

Tune Token



www.TuneToken.io



Tune Software:

*A Democratized Ecosystem
for the Music Industry*

VERSION 1.2 October 2018

AUTHORS:

Bushido Lab, LLC

REVIEWERS:

Erik Mendelson, CEO/Co-Founder at Tune Software
Winston Thomas, Co-Founder at Tune Software
Shawn Mims, Co-Founder at Tune Software

ABSTRACT

PLEASE REVIEW THE SECTIONS BEGINNING ON PAGE 5 CAREFULLY AS THEY SET FORTH MATERIAL REPRESENTATIONS AND WARRANTIES DEEMED MADE BY EACH RECIPIENT OF THIS WHITEPAPER AND OTHER IMPORTANT INFORMATION ABOUT TUNE AND THE CONTENTS OF THIS WHITEPAPER.

This white paper outline's the systematic use case and application of a blockchain enabled distributed platform for the music community. We have proposed an economic model where every participant in the creation and distribution process including: producers, artists, songwriters, users and music fans are incentivized through internal platform cryptocurrency rewards. Tune Software's Decentralized Ledger Technology (DLT) platform and the creation of the TUNE Token will be the driving force for redefining the music industry.

The TUNE token will serve as the primary method of exchange for goods and services in our decentralized music ecosystem. Tune Software's first licensing partner is the award-winning application, Cre8tor.app (formerly RecordGram). Cre8tor.app (formerly RecordGram) will be the first application to adopt Tune's proposed tokenized music ecosystem. We are here to democratize what should have been democratized in the first place. We are here to implement blockchain technology that creates a self-sustaining ecosystem that rewards all parties involved in the creation and discovery of music while compiling a music rights database that is built on transparency.

Table of Contents

Introduction & Vision	5
The \$2.5B Problem	6
Solution	7
Launch: Current Integration	8
Scale: Protocol and Utility	8
Content Storage	8
Transaction Scalability	9
Tune Overview	10
Economic Incentives	11
User Onboarding	12
Token Utility & Protocol Dynamics	13
Protocol	13
Tune Credits and Network Tokens	13
Stakeholders	14
Technology Stack	17
Immediate Implementation	17
Hyperledger	17
Ethereum	24
Long-Term R&D	27
IPFS	27
Content Security	30
DApp Architecture	31
Payment Service	33
Technical Assessment of Tokens (RecordGram -> Tune)	36
Content Credits: eCC - encrypted Content Credits	36
TUNE tokens, TUNC tokens & Smart Contracts	37
Calculating TUNE token and credit Rewards	39
TUNE Credit “Mining” Protocol	40

TUNE Credit Reward Protocol	44
Case Study 1	54
Amateur Artist	54
Case Study 2	55
Playlist Queen	55
Case Study 3	55
Upcoming Producer	55
Token Economics and Distribution	57
Token Launch	57
Token Supply	58
Use of Funds	59
Timeline	60
Cre8tor.app (formerly RecordGram) Timeline	61
Tune Timeline	61
Team	62
Legal Disclaimer	67

Introduction & Vision

Tune Software is a software development and management company primarily focused on creating blockchain based technology, smart contract design and digital tokenized applications. Tune's primary mission is to solve digital songwriter rights and royalty transparency issues for the music industry. Furthermore, the establishment of our tokenized products will enable applications the ability to offer user tipping and user/community incentivized rewards through the sharing, creation and engagement of content, as well as other features. Tune is proud to announce a licensing agreement with the TechCrunch Disrupt Cup award winning Cre8tor.app (formerly RecordGram) application.

The existing Cre8tor.app (formerly RecordGram) community will act as the catalyst in the application and practical utility of the TUNE token to demonstrate, build and grow community.

In a day and age where access to resources, material or otherwise, is becoming increasingly egalitarian; we see access to curated music content and appropriate compensation, thereafter and through, as no different. We are trying to create the first international songwriter database for new music creations by redefining the future of music rights and royalty transparency on the Blockchain. Cre8tor.app (formerly RecordGram) is revolutionizing the way music content is distributed and made available. Tune wants to apply that same methodology and intertwine it with the power and efficiency of cryptography, decentralization and immutability.

Tune seeks the disrupt the current music industry in several ways:

1. The introduction of a token called TUNE which will provide a monetized model as a means of exchange for stakeholders and participants in the ecosystem including musicians, producers, and others

to derive new revenue streams. TUNE token will power transactions such as sale and lease of beats, artist rewards, tipping artists by fans, purchase of advertising and more.

2. The tokenized application of a blockchain enabled platform will create a semi-decentralized economy of musicians, producers, and followers, utilizing the open-source nature of the community to embrace this disruptive technology.

3. Cre8tor.app (formerly RecordGram) will establish itself as a decentralized mobile application that democratizes music collaboration and talent discovery, while providing those users who generate value on the platform an opportunity to be compensated for their involvement through the integration of Tune Software's Blockchain technology and the acceptance of cryptocurrencies.

The Tune protocol aligns the interests of all stakeholders in the media consumption cycle. By creating a new medium of exchange, Consumers get more choice in how they pay for music; Creators and Curators get a new form of compensation and Advertisers get a more transparent accounting and audience information.

The \$2.5B Problem

It is estimated that any given point, there is \$2.5B in uncollected music royalties. This astronomical number manifests in two ways:

1. The historic and perpetual lack of transparency in the music industry
2. The lack of an international database for songwriter rights

New artists simply don't have the time, the wherewithal or the legal understanding to enter into mutually beneficial and holistically sound contractual agreements with other stakeholders in the industry. Major label signed artists are largely beholden to the arbitrary whims of major label companies that dictate how and when an artist will be compensated and even with whom the artist can engage.

Furthermore, there are millions of aspiring artists around the world that have a burning desire to pursue a career in music. However, they either don't have access to quality music producers or they can't afford to pursue a career in music.

Additionally, the cost of renting or even owning a studio is beyond range for most people and learning how to use the equipment and technology necessary to properly produce and / or record on a beat is a complicated process to master, heightening the barrier to entry even further.

Solution

Serving as an integration platform, Tune will be the decentralized data and contract mechanism that will serve as an immutable ledger and integration hub to store all relevant data regarding contractual agreements between artists, producer and now fans.

The objective of having a platform that houses all songwriter contracts is two-fold:

1. It grants full transparency to the industry, in that any party can access and view the contractual relationship between them
2. Payments to rights holders becomes equally as transparent because each party is now bound to a smart contract that automates the process of payments based on certain sets of pre-agreed upon conditionals, smart contract logic and compensation algorithms (see *Calculating TUNE token and credit Rewards*)

Launch: Current Integration

Tune is an evolving ecosystem that is applying current technology and leveraging the innovation of what it can offer. In its current manifestation, the platform will integrate Cre8tor.app (formerly RecordGram) application as its first use-case with several on and off-chain solutions in an effort to accommodate shifting demands both internally and within the market. The TUNE token will be used as a method of exchange to all stakeholders, initially within Cre8tor.app (formerly RecordGram), and in the future with other integrated applications to further grow and add value to the community.

Scale: Protocol and Utility

The next immediate phase of our technical guide is to further enhance our Tune protocol, develop a framework and expand on the utility of the token to further democratize the music landscape. External integration is key and priority to incentivize all participants of the market to take stake in the Tune ecosystem, drive its adoption and value generation.

Community contribution will be assessed using an in-house developed algorithm in conjunction with existing blockchain technology that will log in each transaction with smart contract integration that will distribute TUNE tokens, credits and log contribution data after meeting a certain set of predefined conditions.

Content Storage

We plan to utilize Filecoin for decentralized content storage. For the purposes of our first use-case, the content will be stored centrally until we can further test on other networks such as IPFS to determine its viability and true scalability. For our initial technical assessment, we have drawn out an integration with IPFS (please see *Decentralized Content*). Though we see the value in IPFS and have

successfully tested on it and see and crave its utility, it is not robust enough for us to say with confidence that it is the solution of choice for streamlined decentralized content storage.

Filecoin serves as a good alternative because it is a tokenized platform built on top of the peer-to-peer distributed file system, IPFS. Being able to utilize a decentralized storage network (DSN), maintained by the interoperable IPFS protocol and sustained by its token-driven marketplace, is ideal for a platform like Tune and for most platforms that are leveraging decentralized content storage. It is ideal in that multiple tokenized platforms will be interacting with one another further solidifying the decentralized controlled new private economies.

Transaction Scalability

TUNE tokens will be implemented on the public Ethereum blockchain as an ERC20 token. Ethereum has become the industry standard for custom digital token and asset creation and smart contract integration. The Ethereum network is built for the deployment of both decentralized applications (dApps) that function to drive cryptocurrency and digital asset management and market generation, as well as smart contracts that define and execute terms and conditions through, for and with, said tokens. It's custom compatibility with existing infrastructure makes it an ideal platform for the deployment of digital tokens.

We recognize the limitations of the Ethereum Network in terms of transaction speed and costs. We plan to leverage off-chain solutions (state-channels) such as Raiden Network for micro-payments in alternate payment channels to alleviate the stress from the main-chain resulting in decreased fees and smoother transactions. Raiden is the Ethereum version of the Bitcoin Lightning Network, enabling scalable, low-fee and near-instant transactability and privacy protection. Off-chain payment channels allow for near-unlimited transfer between parties in a closed channel that is only committed to the main-chain at the end of a transaction session. Raiden's balance proof is a binding agreement enforced by the Ethereum blockchain with digital signatures verifying the integrity of the participants.

Furthermore, Tune will utilize the power of Hyperledger Fabric. Hyperledger is an open sourced project sponsored and supported by a wide array of technology and business organizations, created to

advance cross-industry Blockchain technology. It features a modular architecture that seeks to adapt to different industries and their diverging requirements in terms of scalability, confidentiality, workflow complexity, compliance, speed and security. Hyperledger takes a novel approach to the original Blockchain concept, by managing the admission of participants at its core: in other words, it establishes itself as a permissioned shared ledger. With its evolved concept, Hyperledger allows for true scalability, providing a solution to a multitude of industrial use case requirements. It offers a robust model for identity, auditability and privacy.

Tune Overview

Tune Software will create a decentralized community of stakeholders in the music ecosystem, seeking to disrupt the status quo and revolutionize the way music rights are stored, verified and royalties distributed.

However, some elements of the platform will remain centralized until other decentralized options become feasible or desirable. Tune will collaborate in their strategic partnerships for allocating rewards to platforms, dApps and apps on the ecosystem, creating developer extensions that provide visibility into the use of the apps within the ecosystem (such as reporting on and visualization of activity), propagating the Tune Ecosystem to developers and content partners, bringing promising and diverse platforms and apps into the ecosystem and more.

We plan to decentralize as much of the platform as possible as the technology becomes more available and as we continue testing further.

Economic Incentives

Cre8tor.app (formerly RecordGram)m is the first application to utilize this platform and give its users a way to participate in this new ecosystem driving content creation, user engagement and advertiser purchasing power. This, holistically, results in network growth and value added for the community where all participants are fairly rewarded for their contribution.

This is achieved by first launching a cryptocurrency called TUNE. Contributors are rewarded with tokens that unlock functionality, value and status within the community (see more in *Tune Utility and Protocol Dynamics*). As a decentralized and transferable ERC20 token with fixed supply, the Tune Ecosystem is set to grow in proportion to TUNE's utilization by users.

We then created Tune Credits, which will be the internal method of consumption and compensation within the Tune ecosystem. Tune Credits are how every stakeholder will be compensated (artists, producers, fans, curators) for the entire spectrum of engagement (see more in *Tune Utility and Protocol Dynamics*).

User Onboarding

Developing a frictionless transaction process and wallet set-up is crucial for user onboarding because, for many, this will be the first time they interact with cryptocurrencies. The design is to be seamless and will allow flawless entrance into the Tune ecosystem, irrespective of experience or even knowledge of cryptocurrencies.

The fact that most of our users will not have had any experience with cryptocurrencies, tokens or digital assets, represents an opportunity to introduce a whole user base to the world of decentralization and cryptocurrency.

