SHARE: Decentralized distribution using peer-to-peer, pay-for-access micro-transactions on blockchain

Brandon Tory Thorpe
brandon@formless.xyz
www.formless.xyz

Abstract. The evolution of music distribution is largely characterized by incremental developments based on technology advancements. From early recordings and cassettes to digital media and MP3s, a song has always lived as a static piece of information interpreted by a machine to transfer an experience between humans. The problem with this trend is that the machine vendor has always established the terms of access to the song. Along every step in the progression, this model has crushed creators—eliminating any control over pricing, and the ability to directly connect with consumers. In this paper we propose a technology advancement that disrupts this trend by redefining digital content (such as music), not as static information, but as a dynamic self-contained machine. A simple analogy is the vending machine. A vending machine can be placed anywhere in the world, and its owner controls its access independent of the property on which it operates.

SHARE is a protocol that enables creators to make orders of magnitude more money for their digital content by eliminating the *dependence* on centralized content streaming machines (e.g. Apple, Spotify, and YouTube) altogether. Instead, each piece of content encapsulates its own implementation of its access policy, independent of the application used to experience it. This "smart-content" itself is the machine. By expanding the definition of digital content (embodied by creator-controlled smart contracts) beyond a static piece of information, and reimagining distribution using peer-to-peer, pay-for-access micro-transactions on blockchain, any streaming platform can instantly authenticate for access to a piece of digital content, execute an instant micropayment based on creator specified terms, and stream the experience to consumers. Most importantly, the price of the micropayment is set by the creator, not the platform. SHARE inverts the access model.

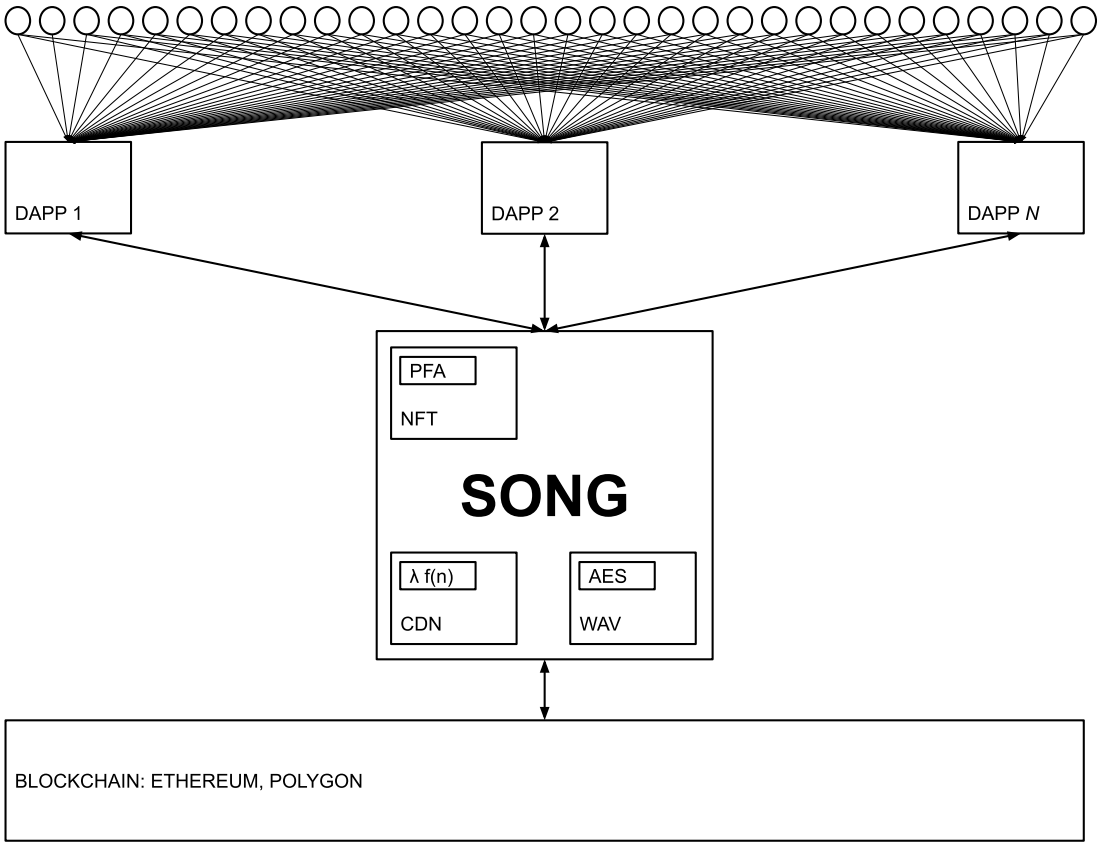

Figure 1: This figure demonstrates features which are traditionally vendor provided, instead being pulled directly into the conceptual boundary of the digital content itself, in this case a song. In doing so, any machine can operate the song, and the song globally administers its own operation.

