INTRO TO TOKENS

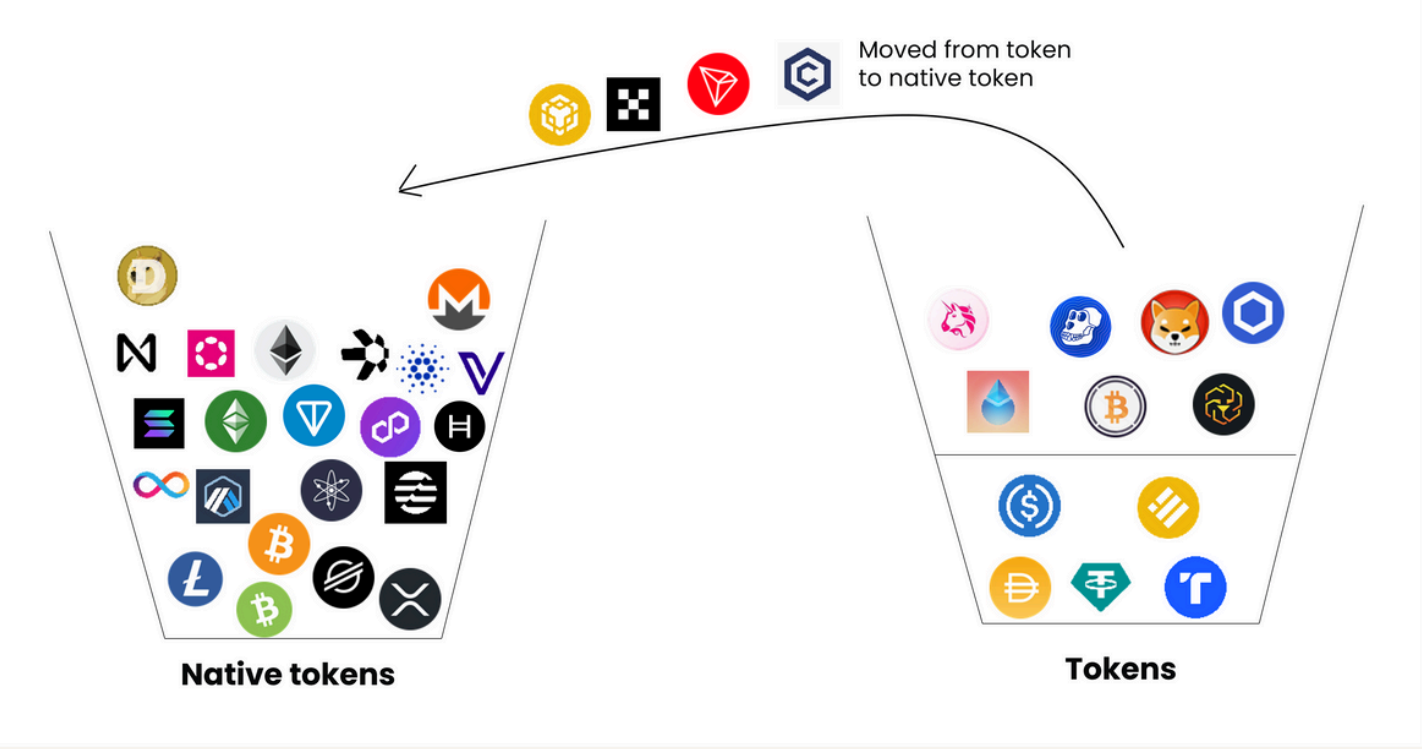
Created: Feb 2025, by David Liu
Last Edited: Feb 2025, by David Liu

-David Liu, Co-Founder and CTO of Uniblock.

TOPICS

1. Native Tokens
2. ERC20
3. ERC721
4. ERC1155

NATIVE TOKENS



EIP

Ethereum Improvement Proposals

All Core Networking Interface ERC Meta Informational

EIP Types

EIPs are separated into a number of types, and each has its own list of EIPs.