The first interaction a user must have when interfacing with cryptocurrencies or tokenized platforms is the wallet setup. Taking a user-first approach, we focused on making that process as frictionless as possible. User wallets are created the moment a user signs up for Cre8tor.app (formerly RecordGram), making onboarding no different from any other app. The wallet will be fully modular, in that users will also have the option to take their tokens off-device and hold them in whatever storage option they deem fit. We will also introduce API integration, minimizing the burden on the user to understand and independently interface with multiple different platforms.

Token Utility & Protocol Dynamics

Protocol

The primary objective of the protocol is to evolve based on market demands and adoption in conjunction with community feedback and, therefore, will be rolled out incrementally. Tune's protocol was designed to fulfill three main objectives in the interplay between consumption and contribution:

1. Verify the participant interacting with the smart contract
2. Calculate contribution to the ecosystem
3. Calculate compensation based on interaction

Tune Credits and Network Tokens

One of the primary objectives, beyond developing a music rights database and transparent royalty accounting model on the blockchain, is the community engagement that will define the utility of TUNE within the Tune ecosystem. That is why we have developed a compensation system that will incentive all parties to participate in the network by having stakeholder compensation commensurate to their contribution in the community. This will be tracked and executed by integrating Ethereum with Hyperledger's Fabric.

The Tune Community will be comprised of three types of tokens:

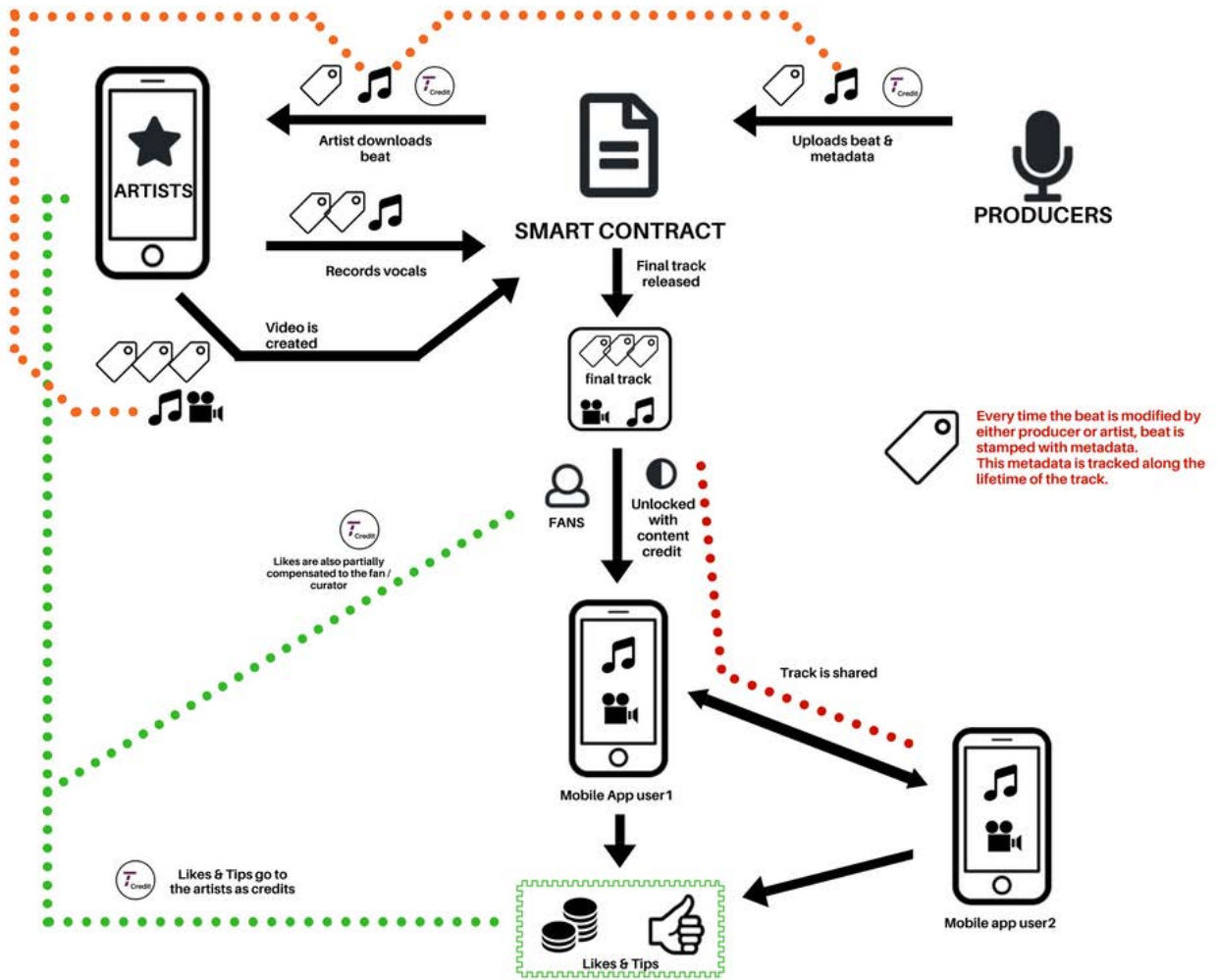
- TUNE Token - TUNE (fungible)
- TUNE Credits - TUNC (fungible)

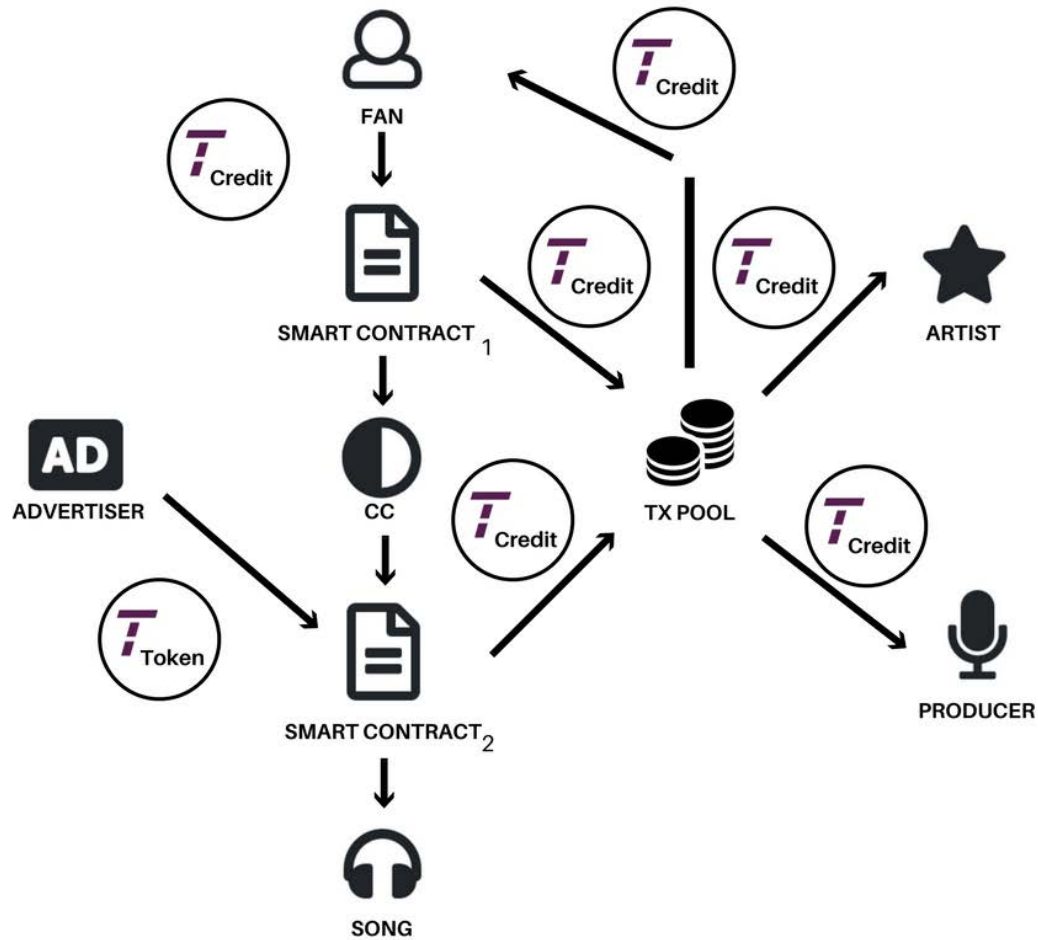
- Content Credits - CC (non-fungible)
 - CC - standard digital asset
 - eCC - encrypted-Content Credit (required to access content)

TUNE will be freely exchanged on the Tune platform for Tune Credits (TUNC). TUNC's functionality is as a medium between the artists and fans in content curation, consumption and between artists and producers in content collaboration, royalty and music right compensation.

Stakeholders

- **Artist** - this is the creative talent who authors the music or lyrics
- **Producer** - this is an individual who finances a project and leases or sells beats
- **Fan** - this is the person who downloads the app and uses it to consume content
- **Advertiser** - this is an entity that advertises its product(s) or service(s) on the Cre8tor.app (formerly RecordGram) network
- **Curator** - this category is not mutually exclusive and is a person who curates content either passively (liking playlists) or active (building a follower base based on curated content)
- **Follower** - this is a person who follows another person in the platform (could be an artist, fan, curator or producer)





A fan, in this case, interacts with a smart contract (1) through the utilization of Tune Credits (TUNC) and in return gets non-fungible Content Credits. The fan then engages with another smart contract (2) specific for the content being access. Each eCC is burned for the part of the song / video being listened to / watched. The advertiser is already engaged with the second smart contract for the content requested with preset distribution metrics for the other stakeholders (in this example the artist and producer). Advertisement is optional within Tune, depending on the subscription model of the stakeholder (see *Payment Service*).

-It is important to remember that the interactions with smart contracts is done in the backend and the user will only interface with the app.

Technology Stack

We have split our tech stack in two phases:

1. The immediate implementation of our platform with the available tested technology at hand.
2. Our projected application of what we will leverage after our token sale through research and development.

For the purposes of Token protocols and technological specifications, we are defaulting to the immediate application phase.

Immediate Implementation

Hyperledger

We believe Tune will have a perpetually high volume of microtransactions. This is problematic for a platform exclusively using Ethereum due to high gas costs (transactions fees) in the Ethereum Network. Thus, we will be using Ethereum judiciously for financial transactions and Hyperledger Fabric for micropayments and smart contract logic.

- **Fabric**

Tune has chosen Hyperledger Fabric for the following reasons:

1. Some transactions can be stored on Hyperledger to avoid incurring transaction fees on the Ethereum network.

2. It provides privileged access to users allowing them the ability to store and share some of the data privately on Blockchain. In contrast with other platforms, Hyperledger Fabric is a permissioned Blockchain, which means that information stored in blocks is not available to the public, but only to their participant nodes. This allows data storage on the Blockchain with the protection of privacy and personal information.

3. Hyperledger Fabric is very customizable, allowing more functionality to be implemented in further versions of Cre8tor.app (formerly RecordGram). Its modular architecture has been intentionally developed with the aim of making it compatible with differing industry needs. Besides this, Hyperledger Fabric can be seamlessly adapted to the evolving requirements of an online music platform as it grows in number of users and strategic partners.

- **Composer**

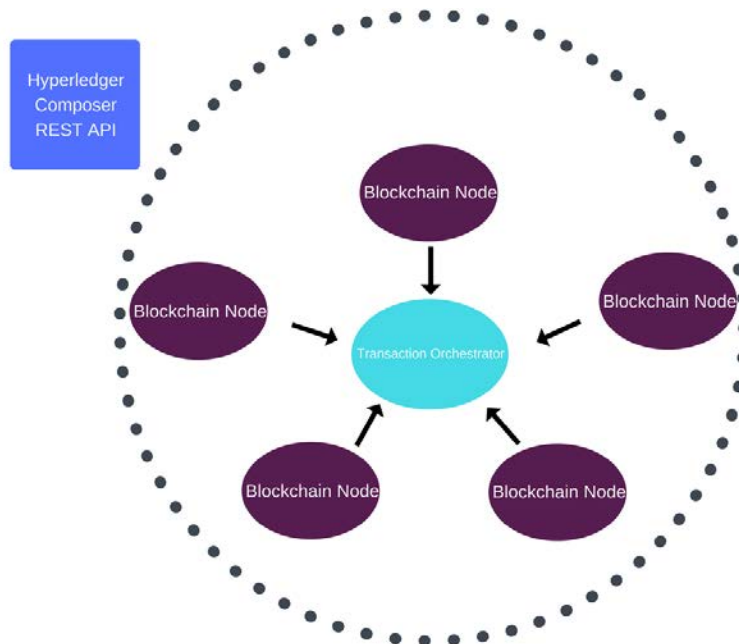
Hyperledger Composer is a toolset and framework designed to build and run applications on top of Hyperledger Fabric.