1. Background

Smart contracts have established a public, immutable chain of digital asset ownership, unlocking a world of possibilities for trading value on permissionless exchanges. What we haven't seen as widely used yet are the capabilities which go beyond the NFT (Non-Fungible Token) standard, allowing content creators to specify additional behavior within their work. In some cases this behavior has been used to create generative content, e.g. art that changes as the result of code execution. However, in this paper we explore the possibilities of specifying the rules of access, rather than the generative qualities. We posit that in addition to scalable ownership this is a fundamental feature of Web3.

We define a living data format, or an abstract machine, as a uniquely tokenized asset which allows digital content creators to set their prices, and receive instant payments, in perpetuity, independent of the machine operator. Historically the thought of a living data format is analogous to a unique "program" or "microservice" that executes code. The technical challenge here is that mainstream servers which host creator content do not allow remote code execution (RCE) for a number of valid reasons. Also, it is unreasonable to expect creators to maintain servers which host executable content themselves.

Past solutions involved Digital Rights Management software (DRM), which was problematic in that it tightly coupled creator content with proprietary distribution channel software and hardware. Blockchain changes this since virtually any server can now execute small portions of *creator-specified* code in a safe, transactional environment known as the Ethereum Virtual Machine (EVM).

SHARE is a protocol which uniquely enables creators to establish the rules of access for their digital content, independent of the distribution channel. Using the SHARE protocol, distribution platforms and Web3 user interfaces bend to the rules of creators (e.g. price to buy, price to experience, price to advertise etc.), rather than the inverse.

2. Creator Economy

The mainstream creator economy is broken. The price paid to creators is static, arbitrary, and determined solely by the large platforms (like YouTube and Spotify) that distribute the content, eliminating free market price discovery, and instead optimizing for their own profits. Unlike other industries where owners set their product price based on a number of unique business factors, blanket pay-per-click rates are applied to digital content, creating an uber-competitive, clickbait driven ecosystem, largely monetized using pooled ad revenue and subscriptions.

To overcome this root cause economic problem where the natural supply and demand of digital content is artificially restrained by centralized intermediaries, SHARE adds a protocol layer within the smart contract infrastructure to enable creator-controlled pay-for-access (PFA) to NFTs.

3. Non-Fungible Tokens (NFTs)

It is generally well known that the supply and demand problem mentioned above is being partially solved today with NFTs. For example, we're seeing a transformation where creators set their own prices for the *sale* of their content, but not the price to *experience* it. In addition, mainstream users tend to use the best user experience—and the best user experiences are often created by those with access to capital. Therefore, if creators are not intentional about influencing contract and protocol level decisions, those with access to the most capital will shape the business model for creators, again.

SHARE takes advantage of the fact that creators have incredible leverage at the content layer of the Web3 stack. This means that while NFT standards are controlled by the research community and applications are largely controlled by tech industry incentives, under the SHARE protocol the token URI, metadata schema, and content containers used to display NFT experiences remain *creator-controlled* territory.

4. Creator Code Execution

To illustrate the power of Web3, and by extension SHARE, consider the fact that a creator cannot execute even the simplest of their own code on a YouTube server. Even if one could, this code would not work on a Spotify server. Blockchain provides an execution environment for running creator specified code. While it can be costly to run this code, we posit that the

code only requires very limited stateful behavior to have a massive impact: confirming financial transactions and updating state variables readable by decentralized content delivery (distribution) networks.

As a result, the following code can be executed from any server: "Check if entity *N* paid what I've asked for", and provide access to the content. Many NFTs today are not exploiting this fact simply because the ERC-721 (Ethereum Request for Comment Number 721) token metadata standard was not explicitly designed for content creators, but rather for maximum flexibility. SHARE introduces a layer which uses this metadata interface as a transaction-based interlock between producer and consumer at every observable entry point of the content.

