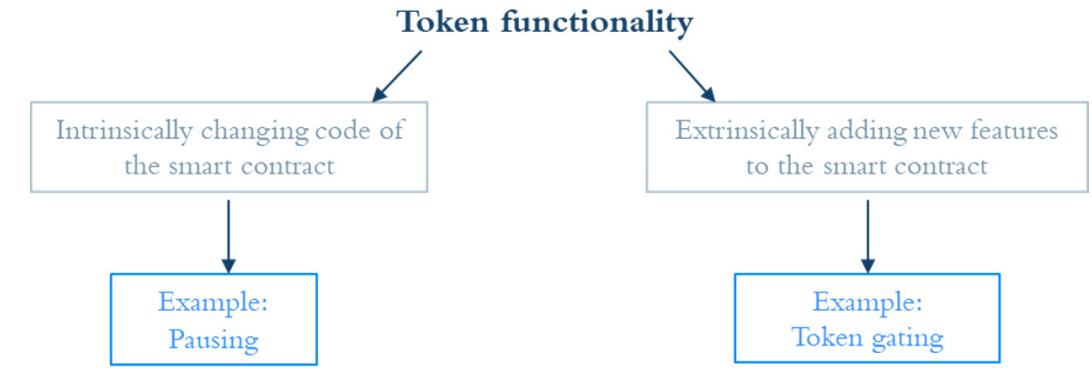

FIGURE 5: Establishing token functionality.

III. INTRINSICALLY ASCRIBING FUNCTIONALITY

To add functionality to tokens intrinsically means to add more code directly to the token standard code, so that the complete internal code of the token smart contract not only implements the chosen fungibility level, but also implements other features that the tokens are meant to have. Open-source software platforms, like OpenZeppelin, provide access to a multitude of ways for developers to easily add code to the three token standards and thereby implement a variety of other features and functionalities. Developers can add on features via code which would include burning tokens, minting new tokens and allowing voting. Developers can also add a mechanism which grants them the power to pause the minting of new tokens. For those functionalities outside of the standard that are ubiquitous, OpenZeppelin also offers pre-written code to easily add such functionalities. Tokens do not have to simply maintain one functionality; developers can add on multiple functionalities, depending on what is ideal. A few of the common intrinsic functionalities are further described, as follows.

MINTING AND BURNING

Minting is a functionality which allows for generating new tokens. While all projects have to start with a token supply, this functionality can be added so that the total supply of tokens can increase.

Burning is the antithesis of minting. Burning a token allows a developer to provide a functionality to remove or "burn" tokens from circulation. The burned tokens are transferred to a wallet which can no longer be accessed by anyone and, thus, are effectively destroyed. By burning, the overall supply of the tokens of a certain project is reduced. This feature often leads to increasing the value and appeal of the remaining tokens, due to further scarcity. Both the functionalities of minting and burning are frequently used for "tokenomics" mechanisms.

VOTING

Developers can build a token to allow for voting by certain token holders. Often in the context of a decentralized autonomous organization (DAO), governance token holders can exercise their voting rights on vital decisions, assist in governing a token project or contribute to proposed changes. The token holder's voting influence and the issues raised for a vote will differ depending on the project.

PAUSING

Another popular functionality added on is pausing. The pausing mechanism prevents a token holder from taking specific actions when the pause "switch" is "flipped" on. For example, pausing can be utilized to prevent a token holder from transferring a token. This functionality can be extremely valuable, especially in emergency situations. For legal practitioners, understanding pausing is essential to discerning what safety valves exist in the context of glitches or emergency situations and further analyzing the ramifications and proper next steps.[10]

Developers intrinsically change the code to allow for the pausing mechanism. Specifically, developers add a new code variable which tracks whether the switch is flipped "on" or "off." Before taking actions that utilize this functionality, the token smart contract will check the switch code variable's current setting and only run if pausing is indeed off.

IV. EXTRINSICALLY ASCRIBING FUNCTIONALITY

Tokens can also be granted functionality extrinsically that is provided to the token holder following obtaining ownership. Under this method, the internal code of the token smart contract need not be amended to provide access to these functionalities; instead, it is added externally to the token smart contract, and access to additional utility, experiences or content is provided once ownership of a given token is verified.

TOKEN GATING

Token gating is a clear example of extrinsic functionality. Token gating is the process of only granting access to exclusive spaces, events and communities for those who own a particular token. The access is not granted via additional intrinsically coded elements (features beyond those that simply implement the standard).

For example, the Flyfish Club is a private dining club which requires individuals to own a Flyfish Club NFT. After purchasing the NFT, the token holder is granted access to an exclusive restaurant and social experiences in New York City, offered by the Club. Membership is afforded by the Flyfish Club NFT.[11] The original mint for the Flyfish Club NFT sold out in 2022.[12]

Of course, private dining clubs in New York City are not assets that live in the token smart contract or are on-chain; rather, the Flyfish Club experience is facilitated extrinsically through the dining experience. This project utilizes one of the core benefits of blockchain technology, as a token contract lookup table, and can only be updated by the token holder.