Hyperledger Composer supports the existing Hyperledger Fabric blockchain infrastructure and runtime, which supports pluggable blockchain consensus protocols to ensure that transactions are validated according to policy by the designated business network participants. Fabric is effectively the network-level framework, and Composer is the application-level build.

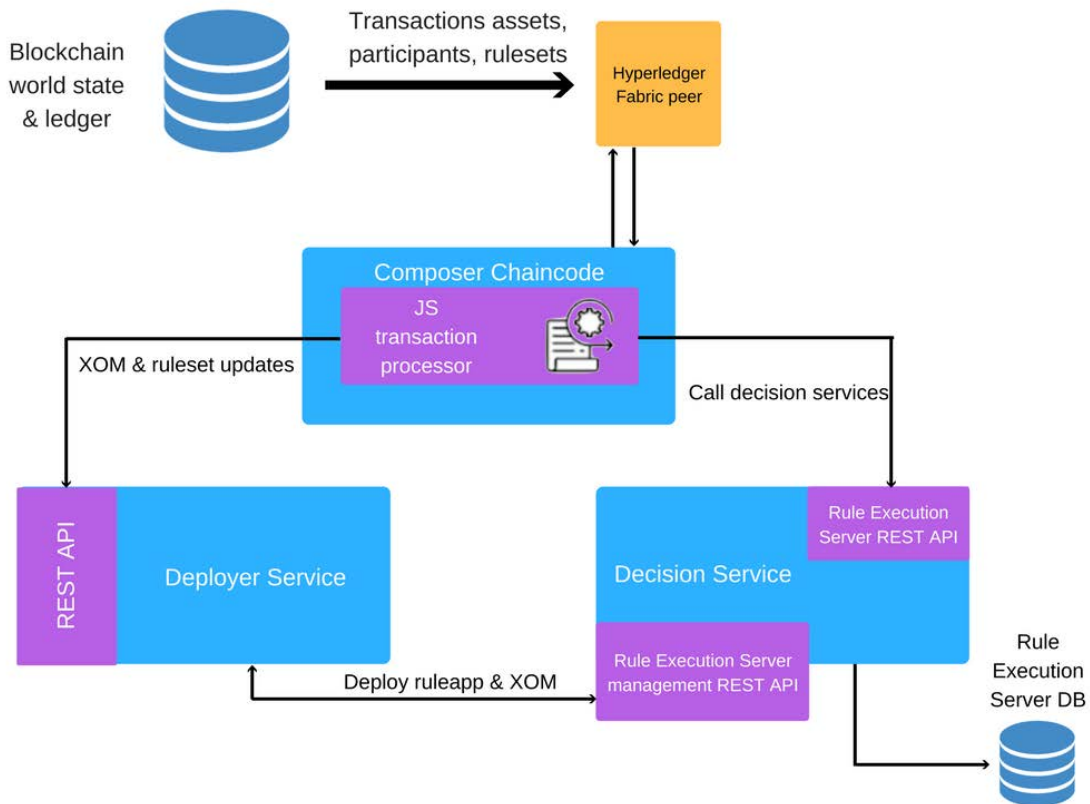
We believe the main benefits of Hyperledger Composer are:

1. The ability to define assets that are exchanged in a blockchain-based use case
2. Define the business rules around what transactions are possible

3. Define participants, identity and access controls for what roles exist and which roles can execute which types of transactions
4. The ability to model custom, reusable, core components in a business network
 - a. assets, participants, transaction logic, and access controls for the business network, which can then be shared across multiple organizations within a given network
5. Generate JavaScript and REST APIs based on the business network definition that can be used to interact with applications, integrate legacy systems, create skeleton applications and run analytics on the blockchain
 - a. Users of such applications don't have to run a local node and can interact with a remote node through an RPC or HTTP REST if needed



Each node has a copy of the transaction ledger and assets stored in the database, called *world state*. The Hyperledger Composer includes a standalone Node.js process that exposes a network as a REST API. The Transaction Orchestrator houses the ACL (Application Control List) and all relevant permissions of the network.



Each Composer application in the business network is represented by a chain code process with a JS interpreter that executes logic for processing the transaction. Each blockchain node is deployed with Docker Compose. Compose allows you to define and run multiple Docker applications

```

87
88 abstract transaction BeatTransaction identified by beatID {
89   o String beatID
90   --> Beat beat
91 }
92
93 transaction PrivateBeatTransfer extends BeatTransaction {
94   --> Person seller
95   --> Person buyer
96   o String specialNotes optional
97 }

```

The `PrivatBeatTransfer` class represents the transaction in Hyperledger Composer

```

{
  "$class": "org.acme.beat.lifecycle.decision.TransferDecisionService",
  "$id":
"resource:org.acme.beat.lifecycle.decision.TransferDecisionService#isTransfe
r",
  "dsId": "Transfer",
  "transaction": {
    "$class": "org.vda.PrivateBeatTransfer",

```

```
"$id": "resource:org.vda.PrivateBeatTransfer#4302c409-96f1-4660-9772-400875e5e2e2",
"seller": {
  "$class": "composer.base.Person",
  "$id": "resource:composer.base.Person#blackout",
  "ssn": "blackout",
  "firstName": "Winston",
  "lastName": "Thomas",
  "gender": "MALE",
  "nationalities": [
    "USA"
  ],
  "contactDetails": {
    "$class": "composer.base.ContactDetails",
    "email": "blackout@acme.org",
    "address": {
      "$class": "composer.base.Address",
      "city": "Miami",
      "country": "USA",
      "region": "North America"
    }
  }
},
"buyer": {
  "$class": "composer.base.Person",
  "$id": "resource:composer.base.Person#tina",
  "ssn": "tina",
  "firstName": "Tina",
  "lastName": "Smith",
  "gender": "FEMALE",
  "nationalities": [
    "French"
  ],
  "contactDetails": {
    "$class": "composer.base.ContactDetails",
    "email": "tina@acme.org",
    "address": {
      "$class": "composer.base.Address",
      "city": "Paris",
```

```

        "country": "France",
        "region": "Europe"
    }
},
"specialNotes": "DJ Blackout selling a beat to Tina Smith",
"beat": {
    "$class": "org.vda.beat",
    "$id": "resource:org.vda.beat#156478954",
    "vin": "156478954",
    "beatDetails": {
        "$class": "org.vda.beatDetails",
        "genre": "hiphop"
        "ISRC": "90827345983242344"
        "VST": "Nexus"
        "DAW": "ProTools"
        "PRO": "ASCAP"
        "beatDuration": "120"
        "ownershipPercentage": "0.70"
    },
    "beatStatus": "ACTIVE",
    "owner": "resource:composer.base.Person#blackout",
    "logEntries": [
        {
            "$class": "org.vda.beatTransferLogEntry",
            "beat": "resource:org.vda.beat#156478954",
            "buyer": "resource:composer.base.Person#tina",
            "timestamp": "2018-07-30T13:57:13.652Z"
        }
    ]
},
"transactionId": "4302c409-96f1-4660-9772-400875e5e2e2",
"timestamp": "2018-07-30T13:57:23.121Z"
}
}

```

As the `PrivateBeatTransfer` transaction points to the buyer, the seller and the beat, the entire object is serialized as a JSON response. It is sent to the decision service when calling the Hyperledger Composer `post()` API.

Ethereum

Payment transactions performed on Cre8tor.app (formerly RecordGram) will be handled by the Ethereum main net (network id 1). Ethereum is a platform for decentralized applications with the Ethereum Virtual Machine (EVM) at its heart. EVM executes smart contract logic through its Turing-complete scripting language, Solidity, and stores the resulting data in the logs. EVM also maintains consensus for Blockchain. A smart contract can be invoked by receiving a transaction to its address. Transactions changing the state of a smart contract (storing data in smart contracts) require gas fee in Ether to cover mining process expenses. There are two primary protocols that define Ethereum tokens at the moment: fungible ERC-20 (Ethereum Request for Comments) and non-fungible digital asset tokens, ERC-721.

TUNE token will follow the ERC20 standard so that it can be easily implemented into other services. ERC20 is a set of rules that defines the parameters of the tokens. In general, having a common set of rules is beneficial for the adoption of tokens and for the future of Blockchain for the purposes of uniformity in token usage and generation.

ERC-20

- **Preamble**

```
ERC: 20
Title: Token standard
Status: Draft
Type: Informational
Created: 19-11.2015
Resolution: https://github.com/ethereum/wiki/wiki/Standardized\_Contract\_APIs
```

- **Methods**

- **name**

```
function name() constant returns (string name)
```

- **symbol**

```
function symbol() constant returns (string symbol)
```

- **decimals**

```
function decimals() constant returns (uint8 decimals)
```

Returns the number of decimals the token uses - e.g. **8**, means to divide the token amount by **100000000** to get its user representation.

- **totalSupply**

```
function totalSupply() constant returns (uint256 totalSupply)
```

Returns the total token supply.

- **balanceOf**

```
function balanceOf(address _owner) constant returns (uint256 balance)
```

Returns the account balance of another account with address `_owner`.

- **transfer**

```
function transfer(address _to, uint256 _value) returns (bool success)
```

Transfers `_value` amount of tokens to address `_to`, and must fire the `Transfer` event. The function should **throw** if the `_from` account balance does not have enough tokens to spend.

- **transferFrom**

```
function transferFrom(address _from, address _to, uint256 _value) returns (bool success)
```

Transfers `_value` amount of tokens from address `_from` to address `_to`, and must fire the `Transfer` event.

The `transferFrom` method is used for a withdraw workflow, allowing contracts to transfer tokens on your behalf. This can be used for example to allow a contract to transfer tokens on your behalf and/or to charge fees in sub-currencies. The function should **throw** unless the `_from` account has deliberately authorized the sender of the message via some mechanism.

- **approve**

```
function approve(address _spender, uint256 _value) returns (bool success)
```

Allows `_spender` to withdraw from your account multiple times, up to the `_value` amount. If this function is called again it overwrites the current allowance with `_value`.

- **allowance**

```
function allowance(address _owner, address _spender) constant returns (uint256 remaining)
```

Returns the amount which `_spender` is still allowed to withdraw from `_owner`.

- **Events**

- **Transfer**

```
event Transfer(address indexed _from, address indexed _to, uint256 _value)
```

Must trigger when tokens are transferred, including zero value transfers.

A token contract which creates new tokens should trigger a Transfer event with the `_from` address set to `0x0` when tokens are created.

- **Approval**

```
event Approval(address indexed _owner, address indexed _spender, uint256 _value)
```

Must trigger on any successful call to `approve(address _spender, uint256 _value)`.

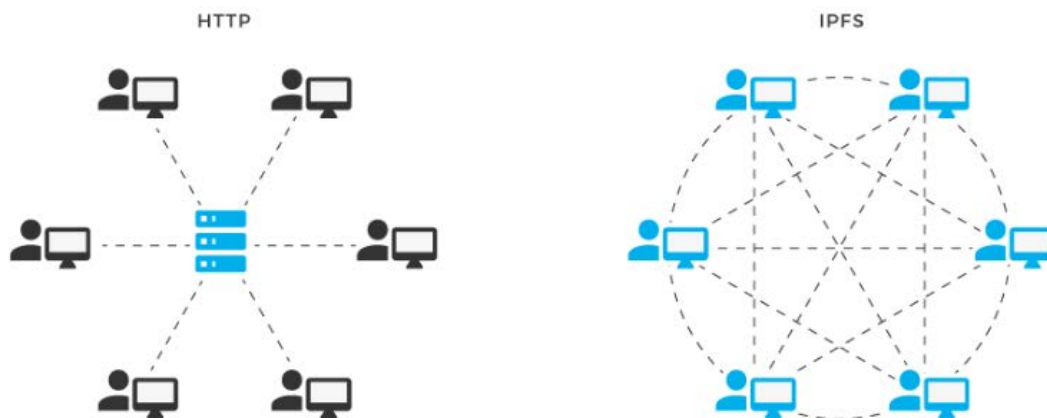
Long-Term R&D

IPFS

InterPlanetary File System (IPFS) is a peer-to-peer network connecting remote servers in a single global decentralized storage platform. A significant amount of storage capacity will be required to store all the compositions in a decentralized manner and this can be achieved through IPFS.