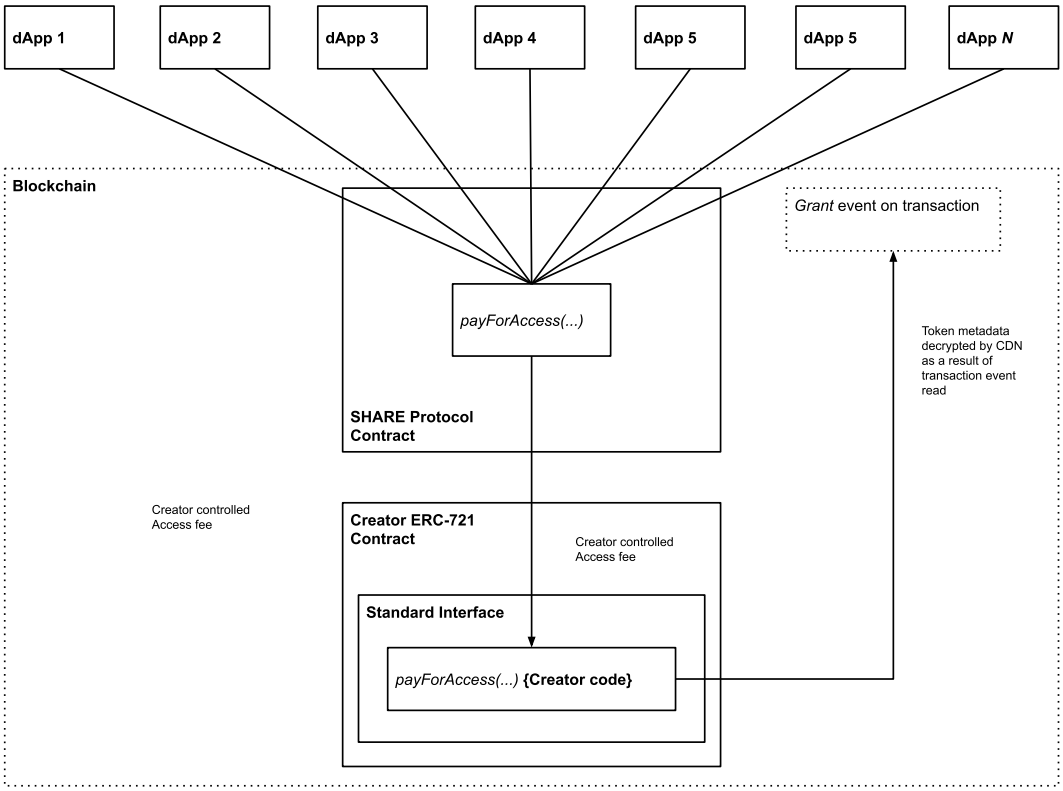

Figure 2: The entity relationship between an unbounded number of decentralized applications, the SHARE protocol contract, and creator NFTs. Note that the Solidity code which ultimately confirms the payment transaction for access is housed within the creator-controlled contract.

Standard NFT token metadata containing the content URI is read from the NFT by the decentralized application, however the content which the URI points to is decrypted if, and only if, the payment transaction event can be found on the blockchain for the address of the viewer. The concept of moving all policy terms from the application layer into the creator-controlled layer is analogous to the dependency inversion pattern from software engineering. We call this *access inversion*, which can be visualized in Figure 3.