Standards Track (861)

Describes any change that affects most or all Ethereum implementations, such as a change to the network protocol, a change in block or transaction validity rules, proposed application standards/conventions, or any change or addition that affects the interoperability of applications using Ethereum. Furthermore Standard EIPs can be broken down into the following categories.

Core (301)

Improvements requiring a consensus fork (e.g. EIP-5, EIP-211), as well as changes that are not necessarily consensus critical but may be relevant to "core dev" discussions (for example, the PoA algorithm for testnets described in EIP-225).

Networking (21)

Includes improvements around devp2p (EIP-8) and Light Ethereum Subprotocol, as well as proposed improvements to network protocol specifications of whisper and swarm.

Interface (52)

Includes improvements around client API/RPC specifications and standards, and also certain language-level standards like method names (EIP-6) and contract ABIs. The label "interface" aligns with the interfaces repo and discussion should primarily occur in that repository before an EIP is submitted to the EIPs repository.

ERC (487)

Application-level standards and conventions, including contract standards such as token standards (EIP-20), name registries (EIP-137), URI schemes (EIP-681), library/package formats (EIP-190), and account abstraction (EIP-4337).

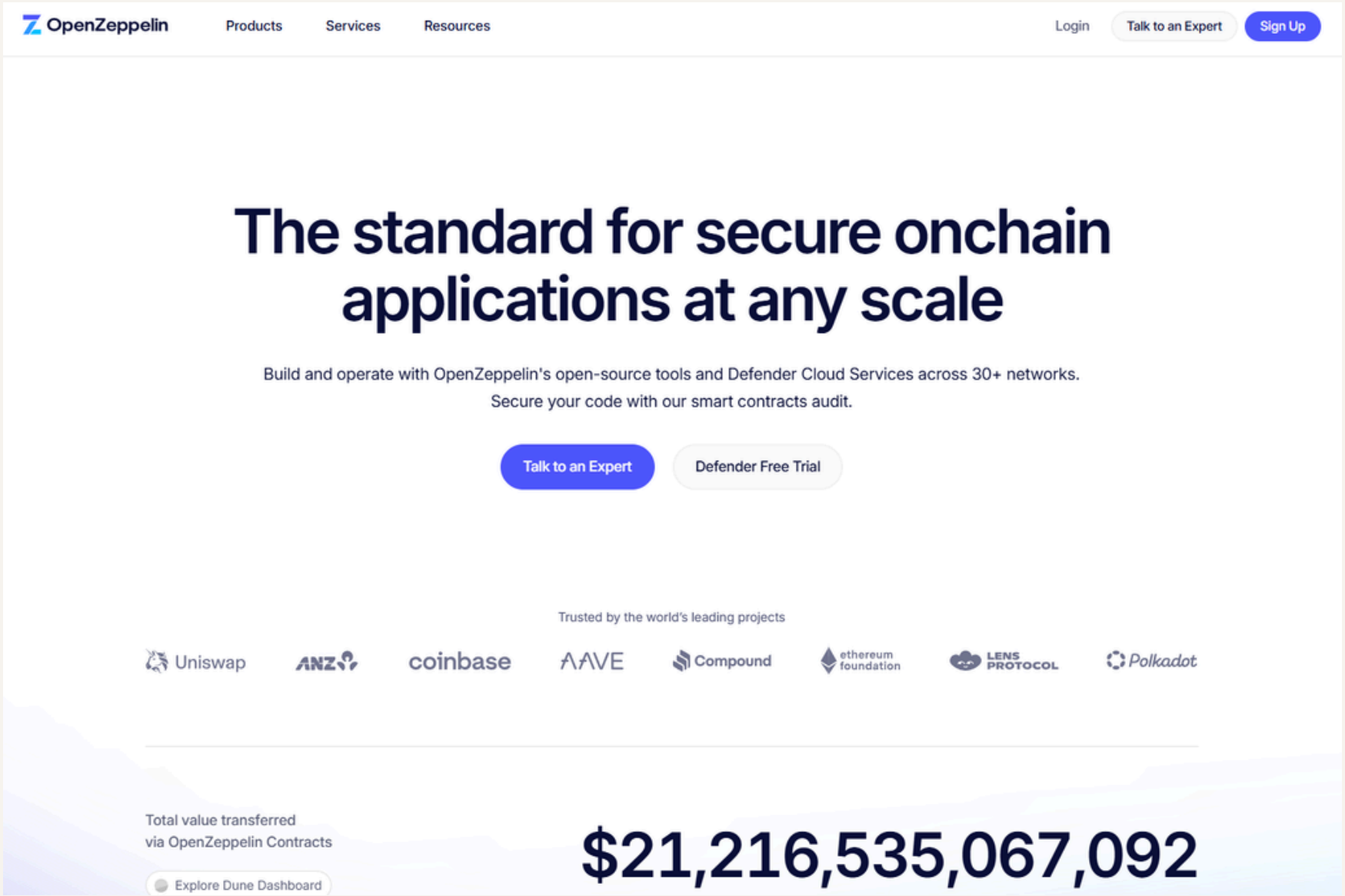
Meta (35)