---

10 Emergency situations could include a glitch in a mint process, which would require developers of a certain project to pause minting new tokens and fix the glitch. Another common example of pausing is to halt bad actors in video games, which protects other players from their negative actions. *See* David J. Kappos, D. Scott Bennett, Michael E. Mariani, Sasha Rosenthal-Larrea, Daniel M. Barabander and Callum A.F. Sproule, *NFTs, Incentives and Control: Technical Mechanisms and Intellectual Property Rights*, 6 Stanford Blockchain Law & Policy L.J. (2023).

11 Access to certain areas of the Flyfish restaurant and ability to book reservations is truly "gated" based on whether a token holder has a Flyfish token or Flyfish Omakase token. The Flyfish token provides access to a restaurant experience. The Omakase token provides further access to exclusive dining experiences and areas.

12 *About Flyfish Club*, FLYFISH CLUB, https://www.flyfishclub.com/about (last visited June 13, 2023).

V. WHY LEGAL PRACTITIONERS SHOULD UNDERSTAND TOKEN CLASSIFICATION

It is vital for practitioners to understand how tokens are classified based on fungibility and how they are granted functionality, whether intrinsically or extrinsically. Thus, when assessing the legal status of a crypto token, practitioners should begin with the following questions to their clients:

- Is the token fungible?
- What standard, if any, does the token implement?
- What is the token's functionality?
- How is the token afforded this functionality?[13]

Through these assessments, legal practitioners will have a solid foundation to apply the relevant legal analysis. As regulation comes down the pipeline to address crypto tokens, many have fixated on the word "token" without understanding what it truly represents – a smart contract that can have additional features. This guide serves to provide a clear, comprehensive breakdown for regulators and legal practitioners to understand how tokens in a particular crypto project manifest their features and functionalities. Sometimes a token project is created to pause bad actors or allow token holders to vote on key issues or open the gate to in-person experiences like an exclusive restaurant. However, it is important for regulators to understand tokens beyond the societal "crypto mythos" which has formed around their usage. Put simply, tokens are a vehicle to provide a functionality. To summarize, we have provided a helpful flow chart which walks through the key questions and considerations when assessing a token of a particular crypto project.

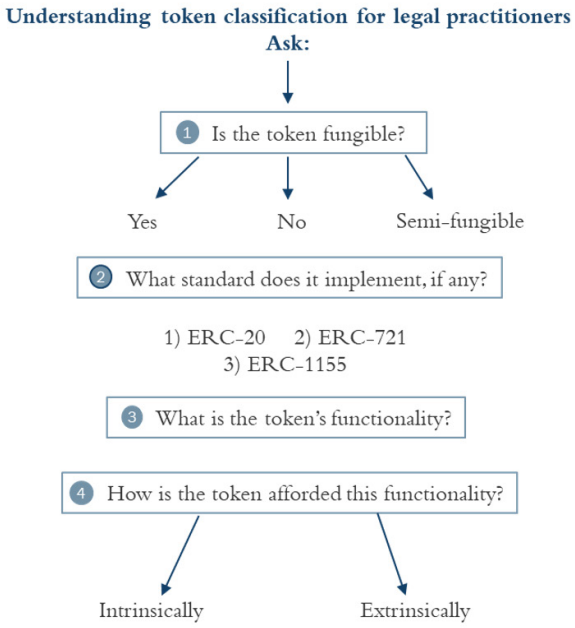

FIGURE 6: Questions for legal practitioners to consider.

13 See David J. Kappos, D. Scott Bennett, Michael E. Mariani, Sasha Rosenthal-Larrea, Daniel M. Barabander and Callum A.F. Sproule, *NFTs, Incentives and Control: Technical Mechanisms and Intellectual Property Rights*, 6 Stanford Blockchain Law & Policy L.J. (2023).

David J. Kappos
+1-212-474-1168
dkappos@cravath.com

Sasha Rosenthal-Larrea
+1-212-474-1967
srosenthal-larrea@cravath.com

Carys J. Webb, *CIPP/US*, *CIPP/E*
+1-212-474-1249
cwebb@cravath.com

Daniel M. Barabander
+1-212-474-1284
dbarabander@cravath.com

Cravath, Swaine & Moore LLP

NEW YORK

Worldwide Plaza
825 Eighth Avenue
New York, NY 10019-7475
T+1-212-474-1000
F+1-212-474-3700

LONDON

CityPoint
One Ropemaker Street
London EC2Y 9HR
T+44-20-7453-1000
F+44-20-7860-1150

WASHINGTON, D.C.

1601 K Street NW
Washington, D.C. 20006-1682
T+1-202-869-7700
F+1-202-869-7600