Tune is planning to utilize customized Clustered IPFS Swarm for the platform. Clustered IPFS nodes should create the necessary storage capacity for storing beats, compositions and videos, while providing faster access to them. Files stored in IPFS are divided into smaller chunks and each chunk is encrypted with keys. Based on our experience, storing files as chunks can significantly reduce download time and make them more secure.

The Tune community will be involved in the storing and streaming of compositions and reward compensation in return. An additional layer is added on top of Clustered IPFS, which will provide a Proof-of-Streaming (see below) reward system for the users who run IPFS nodes, store and stream files.



At the IPFS protocol level, the system is fully neutral and the nodes can run over any transport protocol. IPFS nodes can run through different network architectures such as NDN (Named Data Networking), XIA (expressive Internet Architecture) and CCN (Content-Centric Networking). IPFS nodes do not have to be referenced by a centralized IP. This transport-neutral network protocol will ensure that IPFS can be transmitted through any network and resist censorship.

This implementation also introduces other participants to the ecosystem including IPFS Node Operators, Content Hosters and Content Streamers.

Node Operators: A person who operates a Full node in the decentralized network

Node Hosters: A person who operates a Node that hosts content chunks on the network

Node Streamer: A person who operates a Node that streams audio and video from a Hoster to the end user

Proof-of-Streaming

Proof-of-Streaming is a layer on top of IPFS, extending the standard features of decentralized file storage. The central part of IPFS is the Merkle Directed Acyclic Graph (Merkle DAG). Besides hashed links, the Merkle DAG also contains certain properties for stored files. Public key infrastructure identity (PKI identity) is used in every IPFS network for the purposes of identifying nodes. IPFS bitswap is the data trading module containing a messaging protocol. Messages include want-lists or blocks. Bitswap manages requests of sending blocks between peers. It locates the right blocks, and sends them to other peers down the chain culminating at the user requesting the file.

Merkle DAG, PKI identity and bitswap are key components of the Proof-of-Streaming layer. A Hoster is a peer which hosts the first chunk in a chain. Nodes which stream chunks to the user will be considered as Streamers. Hosters and Streamers are required to broadcast Proof- of-Streaming (PoS) to the network. When consensus is reached and PoS is deemed valid, that stream will be added to the next block. Hoster and Streamer will be rewarded with Tune credits (TUNC). By the end of a streaming session, TUNC can be exchanged to TUNE tokens, based on the exchange rate for that session. TUNE will be provided by the token pool and PoS will be managed by the Hyperledger blockchain.

Content Security

We are testing with other security measures that will serve as initiatives in the growth of our Proof-of-Streaming concept.

Trusted Timestamping

Trusted timestamping is a process of proving that certain data existed at a certain time. Trusted timestamping is explained by RFC 361 Time-Stamp Protocol (Adams et al., Aug., 2001) and the ANSI ASC X9.95-2016 standard (American National Standards Institute), which expands RFC 361 with data-level security requirements. These requirements make a standard for trusted time stamping authority (TSA) to issue and validate timestamps.

Content Identification Marking

In this section, we compare two common methods of Content Identification techniques that may be employed in the Tune system. There are a few ways to mark music files. First, by steganography and second, by tracking fingerprint.

Steganography

Steganography is the science of hiding information in plain sight; hiding a secret message within a regular message and only revealing / accessing the information once at destination. The information is hidden, as opposed to being unreadable. Steganography encrypts certain data into the audio file and makes it impossible to extract, while allowing the original content to be accessed and consumed.

Advantages:

- Sound will contain unique ID
- Difficult to remove embedded data (depending on the algorithm)

Disadvantages:

- Noise can appear on some frequencies (depending on the algorithm)

Fingerprint

Each track has a unique fingerprint, which can be extracted by analyzing the spectrum of the track.

Advantages:

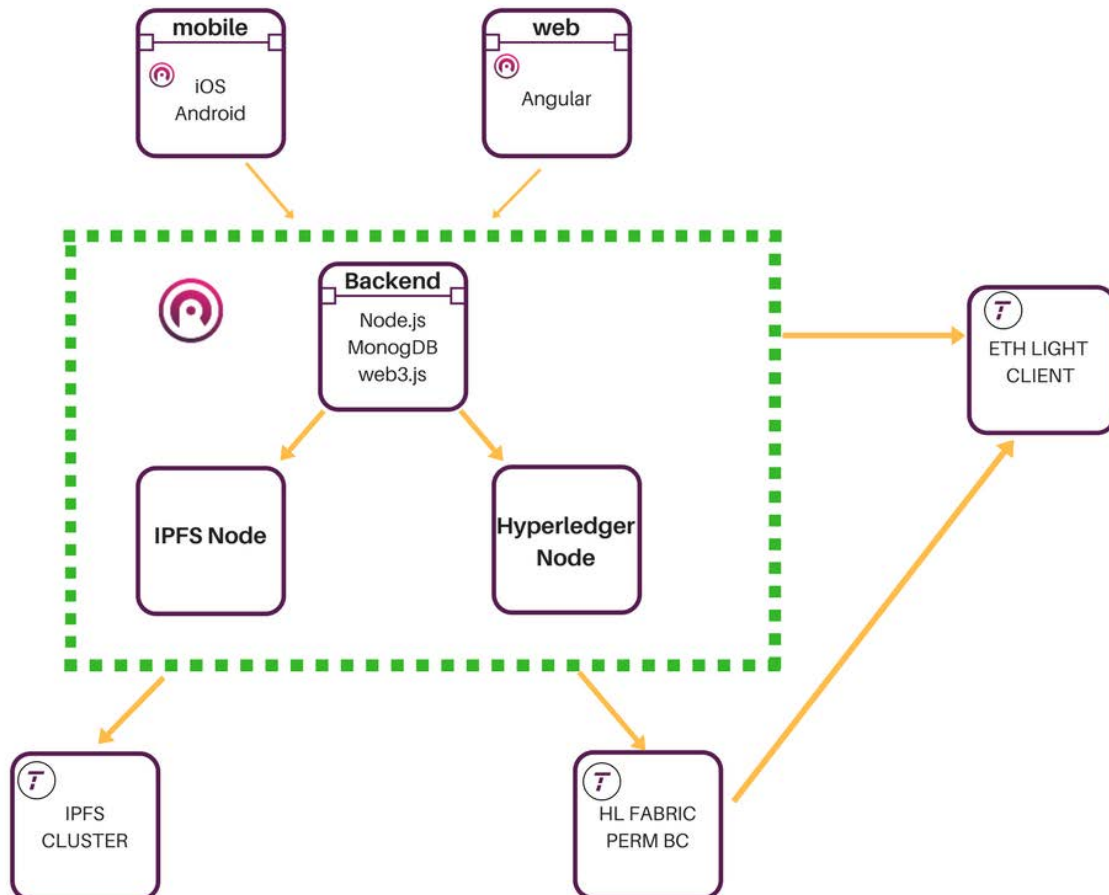
- Fingerprints are imperceptible to human ears, and do not alter the sound of the track
- Fingerprints can be potentially stored on blockchain for identification purposes
- Multiple algorithms can be applied simultaneously, since it makes no impact on the quality

Disadvantages:

- Fingerprints are not embedded into the music file Streaming Service.

DApp Architecture

For DApp architecture we plan to use backend (Node.js), MongoDB, Hyperledger Full Node. Node.js runs with Docker Swarm, allowing the project to scale easily and dynamically. MongoDB clusterization is handled by MongoDB itself when configured in production mode. Hyperledger Full Node will be connected to a private Blockchain network, with a predefined default channel. Each peer will be connected to Fabric's Certificate Authority to handle user creation and authentication and to grant and revoke privileges.



The diagram above shows the architectural relationship between Cre8tor.app (formerly RecordGram)m and Tune. This same relationship applies to partner applications that become a part of the Tune platform.

Angular will be used for the web app and web3.js will used to interact with a local or remote ethereum node, using a HTTP or IPC connection.

All applications that manage large volumes of data are often challenged with cost, capacity, bandwidth and latency issues. We believe implementing IPFS would be a strong competitive advantage because it allows for the minimization of operational costs, and makes the service more affordable to the end consumer.

Tune will incentivize the community when they participate in storing and streaming files, decreasing the overhead cost for storing fees. Through this decentralized system there won't be a single computer or server that holds all the data as it will be spread across multiple nodes.

IPFS solves several problems:

1. Eliminates the need of having a central server for content hosting and streaming
2. Download speed is not limited by the remote server bandwidth channel
3. Simultaneous download from multiple nodes
4. Solve content-addressing and content-signing security issues
5. DDOS attacks are inefficient

Payment Service

All applications and other platform integrations must accept and utilize TUNE tokens. Producers uploading beats will have the ability to sell or lease them for a certain price in Tune credits to an unlimited number of users. The price to unlock a beat and the rights associated with it will be executed through smart contract logic that will ensure that all producers, regardless of their professional

accomplishments, will be treated equally. This price can be fixed in at currency, but transaction clearing shall be done in TUNC.

We will be using Cre8tor.app (formerly RecordGram) as the first use-case.

There will be 2 ways of using TUNE in the Cre8tor.app (formerly RecordGram) app:

1. Deposit to personal Cre8tor.app (formerly RecordGram) account for further usage. This option may be convenient for recurring subscription payments.
2. Direct payment on Blockchain. This option is good for one-time payment and/or tipping an artist.

The different tokens have different use-cases in interplay:

- TUNE Token - TUNE (fungible)
 - **Subscription** - Producers will be able to upload and sell/lease their beats by subscribing to Cre8tor.app (formerly RecordGram). A limitation will exist on the number of uploads for each producer based on the subscription package and there will be options available to expand said package, defined by the app and accepted by the community.
 - **Artist rewards** - The most popular artists, as defined by the Tune community, can be rewarded in Tune Tokens.
 - **Buying ad space** - Advertisers will be able to use tokens to buy ad space on the Cre8tor.app (formerly RecordGram) platform, and contributing users should be able to also buy ads to boost their social media post or to highlight a record they're promoting.

- Advertisement is part of the Tune ecosystem, but its consumption is not a requirement for content consumption, based on the subscription model of the stakeholder
- TUNE Credits - TUNC (fungible)
 - **User rewards** - Rewarding regular users for viewing ads or sharing content (through TUNC).
 - **Tipping artists** - Users will be able to support artists they like by sending them tokens.
 - **Entering into agreements** - TUNC will drive the smart contract agreements between stakeholders and between all parties over content and engagement
 - RG will keep 20% Tune Credits when an Artists unlocks a beat with the other 80% distributed to the Producer's as a royalty
 - Any credits earned after will be split 50/50 between Artist and Producer
- Content Credits - CC & eCC (non-fungible)
 - **Unlocking content** - Users will be able to make payments in TUNC to unlock beats, music or video that they wish to use in their composition or consumption by using eCC's in the backend as part of their content purchase
 - **Unlocking content** - Users will be able to make payments in TUNC to gain access to non-fungible digital assets (CC) released by their favorite artist or producers for promo purposes or otherwise

Tune will be offering a wallet airdrop service for existing users of Cre8tor.app (formerly RecordGram) as early adopters. These early adopters will be able to claim their tokens and begin utilizing the platform immediately.