Describes a process surrounding Ethereum or proposes a change to (or an event in) a process. Process EIPs are like Standards Track EIPs but apply to areas other than the Ethereum protocol itself. They may propose an implementation, but not to Ethereum's codebase; they often require community consensus; unlike Informational EIPs, they are more than recommendations, and users are typically not free to ignore them. Examples include procedures, guidelines, changes to the decision-making process, and changes to the tools or environment used in Ethereum development. Any meta-EIP is also considered a Process EIP.

Informational (10)

Describes a Ethereum design issue, or provides general guidelines or information to the Ethereum community, but does not propose a new feature. Informational EIPs do not necessarily represent Ethereum community consensus or a recommendation, so users and implementers are free to ignore Informational EIPs or follow their advice.

REUSE CONTRACTS



Ethereum Token Standards

ERC - 20

Fungible Tokens

Most basic token standard, used to create interchangeable tokens

Trade-able virtual currencies
Governance/voting tokens
Staking tokens


WBTC


USDC


CRV

ERC - 721

Non-Fungible Tokens

Basic NFT standard, used to create unique tokens, distinguishable from others in the same collection

Collectable art
Digital items and property
Tickets (events, seats, lottery)


CryptoPunks


WMVG


Audioglyph

ERC - 1155

Multi-Token Standard

A single interface that manages any combination of multiple token types (fungible, non-fungible, etc).

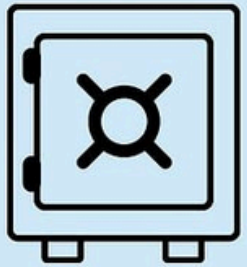
Alternate to ERC-20 and ERC-721
Video game items
Memorabilia


Enjin Tokens


Beanstalk Fertilizer


Adidas Metaverse

ERC - 4626



Tokenized Vault Standard

A standard that represents a yield-bearing vault; extending ERC-20 to include deposit, redeem, etc

Lending markets
Interest bearing tokens
Aggregators


Yearn v3 Vaults


Umami USDC Vault


EnreachDAO Yaggr (BNB)