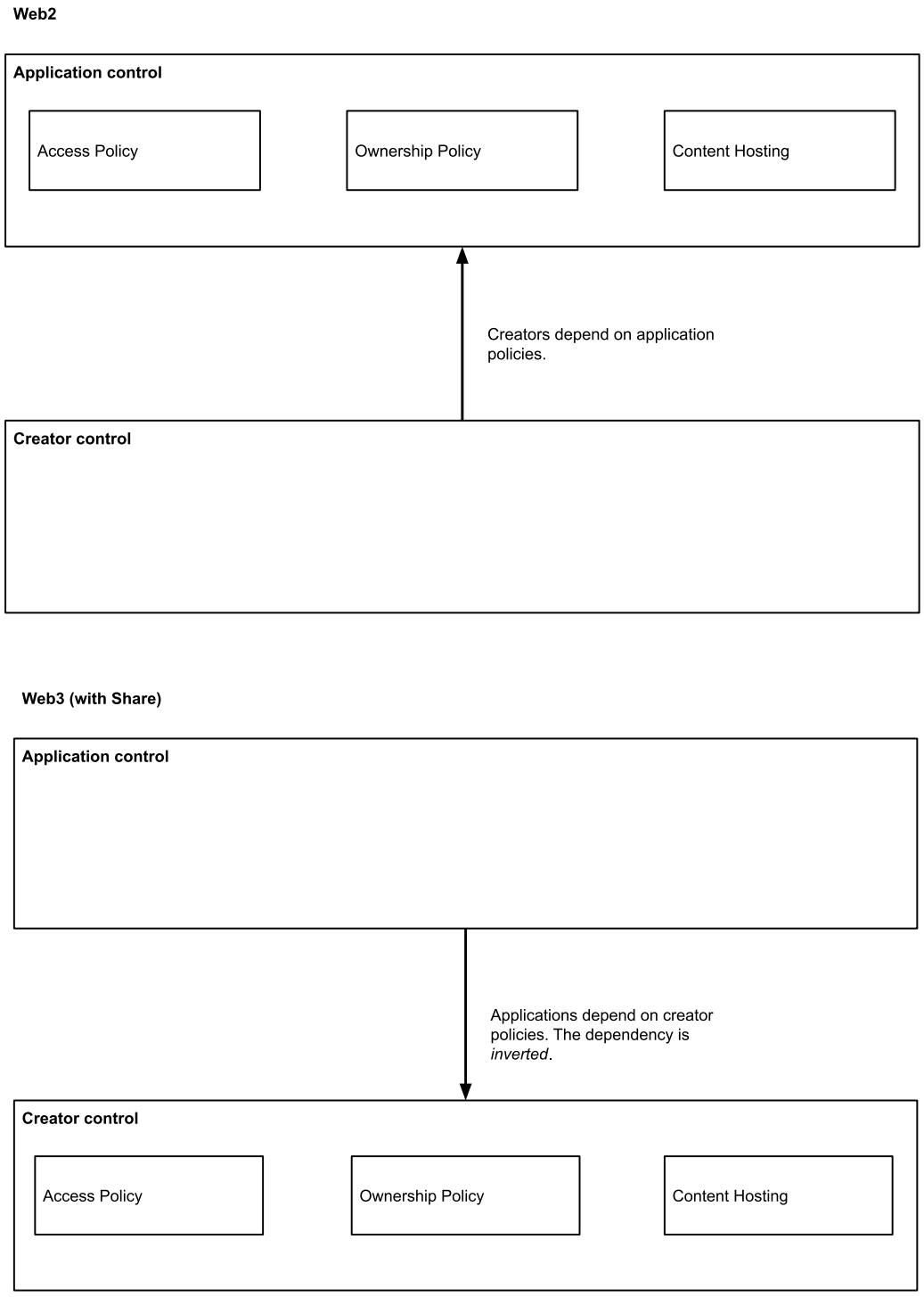

Figure 3: Architectural differences between the past (Web2) and solutions that exist today (Web3).

In Web2, the terms of access for creator content, including the code which sets the price-per-stream, are completely specified, and executed on 3rd party centralized infrastructure,

while the ownership terms and content are hosted on the same centralized application service layer.

In today's Web3, the ownership terms and content are pulled out of the centralized application layer and into the blockchain layer, giving creators sovereignty over those terms. However, in Web3 with SHARE, the dependency is *fully* inverted, as the access terms are also pulled into the blockchain layer. This means that creators don't just set the price for owning the content, they set the price for *experiencing* the content. Both Web2 and Web3 applications can unlock creator content using the SHARE contract on blockchain which actuates the unlock mechanism within the creator smart contract in order to serve the decrypted content URI.

5. Governance

Decentralized communities often require governance. In Web3 this is typically achieved using a governance token, which may also act as a utility token to incentivize behaviors within the community. Governance is a good solution to many problems in the decentralized internet; however, it is often insufficient for creators.

Governance enables power structures (e.g. community members with unusually large access to capital or social capital) to re-emerge within Web3, and, if not properly checked, introduce business terms that may not make sense for creators.

Consider Web3 platforms emerging today with relatively arbitrary features and rules. In addition to decentralizing the control (e.g. putting it into the hands of the community) SHARE fully inverts the terms of access. This means that any terms regarding the rules of access, such as pricing and hosting, are governed by the creator alone.

To do this SHARE adds an additional layer to the NFT backbone which specifies how the actual content is mechanically actuated from within the token. This is largely an afterthought in NFT communities today. Many creators are minting NFTs as token identifiers within aggregate contracts controlled by a governance structure rather than minting their own ERC-721 implementations. These contracts use persistent plaintext storage like CDNs or IPFS to store the actual content file. By encapsulating the content file in a standard interface, SHARE takes advantage of this as an opportunity to introduce creator control and additional revenue streams for the creator ecosystem.

6. Pay-For-Access (PFA) Smart Contracts

At the core of SHARE is the idea of a pay-for-access smart contract. The price per access (e.g. view or listen) can be set to zero as is today for nearly all NFTs, but it can also be set to a non-zero value. This means that content will exist which cannot be observed until a micro-payment is recorded on the blockchain. In order to enable high-volume transactions at low cost, SHARE is built on the Polygon Proof-of-Stake (PoS) sidechain, and will support both additional chains, and new EVM compatible ZK rollups as they launch. Additional details are discussed below in "Layer 2 Optimizations".

Unlockable PFA experiences at scale can create immense impact for creators. At high volume, micro payment transactions on observables are much more sustainable than trying to sell a single NFT for a high price to one buyer. Rallying a community to establish value around an NFT can be a stressful endeavor for artists, and an alternative model is to charge a creator-

specified price for the ability to experience the art. With SHARE, the art is the utility. By putting more content monetization options (PFA, pricing, and sponsor revenue) into the hands of creators, SHARE enables them to choose which combination of models work best for their business.