Technical Assessment of Tokens (Cre8tor.app (formerly RecordGram) -> Tune)

Content Credits: eCC - encrypted Content Credits

Tune is trying solve a major problem within the entertainment industry, specifically with respect to musicians and original content providers. Piracy has beleaguered the music industry since its inception. The difficulty in securely transferring content across platforms, or even internally without fear of theft and copyright infringement has become overwhelming and must be resolved. We propose one such solution that is encrypted, unique non-fungible tokens called eCC's (encrypted-Content Credits) that will give participants in the platform access to content that is only accessible for that user, for that song, for that session (in the event someone is streaming a song). This is made possible through Fabric's smart contracts that will process each microtransaction and each interaction as a separate smart contract, delivering eCC's to the user.

This serves several benefits:

1. It more securely grants access to participants specific to the content they are requesting to access
2. It allows for better compensation through smart contract microtransaction logic
3. It allows for better tracking analytics in a decentralized protocol allowing all stakeholders to view and verify the integrity of usage and interaction.

We are solving several problems with our non-fungible token protocol within Fabric, in that users will utilize several sub tokens (CC) that will represent a multitude of factors within the interchange, sharing and promotion of content within the platform. More specifically, the creation of **encrypted-digital assets credits** (eCC) represent segments within the given content being streamed or purchased. A song, for instance, that is 1:30 seconds long has 3 different credits that represent the sequential 30-second segments of the song. Each credit is encrypted and synchronously used to unlock the next segment of the song. The logic for credit generation and definition will work in conjunction with the algorithms defined below (see *Calculating TUNE tokens and credit Rewards*). A user must have access to all the credits of the content they are requesting access to in order to consume the content.

The credits are automatically generated when a user interacts with a function that calls the smart contract, respective to their given request. When content is being used, the session interacts with a smart contract that was automatically generated specific for that session with that user that will only trigger functions within the contract when the tokens are transferred to the contract on the backend.

This integration will be seamless and the user will not even be aware of the complexity running in the background as they listen to their new favorite song, or lease out beats from a producer to compose through the app's user interface.



TUNE tokens, TUNC tokens & Smart Contracts

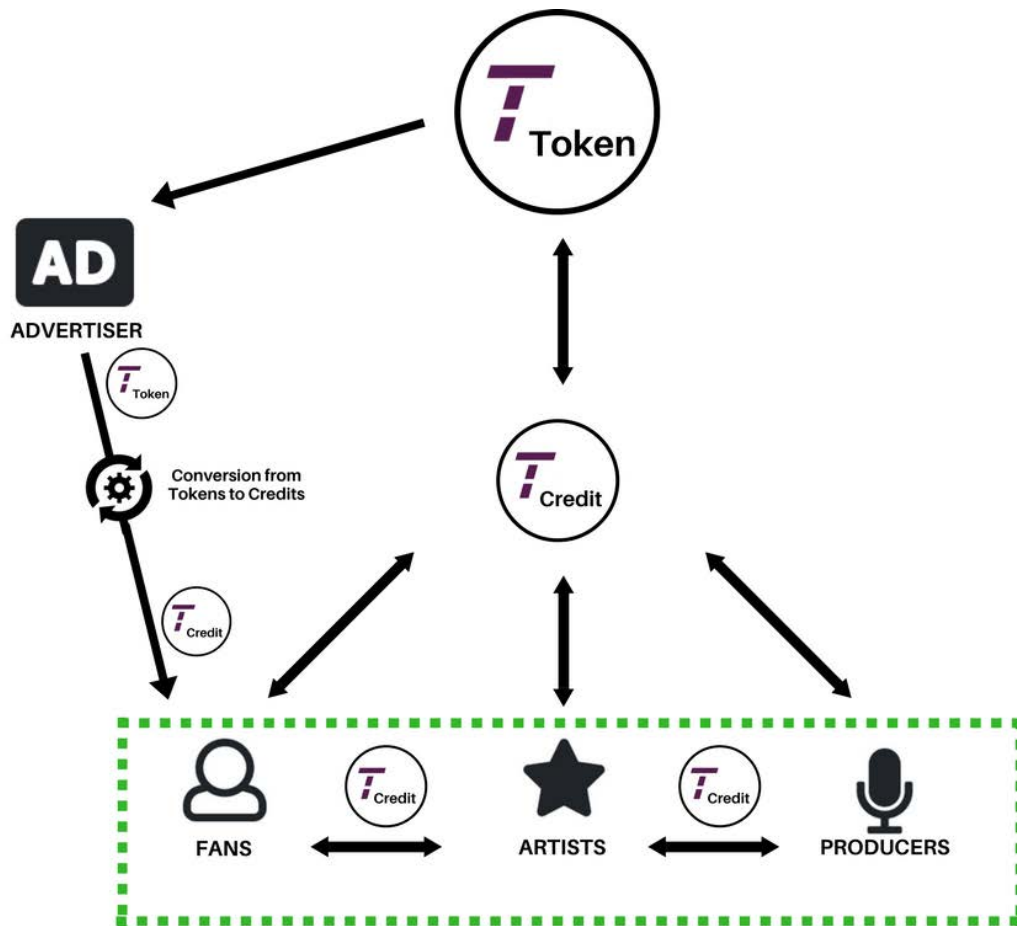
In addition to the eCC required to use / view / listen to a song / video / beat, the user is compensated in fungible Tune credits (TUNC) that can be used to make in-app purchases, listen to and curate more content. TUNC will have a fixed supply of 1 billion credits.

Artists and producers can similarly use TUNC in contractual relations with each other, in addition to other goods and services provided in the platform.

Advertisers further engage in the ecosystem by purchasing ad-time in TUNE tokens before content play or usage.

Fans can further tip artists in TUNC, meaning that the fungible TUNC will be the primary means of exchange within the interplay of content.

Redefining the music industry starts with making the integration of the Tune platform seamless, including entering into smart contracts with other stakeholders that can be requested, denied, modified and optimized for the participant at hand.



Tune tokens grant a participant access to the Tune ecosystem. Advertisers always contribute in TUNE tokens to the stakeholders via smart contracts and each participant is compensated appropriately in credits. Stakeholders engage with one another using Tune credits which they can freely trade back and forth with TUNE tokens.

Calculating TUNE token and credit Rewards

Below is a proposed version of the reward algorithms that mathematically determine the compensation for the participants and stakeholder's participation within the Tune ecosystem.

We have created a self-sustaining ecosystem that rewards all parties involved with the creation, curation, promotion and dissemination of the content created. The algorithms below will power the smart contract logic that rewards all participants, creating a network effect that scales growth and the user adoption of Tune.

TUNE Credit “Mining” Protocol

Stakeholders (these titles are not mutually exclusive. Users can fit into more than one of these categories--every user is a fan)

- Artists
- Producers
- Fans
- Curators

Finite number of TUNCs.

1 Billion

Some different ways to “mine” TUNC.

- Consumption of content.
- Sharing of content (either individual songs/videos and/or curated user playlists)
- Referral of content

Token Credit rewarded in one session C_r

$$C_r = \Gamma \times (C + S)$$

Γ = supply and demand function

C = value of content consumption

S = value of content sharing

Supply/Demand Function - Γ

Function is necessary so that TUNCs are always available to mine.

Mining of credits is divided into different sessions. Once a certain amount of credits has been mined a new session starts.

Parameters used to calculate Γ

$TU_i = 1,000,000,000$ TUNCs

This is the maximum number of TUNCs that will ever exist.

TU_m = The current amount of TUNCs that have been mined in session.

TU_f = The current amount of TUNCs that have not been mined.

$TU_m + TU_f = TU_i = 1 \text{ Billion TUNCs}$

TU_g = The amount of TUNCs that has been added back to TU_f at end of session

φ = Session Coefficient

This coefficient is used to control the value of Γ . It is recalculated each time a session ends.

Γ will change depending on all these factors. Value will decrease as TU_m increases

$$\Gamma = \frac{(TU_f - TU_m) + TU_g}{TU_i} \varphi$$

Value of content consumption, C

$$C = T * F_v$$

T = Consumption Session time value

F_v = Fan value coefficient

Content consumption for a session will be based on several factors

Length of consumption session, T

This will create a maximum value for C and will decay as time increases. This is so that C does not grow infinitely large if the session goes on for perpetuity.

Fan Value, F_v

Fan value coefficient is used to quantify the value of a certain fan. The more value a fan has the larger that C can be. F_v is a number between 0 and 1.

Example: If the fan clicks on the ads and interacts they will have a higher F_v .

Example: Curators with many followers will also have a higher F_v .

Value of Content Sharing, S

$$S = H * F_v * S_v$$

H = Traffic Generation value

F_v = Fan value coefficient

S_v = Share value coefficient

Traffic generating value, H

This will create a maximum value so that infinite shares will not generate an infinite value of H . Like T it will also decay as the number of shares increases.

Fan Value, F_v

Fan value coefficient is used to quantify the value of a certain fan. The more value a fan has the larger that S can be. F_v is a number between 0 and 1.

Example: If the fan clicks on the ads and interacts they will have a higher F_v .

Example: Curators with many followers will also have a higher F_v .

Share Value, S_v

Share value coefficient is used to quantify the value of the share. The more value a fan has the larger that S can be. S_v is a number between 0 and 1.

Example: If the share reaches users with a high F_v then S_v will be higher.

Fans “mine” TUNC by interacting in the app and a certain value of C_r is issued Distribution of mined TUNCs.

R_a =TUNC ratio given to artists

$$C_r \times R_a = \text{TUNCs gained by artist}$$

R_p =TUNC ratio given to producers

$$C_r \times R_p = \text{TUNCs gained by producer}$$

R_f =TUNC ratio given to fans

$$C_r \times R_f = \text{TUNCs gained by fans}$$

R_c =TUNC ratio given to curators

$$C_r \times R_c = \text{TUNCs gained by fans}$$

R_r =TUNC ratio for referrals

$$C_r \times R_r = \text{TUNCs gained through referrals}$$

R_t =TUNC ratio for to be put back into TU_f

$$C_r \times R_t = \text{TUNCs gained by } TU_f$$

$$R_a + R_p + R_f + R_c + R_r + R_t = 1$$

Referrals

$$R_r = x + y$$

x = ratio for person referring

y = ratio for person referred

x and y are of equal value, $x=y$

when x and y is equal to zero R_r is added to R_t

Premise:

Fan A joins TUNE without a referral.

Fan B is referred by Fan A.

Fan C is referred by Fan B.

Scenario 1: Fan A consumes and mines a certain amount of TUNCs.

Fan A Credit Ratio:

$$R_A = R_f + x + y$$

$$x = 0, y = 0$$

Scenario 2: Fan B consumes and mines a certain amount of TUNCs.

Fan A Credit Ratio:

$$R_A = x + y$$

$$x = \frac{R_r}{2}, y = 0$$

Fan B Credit Ratio:

$$R_B = R_f + x + y$$

$$x = 0, y = \frac{R_r}{2}$$

Scenario 3: Fan C consumes and mines a certain amount of TUNCs.

Fan A Credit Ratio:

$$R_A = 0$$

Fan B Credit Ratio:

$$R_B = x + y$$

$$x = \frac{R_r}{2}, y = 0$$

Fan C Credit Ratio:

$$R_C = R_f + x + y$$

$$x = 0, y = \frac{R_r}{2}$$

Scenario 4: Fans B, C consume and mine a certain amount of TUNCs

Fan A Credit Ratio:

$$R_A = x + y$$

$$x = \frac{R_r}{2}, y = 0$$

Fan B Credit Ratio:

$$R_B = R_f + x + y$$

$$x = \frac{R_r}{2}, y = \frac{R_r}{2}$$

Fan C Credit Ratio:

$$R_C = R_f + x + y$$

$$x = 0, y = \frac{R_r}{2}$$

TUNE Credit Reward Protocol

Subscription Fee

Stakeholders and Advertisers are charged a subscription fee for usage in TUNE Tokens.