IERC20

```
import "@openzeppelin/contracts/token/ERC20/IERC20.sol";
```

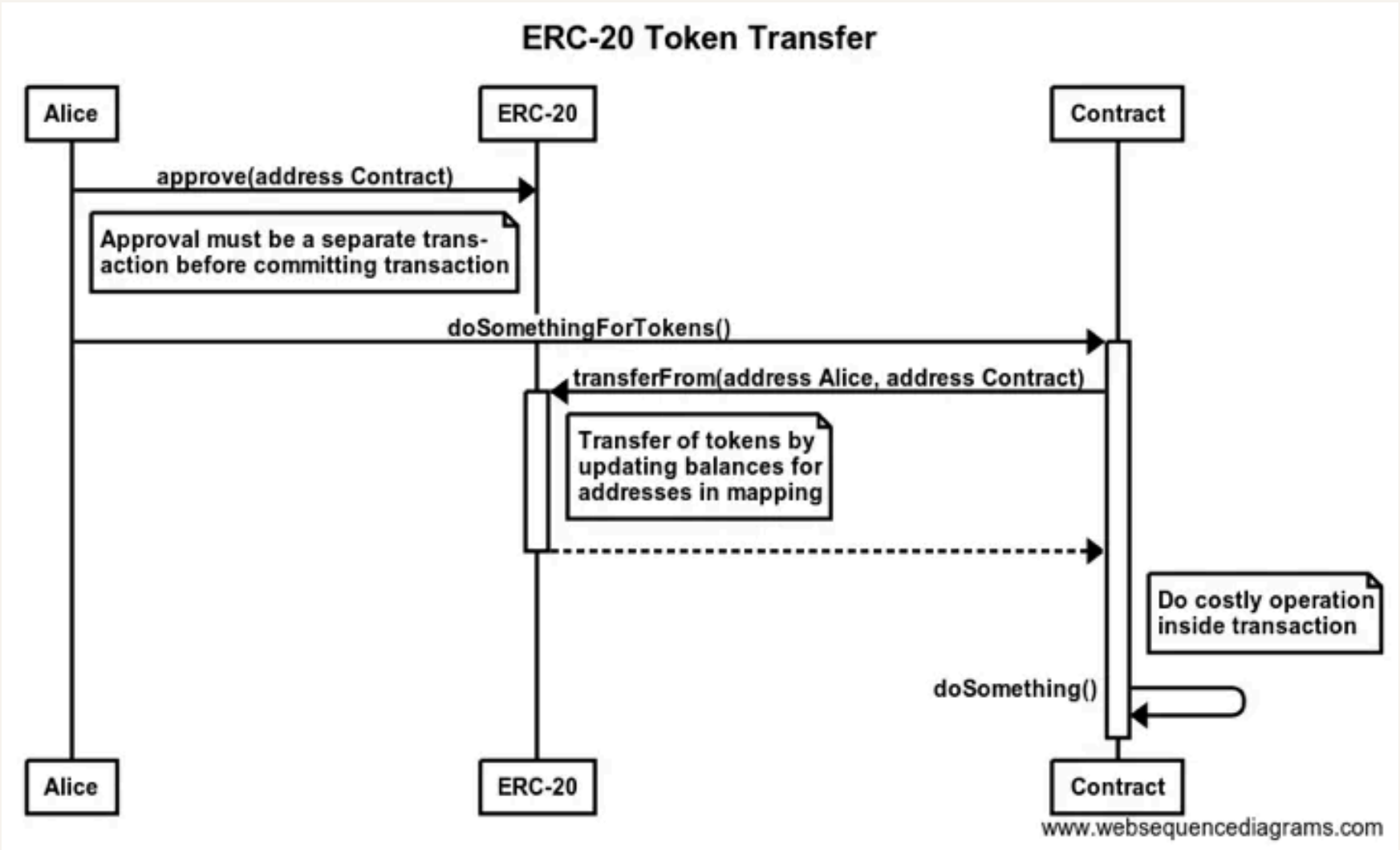
Interface of the ERC-20 standard as defined in the ERC.

FUNCTIONS

```
totalSupply()

balanceOf(account)

transfer(to, value)

allowance(owner, spender)

approve(spender, value)

transferFrom(from, to, value)
```

EVENTS

```
Transfer(from, to, value)

Approval(owner, spender, value)
```


ERC-20 Token Transfer
Alice
ERC-20
Contract
approve(address Contract)
Approval must be a separate trans-
action before committing transaction
doSomethingForTokens()
transferFrom(address Alice, address Contract)
Transfer of tokens by
updating balances for
addresses in mapping
Do costly operation
inside transaction
doSomething()
Alice
ERC-20
Contract
www.websequencediagrams.com

IERC721

```
import "@openzeppelin/contracts/token/ERC721/IERC721.sol";
```

Required interface of an ERC-721 compliant contract.