We call the default implementation of the PFA interface that conforms to the SHARE protocol a "Green NFT (G-NFT)". Green is often associated with the color of life, renewal, nature, and energy. G-NFTs are non-fungible tokens with living, creator-controlled content. G-NFT content is conceptually living and revenue streams flow to creators on each access for the lifetime of the art, independent of a sale of the contract itself.

Applications may also act as agents on behalf of users by paying for the PFA content access using their own revenue and presenting the experience to users for free, or at a surcharge. This unlocks new business opportunities and market competition for application developers using SHARE.

```
interface PFA {
    /// @dev This emits when the recipient address has
    /// successfully been granted access to the NFT content
    /// associated with the tokenId. The content
    /// can be read using the ERC 721 Metadata standard,
    /// e.g. tokenURI. The returned metadata
    /// will include the content decrypted by the SHARE CDN, after ACK of
    /// the interlock "Grant" event emitted within the
    /// creator controlled access function.
    event Grant(address indexed _recipient, uint256 indexed _tokenId);

    /// @notice Returns non-zero value in gwei if this NFT
    /// requires payment for access and zero otherwise.
    function pricePerAccess() external returns (uint256);

    /// @notice Grants or denies access to NFT content based on
    /// creator controlled terms.
    /// @param _tokenId The identifier for an NFT
    function access(uint256 _tokenId) external payable;
}
```

7. Example: Pay-For-Access Content in the Metaverse with SHARE

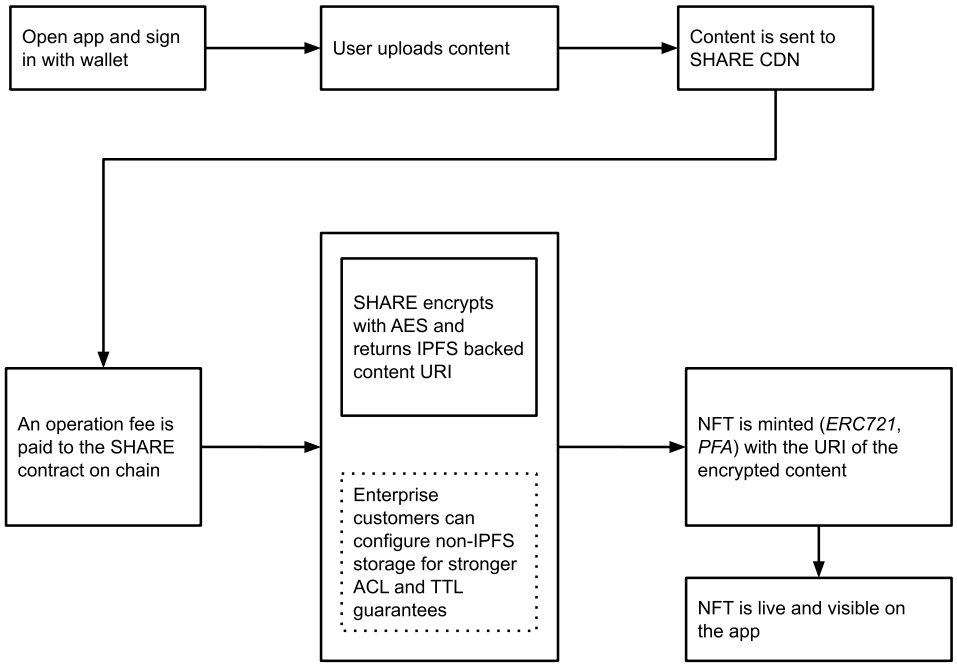

Figure 4: A user journey for **creating** a Pay-For-Access NFT using SHARE and any social media or streaming application in the metaverse.

