L22: NFT Metadata and IPFS

Module C: NFTs & Digital Assets

Blockchain & Cryptocurrency Course

December 2025

Learning Objectives

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

- Understand the JSON metadata format standard for NFTs
- Explain how IPFS content addressing works
- Describe the role of pinning services in NFT permanence
- Compare IPFS and Arweave for decentralized storage
- Evaluate metadata permanence and availability challenges

NFT Metadata: The JSON Standard

Metadata Structure: OpenSea standard (widely adopted)

Core Fields:

- `name` – Display name of the NFT
- `description` – Human-readable description
- `image` – URI to primary visual asset (IPFS, HTTP)
- `external_url` – Link to project website
- `attributes` – Array of trait objects

Example Metadata JSON:

```
{
  "name": "Bored Ape #1234",
  "image": "ipfs://QmXyZ.../1234.png",
  "attributes": [{"trait_type": "Background", "value": "Blue"}]
}
```

Attributes: Defining Rarity

Attributes Array: Defines traits and properties

Attribute Object Structure:

- `trait_type` – Category name (e.g., "Hat", "Eyes")
- `value` – Specific value (e.g., "Beanie", "Laser Eyes")
- `display_type` (optional) – How to render (number, date, boost)

Rarity Calculation:

- Each trait has a frequency distribution in the collection
- Rarer traits (low frequency) increase NFT value
- Tools like Rarity Sniper calculate rarity scores

Example Rarity Analysis:

- Background: Blue (20% frequency) vs. Gold (2% frequency)
- Laser Eyes: 1% frequency (highly rare)

IPFS: The InterPlanetary File System

IPFS: Decentralized peer-to-peer file storage protocol

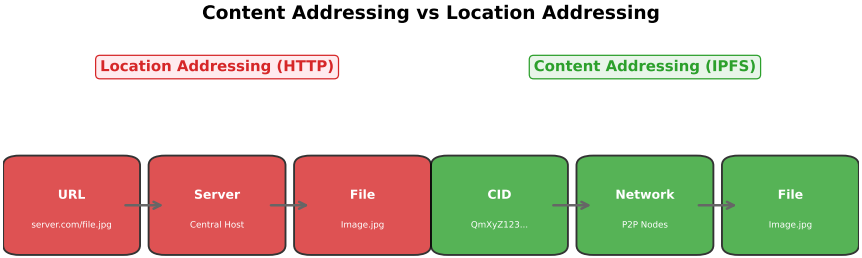
Key Concepts:

- **Content Addressing:** Files identified by cryptographic hash
- **CID (Content Identifier):** Unique hash of file content
- **Immutability:** Same content always produces same CID
- **Distributed:** Files stored across multiple nodes

Example IPFS URI:

- `ipfs://QmXyZ123abc...` (CID)
- Gateway URL: `https://ipfs.io/ipfs/QmXyZ123abc...`

Content Addressing vs Location Addressing



IPFS: Same content always produces same CID, retrievable from any node

Pinning: Ensuring Availability

IPFS Garbage Collection: Nodes delete unpinned files to save space

Without Pinning:

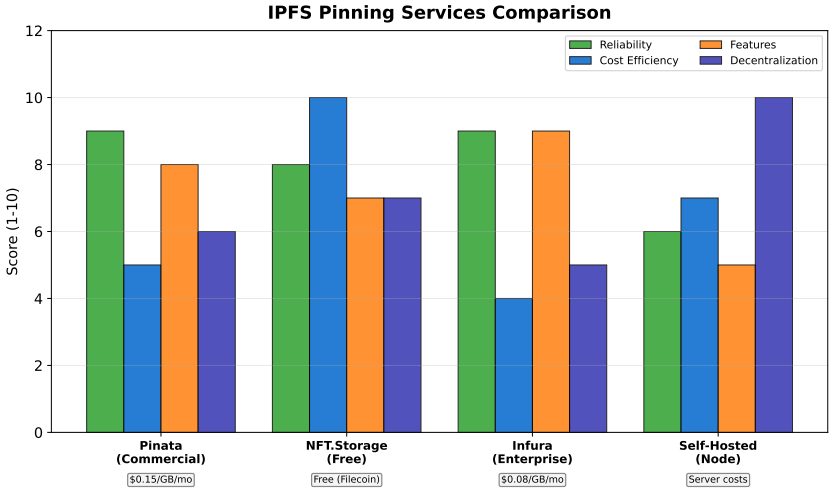
- File may be removed from all nodes (lost)
- NFT metadata/images become unavailable
- Broken image links in wallets and marketplaces