FUNCTIONS

```
balanceOf(owner)
ownerOf(tokenId)
safeTransferFrom(from, to, tokenId, data)
safeTransferFrom(from, to, tokenId)
transferFrom(from, to, tokenId)
approve(to, tokenId)
setApprovalForAll(operator, approved)
getApproved(tokenId)
isApprovedForAll(owner, operator)
```

```
supportsInterface(interfaceId)                    IERC165
```

EVENTS

```
Transfer(from, to, tokenId)
Approval(owner, approved, tokenId)
ApprovalForAll(owner, operator, approved)
```

IERC1155

```
import "@openzeppelin/contracts/token/ERC1155/IERC1155.sol";
```

Required interface of an ERC-1155 compliant contract, as defined in the ERC.

FUNCTIONS

```
balanceOf(account, id)
balanceOfBatch(accounts, ids)
setApprovalForAll(operator, approved)
isApprovedForAll(account, operator)
safeTransferFrom(from, to, id, value, data)
safeBatchTransferFrom(from, to, ids, values, data)
```

```
supportsInterface(interfaceId)
```

IERC165

EVENTS

```
TransferSingle(operator, from, to, id, value)
TransferBatch(operator, from, to, ids, values)
ApprovalForAll(account, operator, approved)
URI(value, id)
```

OpenSea Drops Stats Create Search / Login

Buy now Make offer

Price History

Listings

Offers

Price	USD Price	Quantity	Floor Difference	Expiration	From
14.21 WBERA	$109.56	1	16% below	in 7 days	hungrydogg
14.2 WBERA	$109.48	3	16% below	in 30 days	zhurekk
13 WBERA	$100.23	1	23% below	in 23 hours	gamekid
10 WBERA	$77.10	1	41% below	in 7 days	0xrdinarylife.eth

Description

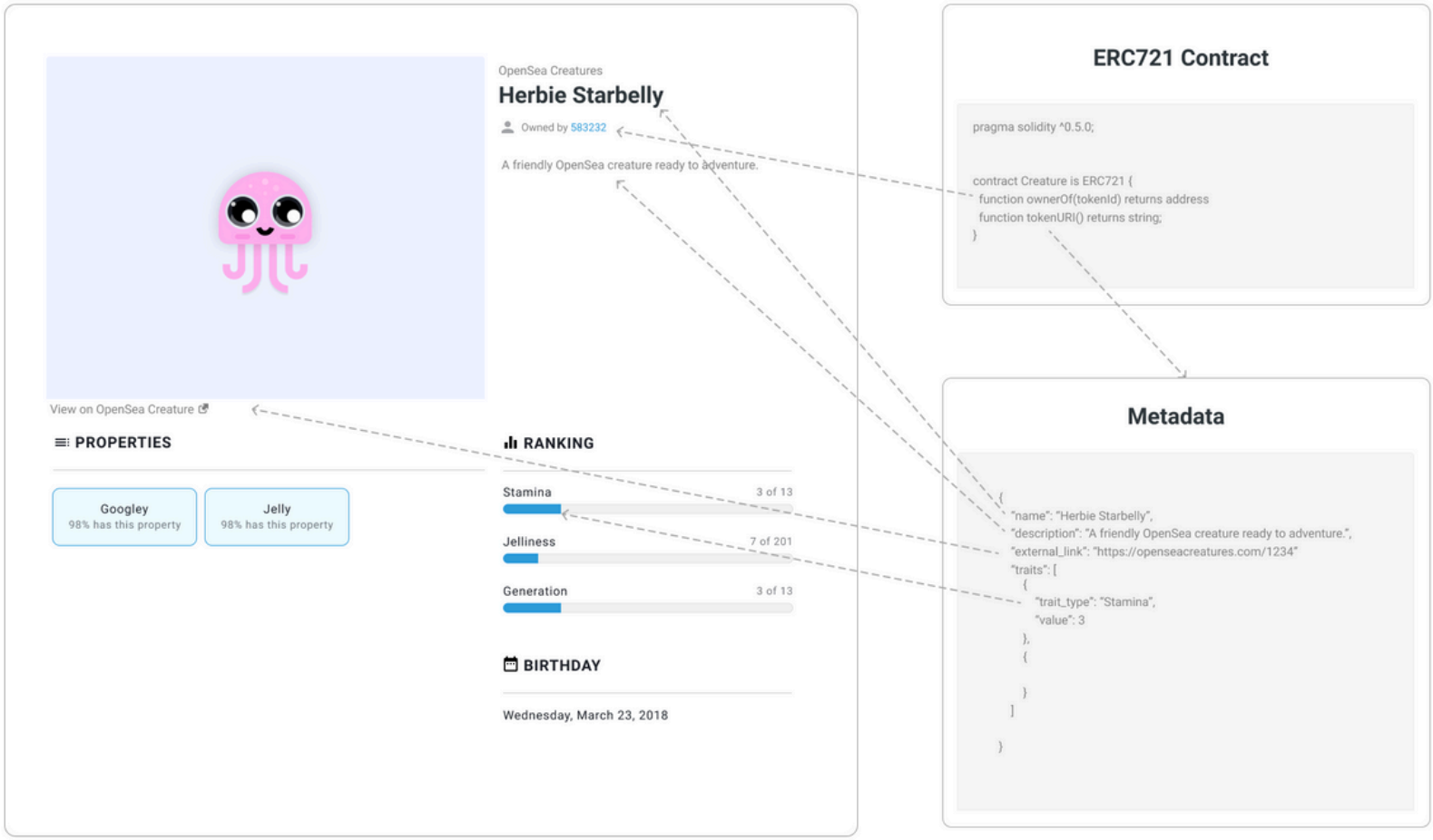
By **89695E**
Beras can't read.

