L08: NFTs & Token Standards

Extended Slides – BSc Blockchain Course

Digital Finance

2026

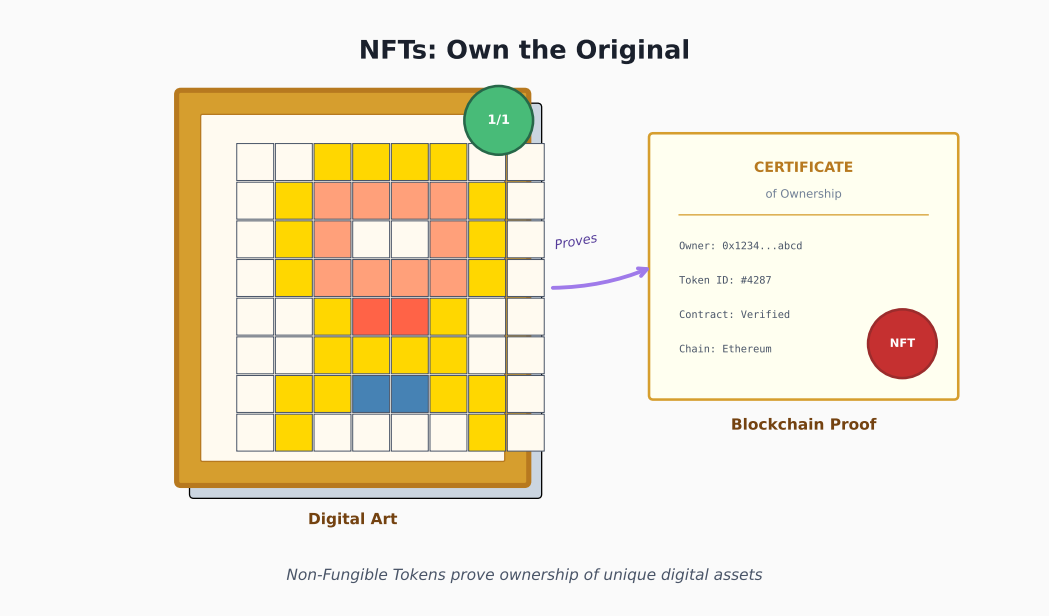
Learning Objectives

By the end of this lesson, you will be able to:

1. Understand ERC-20, ERC-721, and ERC-1155 standards
2. Explain NFT metadata structure and storage options
3. Describe the NFT marketplace ecosystem
4. Analyze tokenomics and distribution models
5. Evaluate NFT use cases beyond digital art

Prerequisites: L05 Ethereum, L06 Solidity.

Owning the Digital



Purpose: NFTs enable verifiable ownership of digital assets. Beyond art speculation, they have applications in gaming, identity, and intellectual property.

The technology behind digital collectibles, but also ticketing, credentials, and more.

What is a Token?

Definition:

- Digital asset on a blockchain
- Defined by smart contract
- Follows a standard interface (ERC-X)

Token vs Coin:

- Coin: native asset (ETH, BTC)
- Token: created by smart contract
- Both can have value and utility

Tokens enabled the ICO boom of 2017.

ERC-20: The Fungible Token Standard

ERC-20: Fungible Token Standard

Required Functions

```
totalSupply()
balanceOf(account)
transfer(to, amount)
allowance(owner, spender)
approve(spender, amount)
transferFrom(from, to, amount)
```

Required Events

```
Transfer(from, to, value)
Approval(owner, spender, value)
```

Optional (Metadata)

```
name()
symbol()
decimals()
```

Key: All tokens are identical and interchangeable (fungible)

Most tokens are ERC-20: USDT, USDC, UNI, LINK, etc.

ERC-20 Deep Dive

Key Functions:

- transfer(): send tokens directly
- approve() + transferFrom(): delegation pattern
- Enables DEX swaps, DeFi integrations

Common Extensions:

- Mintable: create new tokens
- Burnable: destroy tokens
- Pausable: emergency stop
- Permit (EIP-2612): gasless approvals

OpenZeppelin provides audited implementations.

ERC-721: The NFT Standard

ERC-721: Non-Fungible Token Standard

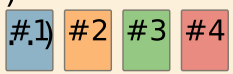
Core Functions

```
balanceOf(owner)
ownerOf(tokenId)
safeTransferFrom(...)
transferFrom(...)
approve(to, tokenId)
setApprovalForAll(...)
```

Metadata Extension

```
name()
symbol()
tokenURI(tokenId)
```

Each token is unique:



Key: Each token has unique tokenId - not interchangeable

CryptoPunks predates ERC-721; wrapped version exists.

ERC-721 Deep Dive

Key Differences from ERC-20:

- Each token has unique tokenId
- ownerOf(tokenId) returns owner; balanceOf returns count only
- safeTransferFrom checks receiver can handle NFTs

Metadata Extension:

- tokenURI(tokenId) returns JSON URL
- JSON contains name, description, image
- Enables wallets/marketplaces to display

ERC-721 Enumerable adds total supply tracking.