Pinning Services:

- **Pinata:** Popular commercial pinning service
- **NFT.Storage:** Free pinning for NFT data (funded by Filecoin)
- **Infura IPFS:** Enterprise-grade IPFS infrastructure
- **Self-hosting:** Run your own IPFS node and pin files

Critical Issue: If all pinning stops, files may disappear from IPFS

Pinning Services Comparison



NFT.Storage offers free pinning backed by Filecoin; Pinata is most popular commercial option

IPFS Gateways

Gateway: HTTP bridge to access IPFS content

Public Gateways:

- `https://ipfs.io/ipfs/[CID]`
- `https://gateway.pinata.cloud/ipfs/[CID]`
- `https://cloudflare-ipfs.com/ipfs/[CID]`

Why Gateways Matter:

- Browsers do not natively support `ipfs://` protocol
- Wallets and marketplaces use gateways to display NFTs
- Gateway downtime = NFTs appear broken

Centralization Risk:

- Most users rely on centralized gateways (ipfs.io, Cloudflare)
- Solution: Native IPFS support in browsers (IPFS Companion extension)

Arweave: Permanent Decentralized Storage

Arweave: Blockchain designed for permanent data storage

Key Differences from IPFS:

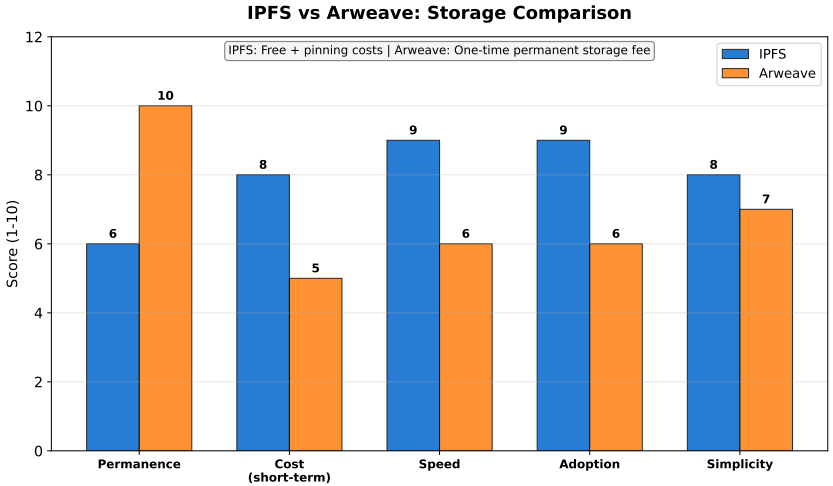
- **Permanence:** One-time payment for perpetual storage
- **Blockchain-based:** Data stored on Arweave blockchain
- **Economic model:** Storage endowment fund pays miners forever
- **No pinning needed:** Data guaranteed to persist

Arweave URI:

- `ar://[Transaction ID]`
- Gateway: `https://arweave.net/[Transaction ID]`

Use Cases: High-value NFTs requiring guaranteed permanence

IPFS vs Arweave Comparison



IPFS for most projects; Arweave for premium NFTs requiring guaranteed permanence

Metadata Permanence Challenges

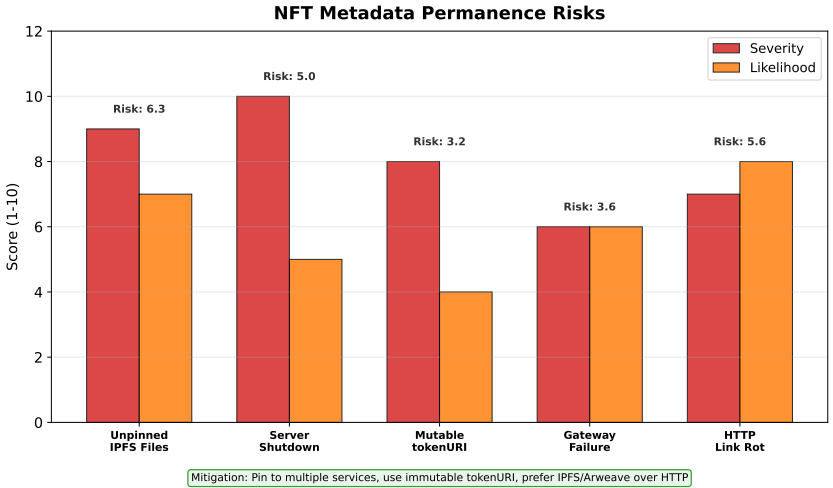
Real-World Issues:

1. **Unpinned IPFS files:** Project abandons pinning, files lost
2. **Centralized metadata servers:** Company shuts down, NFTs break
3. **Mutable tokenURI:** Smart contract allows owner to change metadata
4. **Gateway failures:** IPFS gateways go offline, NFTs appear broken
5. **Link rot:** HTTP URLs stop working (404 errors)

Case Study: Nifty Gateway (2021):

- Platform used centralized servers for metadata
- Outage caused all NFTs to display broken images
- Community backlash led to IPFS migration

Metadata Permanence Risks



Server shutdown has highest severity; HTTP link rot is most likely to occur

Mutable vs. Immutable Metadata

Mutable Metadata:

- Smart contract owner can update `tokenURI`
- Allows bug fixes and metadata improvements
- Risk: Owner could change artwork or traits (rug pull)

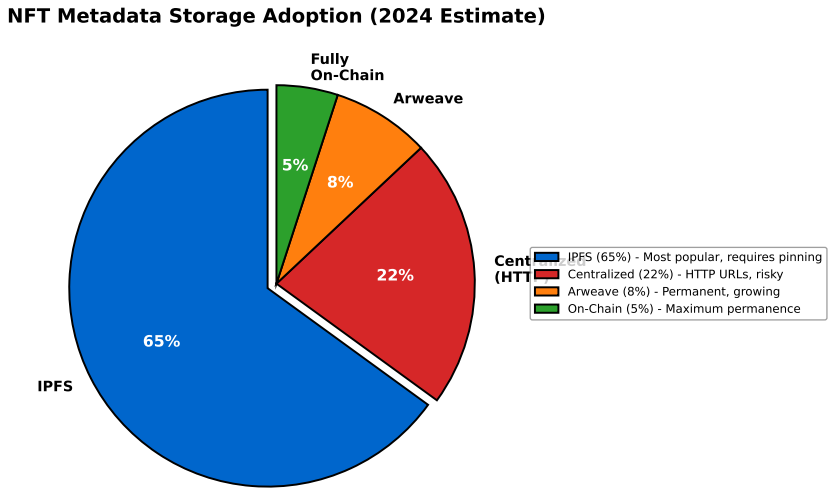
Immutable Metadata:

- `tokenURI` frozen after minting (contract locked)
- Guarantees metadata cannot be altered
- Standard for high-value collections (e.g., CryptoPunks)

Verification:

- Check smart contract code for `setTokenURI()` functions
- Verify contract ownership is renounced (no admin control)
- Use Etherscan to audit contract mutability

NFT Storage Adoption



IPFS dominates NFT storage; Arweave growing for premium projects

Best Practices for NFT Metadata

For NFT Projects:

1. **Use IPFS or Arweave:** Avoid centralized servers
2. **Pin all files:** Use reputable pinning services (Pinata, NFT.Storage)
3. **Freeze metadata:** Make `tokenURI` immutable after reveal
4. **Redundancy:** Pin to multiple services (IPFS + Arweave)
5. **Document storage:** Inform buyers where metadata is hosted

For NFT Buyers:

- Verify metadata is on IPFS/Arweave (not HTTP)
- Check if `tokenURI` is immutable
- Confirm pinning service reputation

Key Takeaways

1. NFT metadata follows a JSON standard (name, description, image, attributes)
2. IPFS uses content addressing (CIDs) for decentralized, immutable storage
3. Pinning is critical for IPFS file permanence (unpinned files can be lost)
4. Arweave provides guaranteed permanent storage for a one-time fee
5. Metadata permanence challenges include unpinned files and mutable URIs
6. Best practice: Use IPFS/Arweave, pin to multiple services, freeze metadata

Discussion Questions

1. What are the trade-offs between IPFS and Arweave for NFT metadata storage?
2. Should NFT metadata be immutable, or is mutability acceptable for bug fixes?
3. How can the NFT community ensure long-term metadata availability?
4. Is fully on-chain metadata the ideal, or are off-chain solutions sufficient?
5. What happens to NFT value if metadata becomes unavailable?

Next Lesson Preview

L23: NFT Marketplaces

We will explore:

- OpenSea, Blur, and Rarible business models
- Listing mechanics and order book systems
- Marketplace fees and royalty enforcement
- Wash trading and market manipulation detection
- Aggregators and cross-marketplace trading

Preparation: Create a wallet and browse NFT collections on OpenSea