Traits

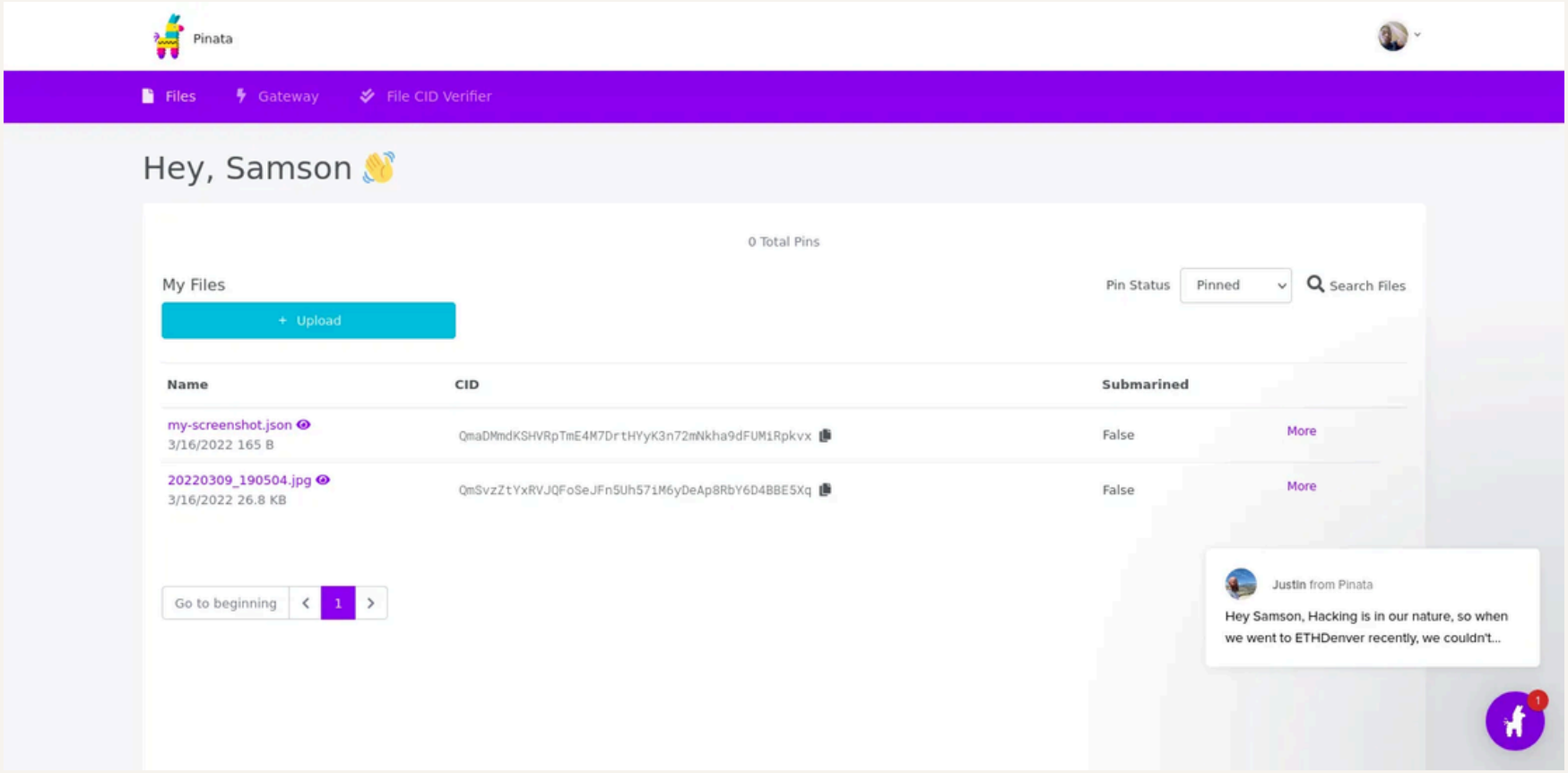
BACKGROUNDS	BEARS WAVY ARMS	HATS
El Greco 4%	**Vibing Brown Bear** 30%	**No Hat** 100%
Floor: 19 BERA	Floor: 19 BERA	Floor: 18.93 BERA
SUNGLASSES	**TSHIRTS**	**WAVY ARM ASSETS**
Snoop Shades 15%	**THC 1** 3%	**Blueberry Vape** 13%
Floor: 19 BERA	Floor: 19 BERA	Floor: 19 BERA

