



Ethereum Tokens

What are ERC standards?

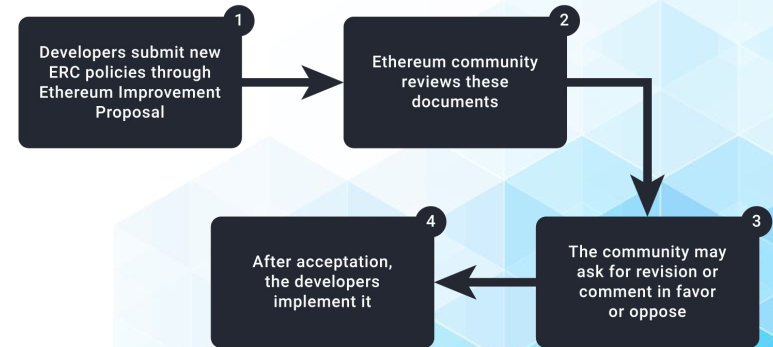
A document called a "Ethereum Request for Comments" (ERC) is created by smart contract programmers who use the Ethereum blockchain platform. They lay out the principles that Ethereum-based tokens must follow in these document.

The Ethereum community reviews these papers using a method known as the "Ethereum Improvement Proposal (EIP)."

After going through the EIP (Ethereum Improvement Proposal) process, the Ethereum community approves some of these reports, finalizes them, and developers adopt them.

An Ethereum Improvement Proposal (EIP) is a design document that informs the Ethereum community or describes a new function for Ethereum, its systems, or its ecosystem.

CREATION PROCESS



Different ERC Standards

<ul style="list-style-type: none">■ Most popular token standard.■ Used in most of the ICOs.■ Fungible token standard.■ Allows the implementation of standard API within a smart contract.	ERC 20	<ul style="list-style-type: none">■ A standard for Non-fungible tokens.■ Wallets and exchanges can reuse tokens.■ Token holders can transfer token while also approving a 3rd party to spend it.■ Not in use, still in EIP phase.	ERC 721
<ul style="list-style-type: none">■ A standard for a method, instead of tokens.■ Covers how interfaces are identified.■ States how any contract can publish the interfaces after the implementation.■ States how to detect when a contract implements ERC-165■ Covers the way to detect when a smart contract uses any given interface.	ERC 165	<ul style="list-style-type: none">■ Reduces friction in crypto transactions.■ Not in use, still in EIP phase.■ Gets rid of the double transaction verification of ERC 20.■ Lowers transaction overhead.■ Allow users to reject incoming tokens from a blacklisted address.	ERC 777
<ul style="list-style-type: none">■ Prevents accidental burns of tokens, a bug in ERC 20.■ Developers can either accept or decline tokens arriving at their smart contract addresses.■ Rejected transactions will fail but won't burn the tokens.■ Not in use, still in EIP phase.	ERC 223	<ul style="list-style-type: none">■ An extension of ERC 20.■ Wallets and exchanges can reuse tokens.■ Token holders can transfer token while also approving a 3rd party to spend it.■ Not in use, still in EIP phase.	ERC 827
<ul style="list-style-type: none">■ An extension in ERC 20 standard■ Uses two functions: 'increaseSupply', and 'decreaseSupply'.■ Can increase or decrease the token supply.■ Not in use, still in EIP phase.	ERC 621	<ul style="list-style-type: none">■ Allows companies to use blockchain to maintain share registries.■ Identity verification and mandatory whitelisting of token holders.■ Only whole value of tokens, i.e., no partial value.■ Recording of information regulators mandate.■ Not in use, still in EIP phase.	ERC 884

What are ERC-20 standards?

ERC-20 tokens are a collection of similar tokens with the same set of properties. They adhere to the ERC-20 standard, which establishes a set of guidelines for the development and management of fungible tokens.

The use of ERC-20 tokens allows for the development of micro-economies with liquid markets for a variety of applications.

Fabian Vogelsteller proposed the ERC-20 (Ethereum Request for Comments 20) Token Standard in November 2015. It is a Token Standard that implements an API for tokens within Smart Contracts.

It includes features such as the ability to move tokens from one account to another, as well as the ability to see an account's current token balance and the overall availability of the token available on the network. It also has several other features, such as approving the use of a certain amount of token from one account by a third-party account.

Trading ERC-20 Tokens

ERC-20 tokens may be exchanged on a variety of websites, including:

- Regular exchanges
- Decentralized exchanges
- Automated liquidity pool, which can facilitate smooth token swapping by utilizing Automated Market Making algorithms to decide the price of purchasing ERC-20 tokens.

ERC-721 Standard

Each ERC-721 token is associated with its own collection of properties and values.

Being one of a kind increases the value of ownership, particularly in the case of highly sought after tokens.

The ERC-721 protocol is a standard that must be implemented for any smart contract that produces ERC-721 tokens.

NFTs can be interacted with using a variety of features, including:

- Identifying the ERC-721 token's owner address
- Approval of an ERC-721 token transfer. Each address to which we want to transfer must be accepted before the transfer can begin.
- Checking an ERC-721 token's accepted addresses.
- ERC-721 Token Transfer

Trading ERC-721 Tokens

ERC-721 tokens can be exchanged for other tokens or Ether in the same way that ERC-20 tokens can.

Automated Market Making algorithms are not possible to apply since all ERC-721 tokens have unique properties.

At the moment, the agreed method of trading them is to auction or exchange them on NFT-specific peer-to-peer marketplaces, the main of which is OpenSea.

This generates new economies for digital collectibles, as well as remarkable figures for a new form of asset:

- The top three types of NFTs have a trading rate of over 1,000 Ether in a single day.
- Another VR simulation, Decentraland, has a total volume of 44,000 Ether divided among 3,662 owners. This equates to a 12 Ether average possession of LAND estate.
- An F1 gaming object worth 415.9 Ether was the most expensive NFT offered in 2019.

ERC-1155 Tokens

The ERC-1155 token is a type of regular token that can store tokens that can behave as ERC-20 or ERC-721 tokens, or both at the same time, under its authority.

Witek Radomski, Andrew Cooke, Philippe Castonguay, James Therien, Eric Binet, and Ronan Sandford created this standard.

New functions and possibilities of the ERC-1155 token are:

- Mass transfers as standard
- Multiple tokens in the same contract
- Integrated token type detection
- Secure token transfer

ERC-721 Vs ERC-1155: Comparison Table

Basis of Comparison	ERC-721	ERC-1155
Token Creation	Only one token in a single contract	Multiple tokens in a single contract
Limitation	Can only create NFTs	Can create both Fungible and Non Fungible Contracts
Cost	Expensive	Cheaper
Ease of Use	Allows single operation for each transaction	Allows multiple operations in a single transaction
Storage Requirement	More	Less
Efficiency	Low	High