ERC-1155 Multi-Token

ERC-1155: Best of Both Worlds

ERC-1155: Multi-Token Standard

Single Contract, Multiple Types:

Fungible:

Non-Fungible:

Key Features:

* Batch transfers
* Mixed fungible/NFT
* Gas efficient
* Single contract

Ideal for: Gaming items (swords x100, unique legendaries)

Created by Enjin for gaming; widely adopted.

ERC-1155 Advantages

Batch Operations:

- Transfer multiple token types in one tx
- Significant gas savings
- safeTransferBatch(from, to, ids, amounts)

Flexibility:

- Fungible: supply ¿ 1, all identical
- Semi-fungible: limited editions
- Non-fungible: supply $= 1$

Ideal for games with items of varying rarity.

Token Metadata Structure

NFT Metadata Structure

NFT Metadata JSON

```
{
  "name": "CryptoPunk #1234",
  "description": "A unique punk",
  "image": "ipfs://Qm...",
  "external_url": "https://...",
  "attributes": [
    {"trait_type": "Background",
     "value": "Blue"},
    {"trait_type": "Rarity",
     "value": "Legendary"}
  ]
}
```

Storage Options

IPFS Decentralized, content-addressed

Arweave Permanent, pay once

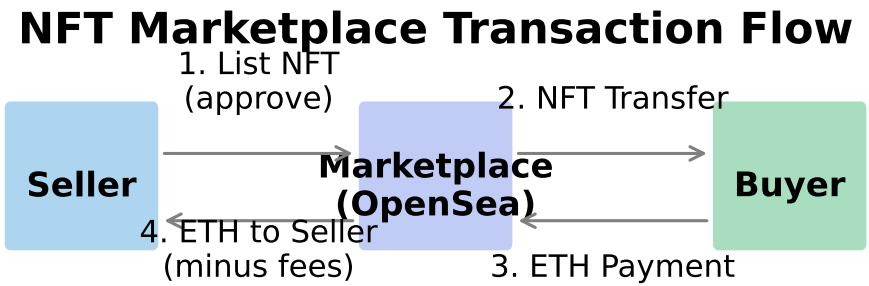
On-chain Most secure, expensive

Centralized Risky, avoid

tokenURI(id) -> returns metadata URL -> JSON -> image URL

Never store metadata on centralized servers.

Marketplace Flow



Fee Structure (typical):

Marketplace: 2.5%

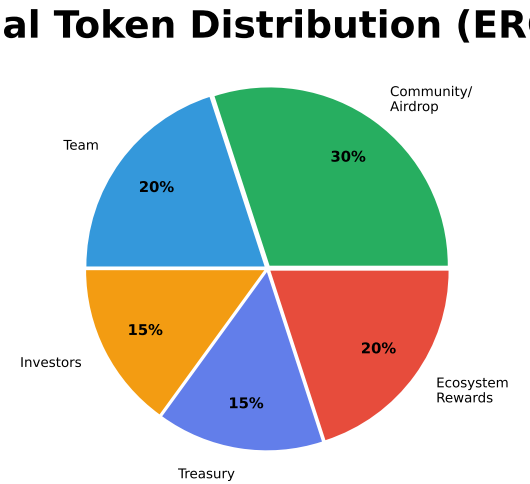
Creator Royalty: 2.5-10%

Gas: variable

Royalties: Creator earns on every resale

Seaport protocol (OpenSea) enables gas-efficient trades.