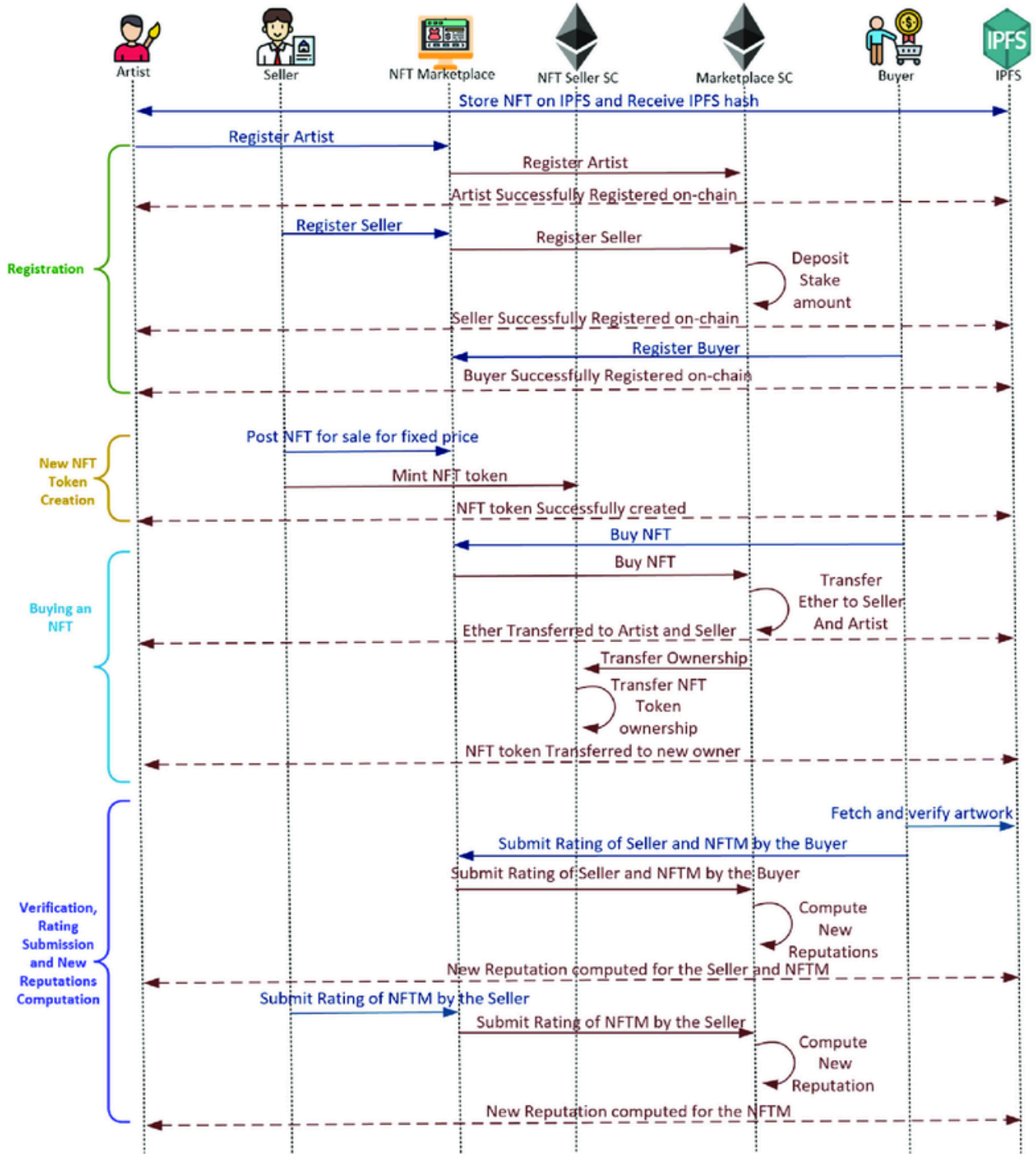
About THC

OPENSEA METADATA STANDARD



IPFS PINNING WITH PINATA




Artist
Seller
NFT Marketplace
NFT Seller SC
Marketplace SC
Buyer
IPFS
Store NFT on IPFS and Receive IPFS hash
Register Artist
Register Artist
Artist Successfully Registered on-chain
Register Seller
Register Seller
Deposit Stake amount
Seller Successfully Registered on-chain
Register Buyer
Buyer Successfully Registered on-chain
Registration
Post NFT for sale for fixed price
Mint NFT token
NFT token Successfully created
New NFT Token Creation
Buy NFT
Buy NFT
Transfer Ether to Seller And Artist
Ether Transferred to Artist and Seller
Transfer Ownership
Transfer NFT Token ownership
NFT token Transferred to new owner
Buying an NFT
Fetch and verify artwork
Submit Rating of Seller and NFTM by the Buyer
Submit Rating of Seller and NFTM by the Buyer
Compute New Reputations
New Reputation computed for the Seller and NFTM
Submit Rating of NFTM by the Seller
Submit Rating of NFTM by the Seller
Compute New Reputation
New Reputation computed for the NFTM
Verification, Rating Submission and New Reputations Computation

REMIX IDE

- A learning platform for developing, deploying and administering ETH Smart Contracts.

Resources Used:
https://miro.medium.com/v2/resize:fit:1400/1*xvkJGTFJ6-rk2SUjS7bHkQ.jpeg
https://medium.com/limechain/atomic-batched-transactions-solving-erc-20-problems-through-metatx-a62e2b85c695
https://docs.opensea.io/docs/metadata-standards
https://medium.com/coinmonks/upload-to-ipfs-using-pinata-bf59ce265ab5