USF = Universal Subscription Fee

$$USF = A + P + C + F + Ad$$

A = Subscription fee for Artists

P = Subscription fee for Producers

C = Subscription fee for Curators

F = Subscription fee for Fans

Ad = Subscription fee for Advertisers

At the end of certain amount of time (t), the value of USF in TUNCs is redistributed

$$USF = (X + Y + Z + V + W) \times USF$$

X = percentage of USF for Artists

Y = percentage of USF for Producers

Z = percentage of USF for Curators

V = percentage of USF for fans

W = percentage of USF for TUNE

$$X + Y + Z + V + W = 1$$

Artist redistribution of TUNCs

$$(X \times USF) = (\alpha_a + \beta_a)(X \times USF)$$

$$\alpha_a + \beta_a = 1$$

α_a = percentage of $(X \times USF)$ to be redistributed to artists based on content consumption
 β_a = percentage of $(X \times USF)$ to be redistributed to artists based on content sharing

Producer redistribution of TUNCs

$$(Y \times USF) = (\alpha_p + \beta_p)(Y \times USF)$$

$$\alpha_p + \beta_p = 1$$

α_p = percentage of $(Y \times USF)$ to be redistributed to artists based on content consumption
 β_p = percentage of $(Y \times USF)$ to be redistributed to artists based on content sharing

Curator redistribution of TUNCs

$$(Z \times USF) = (\alpha_c + \beta_c)(Z \times USF)$$

$$\alpha_c + \beta_c = 1$$

α_c = percentage of $(Z \times USF)$ to be redistributed to artists based on content consumption
 β_c = percentage of $(Z \times USF)$ to be redistributed to artists based on content sharing

Fan redistribution of TUNCs

$$(W \times USF) = \text{TUNC revenue from annual fees.}$$

TUNE redistribution of TUNCs

$$(V \times USF) = (\alpha_f + \beta_f)(V \times USF)$$

$$\alpha_f + \beta_f = 1$$

α_f = percentage of $(V \times USF)$ to be redistributed to artists based on content consumption
 β_f = percentage of $(V \times USF)$ to be redistributed to artists based on content sharing

Artist Revenue not including mining

Different types of artists.

Sponsored Artists

Sponsorship with advertisers through smart contracts creates revenue for artists.

Un-sponsored Artists

Revenue is gained through smart contracts with curators if curators are sponsored and through USFs

Artists can have the freedom to have sponsored and unsponsored content if they wish.

Sponsored Artist

Revenue sources come from:

- Content consumption by fans
- Content sharing by fans
- Content consumption/sharing from curators.
- Tipping by fans

Artist Revenue- a_r

$$a_r = c_r + s_r + g$$

$c_r = \text{value of content consumption}$

$s_r = \text{value of content sharing}$

$g = \text{value of tipping}$

Content consumption- c_r

$$c_r = \sigma_d + \sigma_{in} + X_a$$

$\sigma_d = \text{direct revenue from content consumption by fans}$

The value of content consumption by fans is decided by Smart Contracts between sponsors and artists. It can be based on the number of “consumptions”. It can also have caps etc.

$\sigma_{in} = \text{indirect revenue from content consumption by fans}$

The value of content consumption by fans is decided by Smart Contracts between artists and curators. It can be based on the number of “consumptions”. It can also have caps etc.

$$\rho_a = \frac{X_a = (X \times USF \times \alpha_a) \times \rho_a}{\text{FAN CONSUMPTION OF SPECIFIC ARTIST}} \div \text{TOTAL FAN CONSUMPTION}$$

Content sharing- s_r

$$s_r = \varepsilon_d + \varepsilon_{in} + Y_a$$

$\varepsilon_d = \text{direct revenue from content sharing by fans}$

The value of content consumption by fans is decided by smart contracts between sponsors and artists. It can be based on the number of “shares”.

$\varepsilon_{in} = \text{indirect revenue from content sharing by fans}$

The value of content consumption by fans is decided by smart contracts between artists and curators. It can be based on the number of “shares”.

$$Y_a = (X \times USF \times \beta_a) \times \gamma_a$$

$$\gamma_a = \frac{FAN \ SHARING \ OF \ SPECIFIC \ ARTIST}{TOTAL \ FAN \ SHARING}$$

Artist tipping-g

Un-sponsored Artist

Fans have the ability to tip artists if they like their work.

Revenue sources come from:

- Content consumption by fans
- Content sharing by fans
- Content consumption/sharing from curators.
- Fan tipping

Artist Revenue- ar

$$a_r = c_r + s_r + g$$

$c_r = \text{value of content consumption}$
 $s_r = \text{value of content sharing}$
 $g = \text{value of tipping}$

Content consumption- cr

$$c_r = \sigma_d + \sigma_{in} + X_a$$

$$\sigma_d = 0$$

$\sigma_{in} = \text{indirect revenue from content consumption by fans}$

The value of content consumption by fans is decided by Smart Contracts between artists and curators. It can be based on the number of “consumptions”. It can also have caps etc.

$$X_a = (X \times USF \times \alpha_a) \times \rho_a$$

$$\rho_a = \frac{FAN \ CONSUMPTION \ OF \ SPECIFIC \ ARTIST}{TOTAL \ FAN \ CONSUMPTION}$$

Content sharing- sr

$$s_r = \varepsilon_d + \varepsilon_{in} + Y_a$$

$$\varepsilon_d = 0$$

$\varepsilon_{in} = \text{indirect revenue from content sharing by fans}$

The value of content consumption by fans is decided by smart contracts between artists and curators. It can be based on the number of “shares”.

$$Y_a = (X \times USF \times \beta_a) \times \gamma_a$$

$$\gamma_a = \frac{FAN \ SHARING \ OF \ SPECIFIC \ ARTIST}{TOTAL \ FAN \ SHARING}$$

Artist tipping-g

Fans have the ability to tip artists if they like their work.

Artist Expenses

Artist Expenses will come from two main areas

Smart Contracts with Producers.
Smart Contracts with TUNE.

Smart Contracts with Producers

$$a_p = p_i + c_p + s_p$$

a_p = total artist expenses with producers
 p_i = initial cost of production
 c_p = value of content consumption
 s_p = value of content sharing

c_p and s_p are determined by a smart contract between artist and producers. It can be based on number of consumptions and shares.

Smart Contracts with TUNE

$$a_t = A + c_f + s_f$$

a_t = total artist expenses with TUNE
 A = Annual fee
 c_f = loss due to fan consumption
 s_f = loss due to fan sharing

c_f and s_f are determined by a smart contract between artist and TUNE it is taken from the advertisement revenue and given to fans. It can be based on number of consumptions and shares.

Artist Profit

$$a = a_r - a_p - a_t$$

$$a = c_r + s_r + g - (p_i + c_p + s_p) - (A + c_f + s_f)$$

$$a = \sigma_d + \sigma_{in} + g + X_a + \varepsilon_d + \varepsilon_{in} + Y_a - (p_i + c_p + s_p) - (A + c_f + s_f)$$

$$a = \sigma_d + \sigma_{in} + \varepsilon_d + \varepsilon_{in} + g + [(X \times USF)((\alpha_a \times \rho_a) + (\beta_a \times \gamma_a))] - (p_i + c_{pr} + s_{pr}) - (A + c_f + s_f)$$

$$\rho_a = \frac{\text{FAN CONSUMPTION OF SPECIFIC ARTIST}}{\text{TOTAL FAN CONSUMPTION}} \gamma_a$$

$$= \frac{\text{FAN SHARING OF SPECIFIC ARTIST}}{\text{TOTAL FAN SHARING}}$$

Producer Revenue

A producer is basically an artist that also allows their content to be edited by other artists. Producers and artist will agree on smart contract that will decide their revenue split.

Revenue sources come from:

- Content consumption by fans

- Content sharing by fans
- Content consumption/sharing from curators.

Producer Revenue- p_r

$$p_r = p_i + c_{pr} + s_{pr}$$

$p_i = \text{initial cost of production}$
 decided by producer (can be zero)
 $c_{pr} = \text{value of content consumption}$
 $s_{pr} = \text{value of content sharing}$

Content consumption- c_{pr}

$$c_{pr} = c_p + X_p$$

$c_p = \text{direct revenue from content consumption by fans}$
 The value of content consumption by fans is decided by Smart Contracts between artists and producers. It can be based on the number of “consumptions”. Can also be equal to zero.
 $X_p = (Y \times USF \times \alpha_p) \times \rho_p$

$$\rho_p = \frac{\text{FAN CONSUMPTION OF SPECIFIC PRODUCER}}{\text{TOTAL FAN CONSUMPTION}}$$

Content sharing- s_{pr}

$$s_{pr} = s_p + Y_p$$

$s_p = \text{direct revenue from content sharing by fans}$
 The value of content consumption by fans is decided by smart contracts between artists and producers. It can be based on the number of “shares”.
 $Y_p = (Y \times USF \times \beta_p) \times \gamma_p$

$$\gamma_p = \frac{\text{FAN SHARING OF SPECIFIC PRODUCER}}{\text{TOTAL FAN SHARING}}$$

Producer Expenses

Producer Expenses come from:
 Smart Contracts with TUNE

Smart Contracts with TUNE

$$p_t = P$$

$p_t = \text{total producer expenses with TUNE}$
 $P = \text{Annual fee}$

Producer Profit

$$\begin{aligned}
 p &= p_r - p_t \\
 p &= p_i + c_{pr} + s_{pr} - P \\
 p &= p_i + c_p + X_p + s_p + Y_p - P \\
 p &= p_i + c_p + s_p + [(Y \times USF)((\alpha_p \times \rho_p) + (\beta_p \times \gamma_p))] - P \\
 \rho_p &= \frac{FAN \ CONSUMPTION - SPECIFIC \ PRODUCER}{TOTAL \ FAN \ CONSUMPTION} \gamma_p \\
 &= \frac{FAN \ SHARING - SPECIFIC \ PRODUCER}{TOTAL \ FAN \ SHARING}
 \end{aligned}$$

Curator Revenue

Curator is sponsored to share their choice of music. A fan is a curator that is not sponsored.

Revenue sources come from:

- Content consumption by fans
- Content sharing by fans
- Tipping by fans

Curator Revenue- u_r

$$\begin{aligned}
 u_r &= c_c + s_c + g \\
 c_c &= \text{value of content consumption} \\
 s_c &= \text{value of content sharing} \\
 g &= \text{value of tipping}
 \end{aligned}$$

Content consumption- c_c

$$c_c = \pi_c + X_c$$

π_c = direct revenue from content consumption by fans

The value of content consumption by fans is decided by Smart Contracts between sponsors and curators. It can be based on the number of “consumptions”. It can also have caps etc.

$$\begin{aligned}
 X_c &= (Z \times USF \times \alpha_c) \times \rho_c \\
 \rho_c &= \frac{FAN \ CONSUMPTION \ OF \ SPECIFIC \ CURATOR}{TOTAL \ FAN \ CONSUMPTION}
 \end{aligned}$$

Content sharing- s_c

$$s_r = \omega_c + Y_c$$

$\omega_c = \text{direct revenue from content sharing by fans}$

The value of content consumption by fans is decided by smart contracts between sponsors and curators. It can be based on the number of “shares”.

$$Y_c = (Z \times USF \times \beta_c) \times \gamma_c$$

$$\gamma_c = \frac{\text{FAN SHARING OF SPECIFIC CURATOR}}{\text{TOTAL FAN SHARING}}$$

Curator Expenses

Curator Expenses will come from two main areas

- Smart Contracts with Artists
- Smart Contracts with Tune

Smart Contracts with Artists

$$u_p = \sigma_{in} + \varepsilon_{in}$$

$u_p = \text{total curator expenses with artists}$

$\sigma_{in} = \text{value of content consumption}$

Determined by smart contract with artist

$\varepsilon_{in} = \text{value of content sharing}$

Determined by smart contract with artists

Smart Contracts with Tune

$$u_t = C + \pi_f + \omega_f$$

$u_t = \text{total artist expenses with TUNE}$

$C = \text{Annual fee}$

$\pi_f = \text{loss due to fan consumption}$

Determined by smart contract between curator and TUNE. It is taken from advertisement revenue and given to fans.

$\omega_f = \text{loss due to fan sharing}$

Determined by smart contract between curator and TUNE. It is taken from advertisement revenue and given to fans.