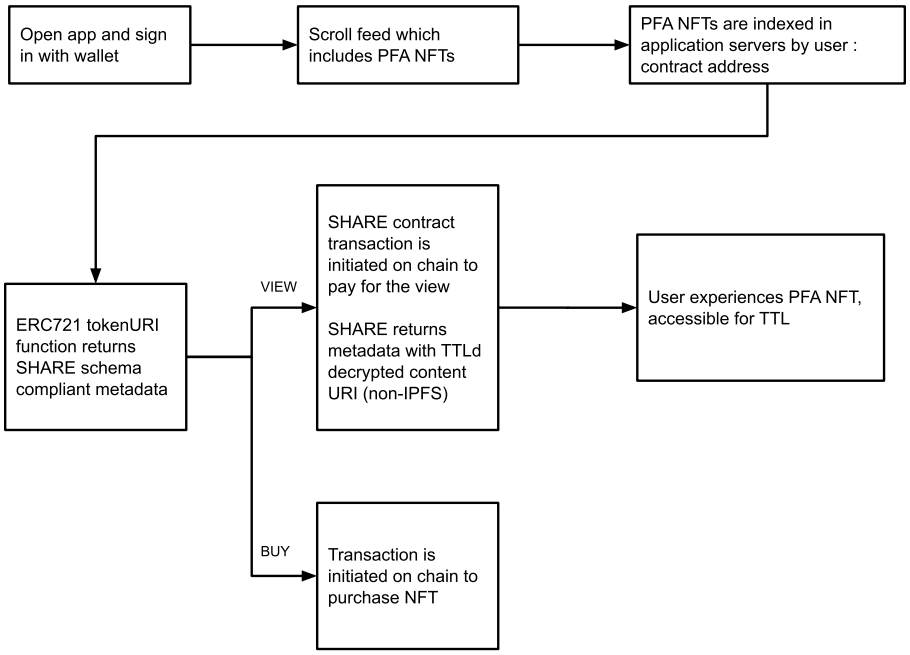

Figure 5: A user journey for **experiencing** a PFA NFT using SHARE and any social media or streaming application in the metaverse.

8. Multi-Chain and Multi-Cloud Decentralized Distribution Networks

Decentralized distribution networks (DDNs) are special microservices or interplanetary filesystem (IPFS) gateways that host creator content using content-level encryption and blockchain transaction gated unlocks. Creator smart contracts implement a simple unlock function that writes a transaction event to the chain for delivery network nodes to read, resulting in decryption before serving the token URI.

If the transaction event is not confirmed, the token URI is still served, however the content remains encrypted, and a preview URI is supplied within the associated metadata. To prevent plaintext storage of premium content on the public IPFS, after payment, SHARE returns a TTLd version of the content hosted on cloud storage which lasts for the duration of the access term (e.g. short term storage). Access grant lifetimes are measured as creator specified time spans beginning at the moment the grant event is recorded on chain.

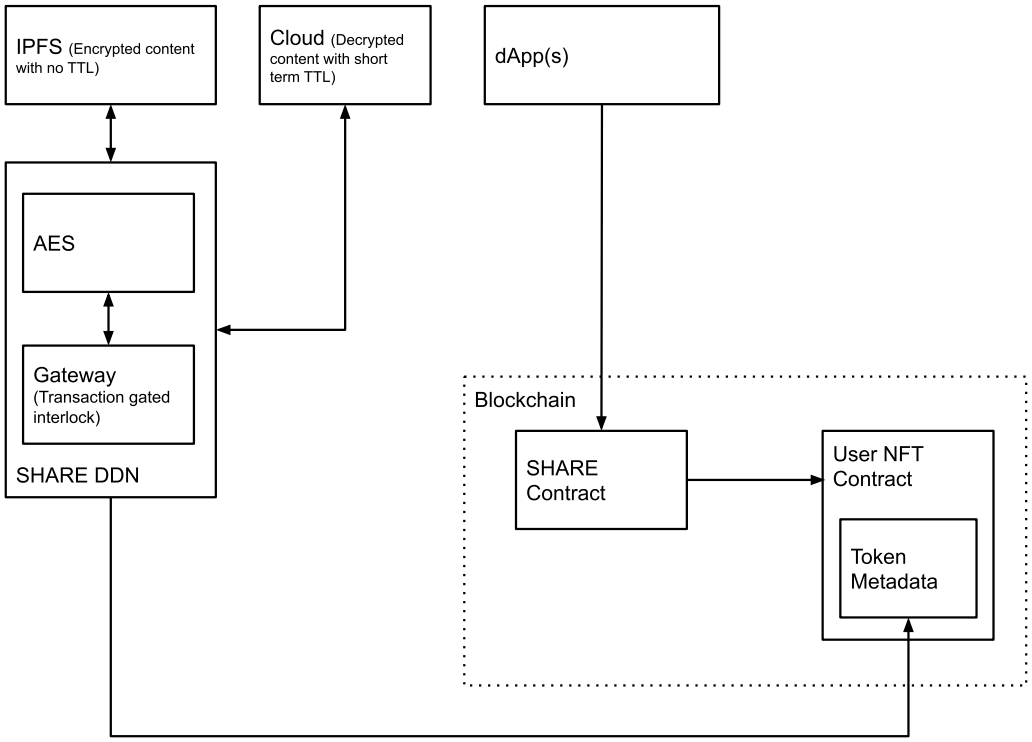

Figure 6: The entity relationship between SHARE DDNs, IPFS and NFTs which use SHARE DDNs for serving creator-controlled token URIs. SHARE introduces a contract between the tokenURI, metadata, and the DDN which unlocks content based on the presence of blockchain micro transactions and honors the rules specified in the creator contract. This system contains no single point of failure since at any time a creator can update their contract code to point to a different IPFS gateway or DDN.