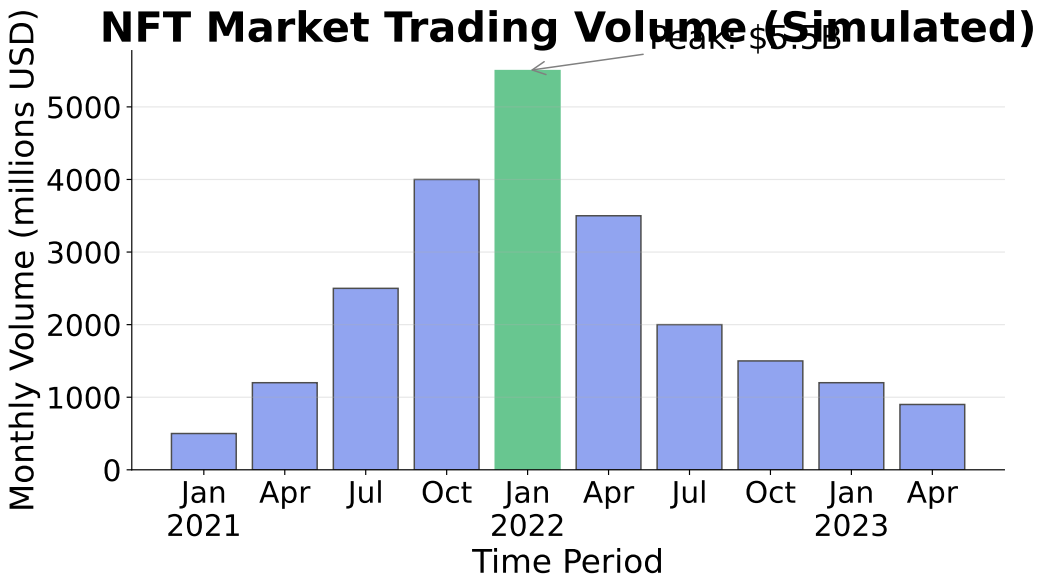
Token Distribution



Note: Team/Investor tokens typically have vesting periods (1-4 years)

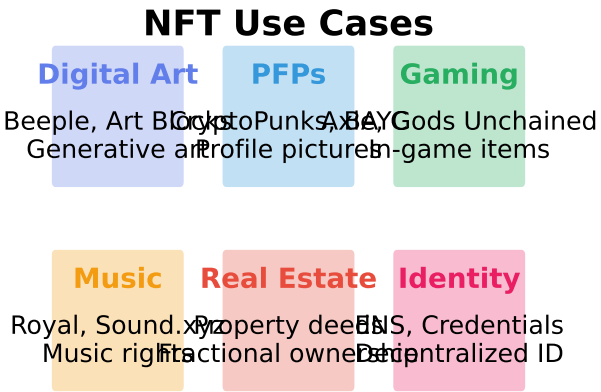
Watch for unlock schedules; can cause price pressure.

NFT Market Volume



Market cooled from 2021-2022 peak.

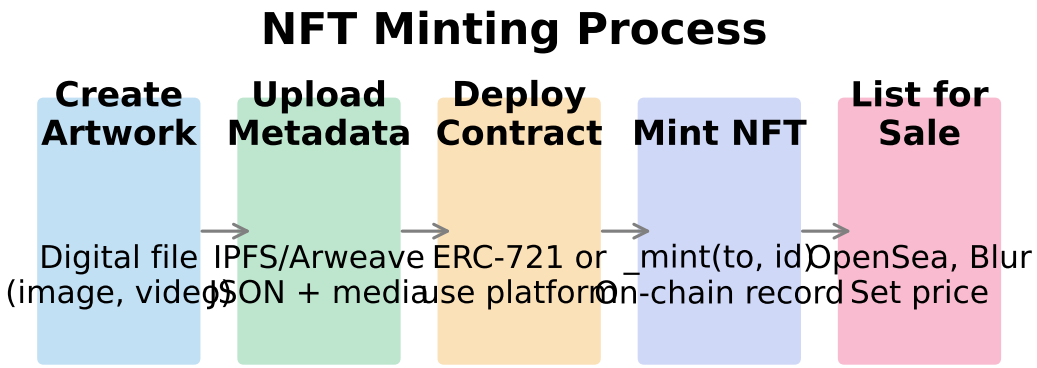
NFT Use Cases



NFTs enable provable ownership and scarcity for any digital asset

Utility NFTs may outlast speculative collections.

Minting Process

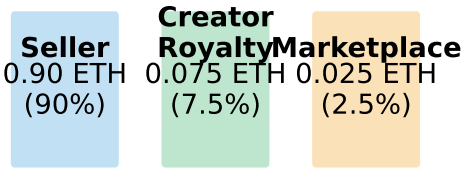


Costs:
Gas for deployment: 0.01-0.1 ETH
Gas per mint: 0.001-0.01 ETH
Lazy minting: buyer pays gas

Lazy minting: metadata stored, buyer pays gas to mint.

Royalties Mechanism

NFT Royalties: Creator Earnings on Resales
Secondary Sale: 1 ETH



EIP-2981: Royalty Standard

`royaltyInfo(tokenId, salePrice) returns (receiver, amount)`

Note: On-chain royalties are optional - marketplaces choose to honor them

Blur's rise challenged royalty enforcement; ongoing debate.

Soulbound Tokens (SBTs)

Non-Transferable NFTs:

- Bound to wallet forever
- Represent credentials, achievements
- Cannot be sold or transferred

Use Cases:

- Diplomas and certifications
- Membership and reputation
- Proof of attendance (POAPs)

Vitalik Buterin proposed SBTs for "decentralized society."

Key Takeaways

Remember These Points

1. ERC-20: fungible tokens, approve/transfer pattern
2. ERC-721: NFTs with unique tokenId
3. ERC-1155: multi-token, batch operations
4. Metadata: IPFS or Arweave for decentralization
5. Royalties: EIP-2981, optional enforcement
6. SBTs: non-transferable credentials

Next Lesson: Layer 2 and Scaling Solutions.