Curator Profit

$$u = u_r - u_p - u_t$$

$$u = c_c + s_c + g - (\sigma_{in} + \varepsilon_{in}) - (C + \pi_f + \omega_f)$$

$$u = \pi_c + X_c + g + \omega_c + Y_c - (\sigma_{in} + \varepsilon_{in}) - (C + \pi_f + \omega_f)$$

$$u = \pi_c + \omega_c + g + [(Z \times USF)((\alpha_c \times \rho_c) + (\beta_c \times \gamma_c))] - (\sigma_{in} + \varepsilon_{in}) - (C + \pi_f + \omega_f)$$

$$\rho_c = \frac{\text{FAN CONSUMPTION - SPECIFIC PRODUCER}}{\text{TOTAL FAN CONSUMPTION}} \gamma_c$$

$$= \frac{\text{FAN SHARING - SPECIFIC PRODUCER}}{\text{TOTAL FAN SHARING}}$$

Fan Revenue

A fan is an unsponsored curator

Revenue sources for fans come from:

- content consumption
- content sharing

Fan Revenue- f_r

$$f_r = c_{pf} + s_{pf}$$

c_{pf} = value of content consumption

s_{pf} = value of content sharing

Content consumption- c_{pf}

$$c_{pf} = c_f + \pi_f + X_f$$

c_f = direct revenue from content consumption by fans

The value of content consumption by fans is decided by Smart Contracts between artists and TUNE. It is taken from the ad revenue of sponsored artists.

π_f = indirect revenue from content consumption by fans

The value of content consumption by fans is decided by Smart Contracts between curators and TUNE. It is taken from the ad revenue of curators.

$$X_f = (V \times USF \times \alpha_f) \times \rho_f$$

$$\rho_f = \frac{\text{CONSUMPTION BY SPECIFIC FAN}}{\text{TOTAL FAN CONSUMPTION}}$$

Content sharing- s_{pf}

$$s_{pf} = s_f + \omega_f + Y_f$$

s_f = direct revenue from content sharing by fans

The value of content consumption by fans is decided by smart contracts between artists and TUNE. It is taken from ad revenue of artists.

ω_f = indirect revenue from content sharing by fans

The value of content consumption by fans is decided by smart contracts between curators and TUNE. It is taken from ad revenue of curators.

$$Y_f = (V \times USF \times \beta_f) \times \gamma_f$$

$$\gamma_f = \frac{\text{SHARING BY SPECIFIC FAN}}{\text{TOTAL FAN SHARING}}$$

Fan Expenses

Fan Expenses come from

- Smart Contracts with TUNE
- Tipping

Smart Contracts with TUNE.

$$f_t = F + g$$

f_t = total producer expenses with TUNE

g = value of tipping

F = Annual fee

Fan Profit

$$f = f_r - f_t$$

$$f = c_{pf} + s_{pf} - F - g$$

$$f = c_f + \pi_f + X_f + s_f + \omega_f + Y_f - F - g$$

$$p = c_f + \pi_f + s_f + \omega_f + [(V \times USF)((\alpha_f \times \rho_f) + (\beta_f \times \gamma_f))] - F - g$$

$$\rho_f = \frac{\text{CONSUMPTION OF SPECIFIC FAN}}{\text{TOTAL FAN CONSUMPTION}} \gamma_f$$

$$= \frac{\text{SHARING OF SPECIFIC FAN}}{\text{TOTAL FAN SHARING}}$$

Advertisers Expenses

Advertisers have three expenses:

Smart Contracts with Artists

Smart Contracts with Curators

Smart Contract with TUNE

Expenses

$$d_e = \sigma_d + \pi_c + Ad$$

σ_d = direct revenue from content consumption by fans

The value of content consumption by fans is decided by Smart Contracts between sponsors and artists. It can be based on the number of "consumptions". It can also have caps etc.

π_c = direct revenue from content consumption by fans

The value of content consumption by fans is decided by Smart Contracts between sponsors and curators. It can be based on the number of "consumptions". It can also have caps etc.

Ad = Annual fee

Ad is a smart contract between TUNE and Advertisers.

Case Study 1

Amateur Artist

Kyle is a 19-year-old up and coming artist from Cleveland. Kyle doesn't have a lot of money. He has a large fan-base of over 12,000 followers across his social media accounts including Facebook, YouTube, Instagram and Twitter. But he wants to further his career and thinks releasing a single on Spotify will help him gain more followers and possibly generate royalties from the Spotify streams.

However, Kyle can't afford the high cost of professional recording equipment and doesn't have access to quality music producers. And even if he did, he wouldn't be able to afford their production.

There are also additional costs associated with uploading his music on Spotify and it takes some time and diligence to register his song rights properly.

Bottom line- Kyle will have to forgo his chances of growing his base and sharing his art with the world.

And then, Kyle discovers Cre8tor.app (formerly RecordGram)m and the integrated Tune blockchain tokenized ecosystem. With Tune, Kyle can register with Cre8tor.app (formerly RecordGram)m as an artist and monetize his work as part of a massive music ecosystem where others can share his content and organically grow his exposure. Where producers can discover and collaborate with talent that they wouldn't have otherwise seen.

Case Study 2

Playlist Queen

Shayla, from Germany, is the person you go to when you need that perfect playlist for any occasion. When not at work or at school, Shayla is constantly curating amazing content on SoundCloud and Spotify across various genres. Finding unknown artists and bringing them to forefront of her 2,000 followers. However, she wishes that she could somehow get compensated for her time curating the perfect playlist. She's also growing her own following as tastemaker for discovering talent much like an A&R at a major record label is known for.

Through Tune, not only does she benefit from getting rewarded every time one of her followers engages with her curated content, but producers and artists will also receive partial compensation, talent discovery and collaboration from her shared, curated playlists.

Case Study 3

Upcoming Producer

Tony is an upcoming producer from Japan. He's extremely talented and makes great beats. Unlike Kyle our aspiring artist from Cleveland, Tony has some money and spends it all on expensive production equipment, even building his own recording studio.

However, Tony doesn't have relationships with artists like Kyle to sell his production and make money from his craft. So, he resorts to uploading his beats for free across all online platforms in the hopes of getting discovered.

And then Tony discovers Cre8tor.app (formerly RecordGram)m and Tune's ecosystem which allows him to get compensated from his beats through a user friendly, affordable and convenient mobile platform that puts his beats in front of artists.

Furthermore, with Tune he's able to register his song rights and get transparently compensated every time his beat is recorded over, shared or licensed. Access between artist's like Kyle and producer's like Tony is democratized and signals the redefinition of the music industry.

Token Economics and Distribution

Token Launch

Token: Tune Token

Symbol: TUNE

Total Supply: 1,000,000,000 TUNE

Total Being Sold: 400,000,000 TUNE

Payment forms Accepted: ETH, BTC

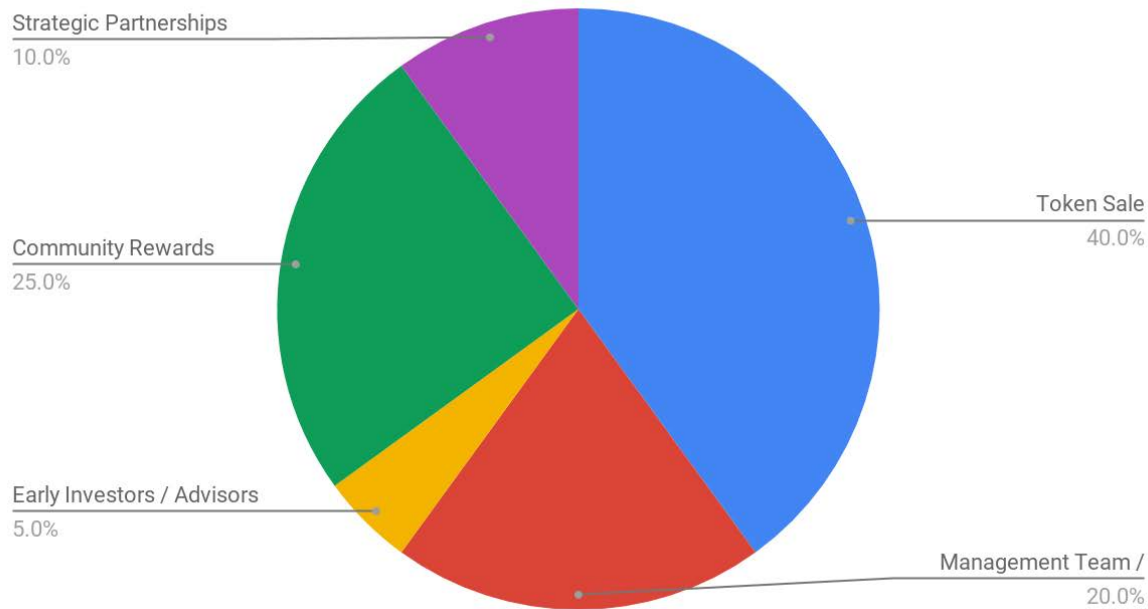
Public Sale: November/December 2018* (if needed)

Wallet Airdrops for existing Cre8tor.app
(formerly RecordGram) users will be distributed
between private and crowdsale.

Tune tokens will be distributed via smart
contracts 1 month from the public token sale so
we can properly audit the smart contracts.

*dates subject to change

Total Token Supply Allocation - 1 billion TUNE



Token Sale

- For the funding of research and development, engineering, business development, legal, administrative and miscellaneous expenses that will enhance the platform

Community Rewards

- Funding for incentivized programs for both existing community participants and those seeking entrance into our platform

Early Investors and Advisors

- Funding for early investors and advisors' members that provide the direction and expertise in business, media and blockchain, delivering on the vision of Tune.

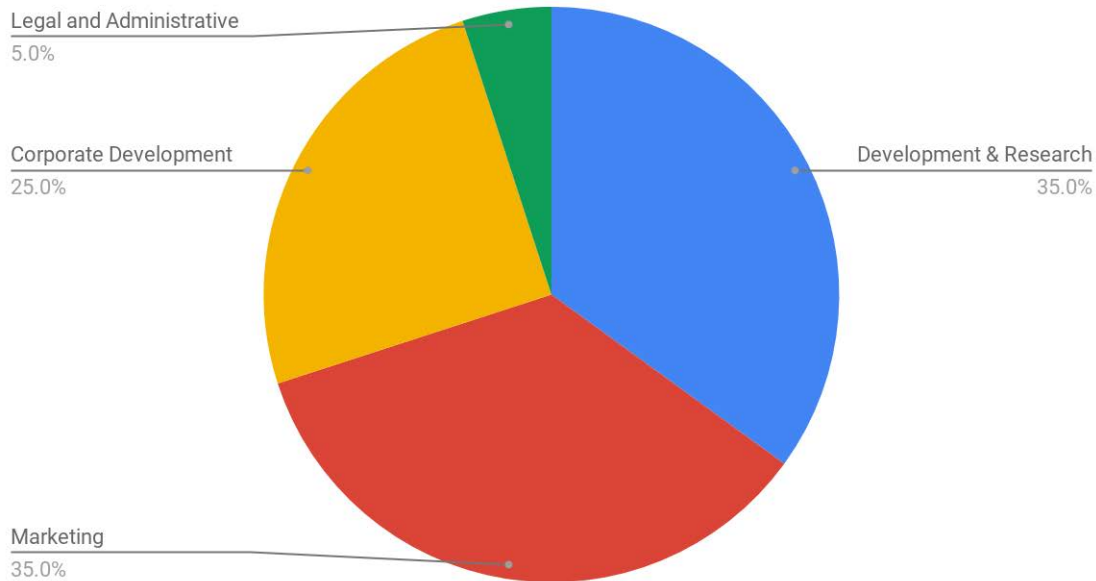
Strategic Partnerships

- Funding for incentivized programs for both existing community participants and those seeking entrance into our platform

Management Team / Founders

- Founders, Management Execs, early employees

Use of Funds from Token Sale



Development and Research

- Funding for full-time salaried employees that will be working to create and maintain the platform

Corporate Development

- Funding to deliver on business objectives that includes strategic partnerships that drives user growth and adoption.