Decentralization is achieved using a multi-chain and multi-cloud architecture. Within the network, creators can choose which chain and which cloud to use. This is mutable in the PFA contract, meaning that if a cloud service provider introduces terms which are no longer

favorable, creators can change providers with no impact to end users. In fact, creators can elide the cloud altogether and host DDN microservices themselves, however cloud services offer an incredibly inexpensive, efficient, and simple mechanism for doing exactly this.
We use the NFT only as the infrastructure to transact ownership, and the metadata schema as the contractual mechanism to dictate creative control. We believe that creators today have the leverage to dictate this interface, and applications will adapt.

SHARE DDNs also comprise "sponsorship delivery". These are a set of decentralized services which deliver sponsored content into user experiences based on creator specified terms. This means NFT creators can enable ad revenue on their NFTs. In the SHARE protocol, creators specify their own terms for ads—setting the price and choosing the advertisers. DDNs execute creator code on the EVM and serve the content to the presentation layer.

The vision is to enable artist curated sponsorships that are thoughtfully placed and non-disruptive to the user experience, rather than bringing traditional advertising to Web3. We believe there's an aesthetically pleasing balance to be achieved which will mutually benefit creators, consumers, and sponsoring brands.

9. Metadata Schema

The code examples below show a JSON schema for SHARE NFT metadata which includes standard ERC-721 Metadata JSON Schema keys: title, name, description, image, as well as SHARE protocol metadata.

```
[
  {
    "title": "SONG XYZ",
    "properties": {
      "name": "SONG XYZ",
      "description": "NOP",
      "image": "NOP",
      "artist": "ARTIST X COLLABORATOR",
      "type": "audio",
      "uri": "share.ddn.xyz/1",
      "artwork": {
        "properties": {
          "type": "image",
          "uri": "share.ddn.xyz/2"
        }
      }
    },
    "sponsors": [
      {
        "title": "Nike",
        "properties": {
          "message": "Express your versatility, score from
anywhere.",
          "uri": "share.ddn.xyz/3",
```

```
            "link": "https://nike.com"
          }
        }
      ]
    },
    {
      "title": "VIDEO XYZ",
      "properties": {
        "name": "VIDEO XYZ",
        "description": "NOP",
        "image": "NOP",
        "artist": "ARTIST X COLLABORATOR",
        "type": "video",
        "uri": "share.ddn.xyz/4"
      },
      "sponsors": [
        {
          "title": "Adidas",
          "properties": {
            "message": "Into the metaverse.",
            "uri": "share.ddn.xyz/5",
            "link": "https://adidas.com"
          }
        }
      ]
    }
  ]
```

10. Layer 2 Optimizations

For PFA NFTs with unlockable experiences to work a near zero transaction fee is desired, otherwise the transaction fee might strongly exceed the price for access. Transaction fees are a well-known bottleneck for micropayments in Web3.

Layer 2s (L2s) are blockchain scalability solutions which increase transaction throughput and decrease transaction fees. L2s include rollups (e.g. Optimistic and ZK), which inherit security from the Ethereum layer 1 (L1) chain. Side chains are not conventionally considered L2 scaling solutions but are networks of nodes running in parallel to the Ethereum main chain, and which periodically checkpoint state to a main chain contract using the Merkle tree root hash corresponding to the set of transactions between state transitions. This scheme enables low-cost transactions and asset bridging between the side and main chain.

SHARE aims to be a multi-chain solution which enables creators to take advantage of L1s, L2s, and side chains, based on what works best for their business. In the initial implementation, SHARE supports the Ethereum main chain and the Polygon PoS side chain. SHARE will expand to support EVM compatible ZK rollups (e.g. zkEVM) as they become available.

11. Ownership, Trading and Fractionalization

NFT ownership, trading, and fractionalization opportunities exist in addition to the PFA contract terms. Content which adopts the SHARE protocol can be viewed on any platform which supports the ERC-721 Metadata JSON Schema.

Fractionalization of NFTs is another monetization opportunity for artists. From a technical perspective, fractionalizing the copyright of a SHARE protocol compliant NFT involves the creation of an ERC-20 custodial contract which exposes fractional token-based ownership of the copyright that underlies the referenced root ERC-721 token. The ERC-721 token owner is then set to the address of the ERC-20 contract.

ERC-20 tokens can be exchanged on any decentralized exchange where liquidity for the token exists, and initial liquidity can be provided by the creator by staking a pair which comprises their ERC-20 and another liquid asset such as ETH. A creator-executed copyright assignment agreement is encapsulated within the ERC-721 metadata and denotes token holders as rights holders of the copyright. This is done under the hood in the SHARE application and is an optional feature for creators that wish to fractionalize ownership.

All revenues generated from PFA sales and asset ownership sales are sharded according to the rights holders as specified by their addresses in the creator contract. For large computations (a large number of fractions) off-chain processing is done using blockchain indexing services such as The Graph) at predetermined intervals and accrued revenues are available for withdrawal from the contract at those times. By default, non-fractionalized NFTs with a small number of collaborators are configured to send instant payment to contract owners at the time of the access grant transaction.

SHARE fully complies with SEC definitions around securities and does not endorse any market making practices for fractionalized NFTs which would violate SEC guidelines now or in the future.

12. Digital Rights Management and Piracy

As mentioned, DRM combines vendor software with digital content to lock the content within a vendor-controlled product ecosystem. In SHARE, the logic which grants access to content is not controlled by any vendor, or group of vendors, but instead by creators and executed on a public blockchain based virtual machine. SHARE is not DRM.

With that in mind, there is an unavoidable, relatively small portion of the global population that will by technical means find a way to illegally redistribute content which is gated by pay-for-access contracts. If not by technical means, then the well-known analog-hole problem imposes a physical limit to the ability to prevent retransmission of any form of intellectual property.

Rather than trying to block all vectors for piracy through engineering alone, instead, we rely on the fact that the *effort* cost to consume pirated content (E_P) is increasingly higher than the effort cost to legally access content (E_L). Furthermore, pirated content is easily attributable using modern digital fingerprinting and therefore there is some maximum time (T) that duplicate content can exist on any *legally* managed platform accessible to mainstream consumers before a takedown is issued by the owner.

If *T* is small, and $E_P >> E_L$ then theoretically virtually all creator revenues (R_c) generated by content uploaded to the SHARE decentralized distribution network will flow to the rightful owners with some anticipated loss of *L*.

R_c - L (creator net revenue) on SHARE is still higher than net revenue on any platform which exists today and therefore the cost of piracy is negligible.

13. Sponsor Tools

SHARE will create backend tools for sponsors that enable them to publish ads and identify creators accepting ad buy proposals. SHARE content sponsorship is by default charged by duration. This means creators can offer ad slots with terms such as "image below video for 30 days for $*N*" and advertisers can pay for this package in its entirety up front.

Since smart contracts are transparent and fully visible on the blockchain, transaction activity and revenue are available to advertisers. SHARE can also provide more advanced tools to optimize advertiser spending. The SHARE protocol represents advertisements as smart contracts stored on chain, enabling artists to have complete control of the code which specifies the terms of an acceptable advertiser. Once an ad contract is purchased, DDNs automatically serve approved ads as IPFS backed content URIs within token metadata as shown in the above schema.

14. Streaming, Curation and Playlisting

In SHARE, all experiences: songs, videos, albums, and playlists are PFA smart contracts. While a single song may have a PFA price of $0.50, an entire album can also be represented as a PFA with a price of $5.00. Similarly, a curated playlist with 20 songs can be represented as a PFA with the price set by the curator, and the revenue flowing proportionally to all songs included in the playlist. All revenue is then automatically distributed to any co-creators specified in the underlying smart contracts.

This model gives creators, curators, and consumers maximum flexibility for creating, experiencing, and *reselling* revenue generating PFA contracts on secondary markets. In the future we envision many PFA friendly applications using SHARE that provide users with incredible experiences and can independently access the same content from a global ledger while paying rights holders fairly and efficiently.

15. Conclusion

We believe music distribution will go in one of two directions. In the first, music will be distributed on blockchain with creator-controlled pricing (e.g. using the SHARE protocol), and since no exclusive sub-license agreement exists between creators and any set of frontend applications, the marketplace for those applications will *explode*. This means that any team capable of building a Web3 application can compete for users that want to listen to mainstream music.

In the second, music will still be distributed on blockchain with creator-controlled pricing, but companies will recognize the value of the decentralized distribution protocol. In this case,

companies such as Spotify and Apple Music will become *consumers of the protocol*—adhering to creator controlled smart contract terms.

In both scenarios, the monopoly advantage enjoyed by centralized access points functioning as the primary financial intermediary through which users can transact in exchange for access to music, will be gone. The future of music distribution is the *inversion* of the power structure from machine vendors to content creators.

Further Reading

1. Ethereum Whitepaper
2. Ethereum Yellowpaper
3. Polygon MATIC Whitepaper
4. HDCP Whitepaper
5. ERC-721 Non-Fungible Token Standard
6. EIP 864: Divisible Non-Fungible Tokens (Shared Ownership Over NFTs)
7. IPFS - Content Addressed, Versioned, P2P File System
8. The Graph: A Decentralized Query Protocol for Blockchains
9. Thoughts on Music by Steve Jobs
10. zkSync Trustless Protocol for Scalable Low-Cost Payments on Ethereum: zkEVM
11. Plasma: Scalable Autonomous Smart Contracts
12. An Incomplete Guide to Rollups