Marketing

- Funding to build a dependable and robust marketing campaign designed to draw in participants from several different demographics and geographical areas

Legal and Administrative

- Funding for legal, accounting and other third-party service to run the organization

Timeline

Cre8tor.app (formerly RecordGram) Timeline

1. Nov 2015 - Inception of idea
2. Jan 2016 - 1 of 6 companies accepted for the Project Music accelerator
3. May 2016 - Completed Project Music accelerator
4. June 2016 - Beta version RG released in iOS store
5. Sep 14 2016- First place at the YPO Global Networking Summit
6. Nov 2016 - 1 of 36 companies selected to appear on Apple music original content show Planet of the Apps
7. Feb 2017 - Final taping of Planet of the Apps
8. April 2017 - Close seed round with Lightspeed Venture Partners, New World Angels (NWA), Monique Mosley
9. May 2017 - Won TechCrunch Disrupt NYC1
10. June 2017 - Started developing white paper and Blockchain integration
11. Feb - 2018 Android version released
12. Feb - 2018 Published White paper
13. Feb - 2018 Launch tunetoken.io
14. November – 2018 Finalize Private Sale
15. Nov-December 2018- Initiate Public Sale on Element Group Platform

Tune Timeline

1. February 2018 - Published White Paper
2. KYC/AML Integration and Token Economics on website
3. March 25th - Develop platform and token with first use-case integration
4. End of Q1 - Token Sale
5. Beginning of Q2 - Launch Tune token, beta platform and integration with Cre8tor.app (formerly RecordGram)m application
 - a. demonstrate use-case for community stakeholder participation in music distribution and consumption
6. Middle of Q2 - Expand platform and begin testing on alternative blockchains and protocols
 - a. begin leveraging decentralized content storage and off-chain solutions for enhanced user and community experience
7. End of Q2 - Add additional Partners to Ecosystem
 - a. partners will be key for the Tune community. We need reliable and creative partners to drive the value of this market, maintain and invigorate it
8. Beginning of Q3 - Open Tune platform source code and begin community development
 - a. Developers are key to our success in that they will be the ones truly innovating new and novel ways to implement our vision and ultimately make the community a better place.

Team

The Founders



Erik Mendelson, CEO / CoFounder

As an Executive Music Producer and artist manager, Erik has signed, managed, and developed 3 platinum recording artists, generating over \$20 million in gross revenue under his direction.

Shawn Mims, Co-founder /

Creative Director & Celebrity Relations

Shawn Mims or p/k/a, MIMS is a multi-platinum recording artist, Grammy award winning writer, well-known for his global #1 single This Is Why I'm Hot, and engineer who has generated over \$16 million in gross revenue for EMI/Capitol Records.



Winston Thomas, Co-founder / Product Manager

Winston Thomas or p/k/a DJ Blackout is a Grammy Award Winning music producer and international DJ who has produced over 30 commercial records including a recent #1 single in India and previous #1 global record.

Music Business Advisors



will.i.am: Grammy winning musician, rapper, singer, songwriter, record producer, entrepreneur, actor, and philanthropist.

Monique Mosley: Globally recognized entertainment executive, Angel investor with Reign Venture Capital—One of Gotham Magazine’s Powerful 100 names to know.



Brian Zisk: Creator of the Future of Money & Technology Summit, Creator of San Fran Music Tech Summit, and Partner at SF Music Tech Fund

Cameo Carlson: President mtheory; Previously: Head of Digital Biz Development/Borman Entertainment, Executive Vice President at Universal Records, Label Relations & Music Programming Manager for Apple iTunes





Samantha Saturn: President Digital Marketing Agency; Previously: CMO comiXology (a division of Amazon.com), Senior Vice President Digital Marketing for Columbia Records

Chris Sloan ESQ: Chair Emerging Companies at Baker Donelson



Bill Brennan: Serial Entrepreneur, Mentor, Angel Investor, Speaker
Mr. Brennan has 30+ years business experience as a founder and/or co-founder of 10+ start-ups.

Beth Raebeck Hall: Grammy nominated singer/songwriter with over 20 years in the Music Business





Ryder Lee: Independent law practice with emphasis in Entertainment & Media, Intellectual Property, and Emerging Businesses / Entrepreneurial Ventures.

Blockchain / Technology Advisors

Shariar Sikder: Cre8tor.app (formerly RecordGram) CTO. Technology expert and International Business enthusiast



Gary Manheimer: New World Angels. Seasoned business executive and serial entrepreneur with over 25 year's experience. Consultant and Entrepreneur. Board Member and Advisor to several successful companies

Justin Wu: Information Architect, Token Advisor and Growth Marketer for Blockchain companies. He has collaborated with both Wall Street Journal, CES and NASDAQ. In the past, he has founded several companies such as Sidevision (Invested by Warner Bros), and Vytmn.com, a marketing tech company where he led growth \$1M revenue in its first year.



Sam Abbassi: Blockchain architect and engineer, tokenomics analyst, ethereum and Hyperledger developer



Christopher Arguello: Blockchain engineer, Ethereum and Hyperledger developer, dApp architect

Legal Disclaimer

Important Notice

PLEASE READ THIS SECTION CAREFULLY AS IT CONTAINS IMPORTANT INFORMATION. IF YOU ARE IN ANY DOUBT AS TO THE ACTION YOU SHOULD TAKE, YOU SHOULD CONSULT YOUR PERSONAL LEGAL, FINANCIAL, TAX OR OTHER PROFESSIONAL ADVISOR(S).

The Tune Tokens (as defined in this Whitepaper) are not intended to constitute securities in any jurisdiction. This Whitepaper does not constitute a prospectus or offer document of any sort and is not intended to constitute an offer of securities or a solicitation for investment in securities in any jurisdiction.

This Whitepaper does not constitute or form part of any opinion on any advice to sell, or any solicitation of any offer by the distributor of the Tune Tokens to purchase any Tune Tokens nor shall it or any part of nor the fact of its presentation form the basis of, or be relied upon in connection with, any contract or investment decision.

The distributor may be an affiliate of Cre8tor.app (formerly RecordGram), a Delaware corporation (individually or collectively “Cre8tor.app (formerly RecordGram) and will use all proceeds of sale of the Tune Tokens to grow Cre8tor.app (formerly RecordGram) cryptocurrency project, businesses and operations.

No person is bound to enter into any contract or binding legal commitment in relation to the sale and purchase of the Tune Tokens, and no cryptocurrency or other form of payment is to be accepted based on this Whitepaper.

No part of this Whitepaper is to be reproduced, distributed or disseminated without including this section and the following sections entitled “Disclaimer of Liability”, “No Representations and Warranties”, “Representations and Warranties By You”, “Cautionary Note on Forward-Looking

Statements”, “No Advice”, “Restrictions on Distribution and Dissemination”, “No Offer of Securities or Registration” and “Risks and Uncertainties”.

This Whitepaper is a draft, and Tune Software may revise its contents at any time without notice to you.

DISCLAIMER OF LIABILITY

To the maximum extent permitted by the applicable laws, regulations and rules, Tune Software shall not be liable for any indirect, special, incidental, consequential or other losses of any kind, in tort, contract or otherwise (including but not limited to loss of revenue, income or profits, and loss of use or data), arising out of or in connection with any acceptance of or reliance on this Whitepaper or any part thereof by you.

NO REPRESENTATION AND WARRANTIES

Tune Software makes or purports to make no, and hereby disclaims, any representation, warranty or undertaking in any form whatsoever to any entity or person, including any representation, warranty or undertaking in relation to the truth, accuracy and completeness of any of the information set out in this Whitepaper.

REPRESENTATIONS AND WARRANTIES BY YOU

By accessing or accepting possession of any information in this Whitepaper or any part thereof, you represent and warrant to Tune Software as follows:

1. you agree and acknowledge that the Tune Tokens do not constitute securities in any form in any jurisdiction;
2. you agree and acknowledge that this Whitepaper does not constitute a prospectus or offer document of any sort and is not intended to constitute an offer of securities in any jurisdiction or solicitation for investment in securities and you are not bound to enter into any contract or

binding legal commitment and no cryptocurrency or other form of payment is to be accepted based on this Whitepaper;

3. you agree and acknowledge that no regulatory authority has examined or approved of the information set out in this Whitepaper, no action has been or will be taken under the laws or regulations of any jurisdiction, and the publication, distribution or dissemination of this Whitepaper to you does not imply that any applicable laws or regulations have been complied with;

4. you agree and acknowledge that this Whitepaper, the Tune Token Distribution Event (TDE), or future trading of the Tune Tokens on any cryptocurrency exchange, shall not be construed, interpreted or deemed by you as an indication of the merits of Tune Software, the Tune Tokens or the Tune TDE;

5. the distribution or dissemination of this Whitepaper, any part thereof or any copy thereof, or acceptance of the same by you, is not prohibited or restricted by any laws and regulations applicable to you, and where any restrictions in relation to possession are applicable, you have observed and complied with all such restrictions at your own expense and without liability to Tune Software;

6. you agree and acknowledge that if you elect to purchase any Tune Tokens, the Tune Tokens are not to be construed, interpreted, classified or treated as:

- a. any kind of currency other than cryptocurrency;
- b. debentures, stocks or shares issued by any person or entity (whether Tune Software), including any rights, options or derivatives thereof;

- c. units in a collective investment scheme;
- d. units in a business trust;
- e. derivatives of units in a business trust; or
- f. any other security or class of securities.

7. you have a basic degree of understanding of cryptocurrencies, blockchain-based software systems, cryptocurrency wallets or other related token storage mechanisms, blockchain technology and smart contract technology;

8. you are aware and understand that in the case where you wish to purchase any Tune Tokens, there are material risks associated with Tune Software and business, plans and operations, the Tune Tokens, and the Tune TDE;

9. you agree and acknowledge that Tune Software or its affiliates are not liable for any indirect, special, incidental, consequential or other losses of any kind, in tort, contract or otherwise (including but not limited to loss of revenue, income or profits, and loss of use or data), arising out of or in connection with any acceptance of or reliance on this Whitepaper or any part thereof by you;

10. all the above representations and warranties are true, complete, accurate and non-misleading from the time of your access to and/or acceptance of possession this Whitepaper or such part thereof (as the case may be).

CAUTIONARY NOTE ON FORWARD-LOOKING STATEMENT

This Whitepaper may contain forward-looking statements including, but not limited to, statements as to future operating results and plans that involve risks and uncertainties. We use words such as “will”, “expects”, “anticipates”, “believes”, “estimates”, the negative of these terms

and similar expressions to identify forward-looking statements. Such forward looking statements involve known and unknown risks, uncertainties and other factors which may cause the actual results, performance or achievements of Tune Software to differ materially from any future results, performance or achievements expressed or implied by those projected in the forward-looking statements for any reason.

NO ADVICE

No information in this Whitepaper should be considered business, legal, financial or tax advice regarding Tune Software, the Tune Tokens or the Tune TDE. You should consult your own legal, financial, tax or other professional adviser regarding Tune Software and their respective businesses and operations, the Tune Token, and the Tune TDE. You should be aware that you may be required to bear the financial risk of any purchase of Tune Tokens for an indefinite period.

RESTRICTIONS ON DISTRIBUTION AND DISSEMINATION

The distribution or dissemination of this Whitepaper or any part thereof may be prohibited or restricted by the laws, regulatory requirements and rules of any jurisdiction. In the case where any restriction applies, you are to inform yourself about, and to observe, any restrictions that are applicable to your possession of this Whitepaper or such part thereof (as the case may be) at your own expense and without liability to Tune Software. Persons to whom a copy of this Whitepaper has been distributed or disseminated, provided access to or who otherwise have the Whitepaper in their possession shall not circulate it to any other persons, reproduce or otherwise distribute this Whitepaper or any information contained herein for any purpose whatsoever nor permit or cause the same to occur.

NO OFFER OF SECURITIES OR REGISTRATION

This Whitepaper does not constitute a prospectus or offer document of any sort and is not intended to constitute an offer of securities or a solicitation for investments in securities in any jurisdiction. No person is bound to enter into any contract or binding legal commitment, and no cryptocurrency or other form of payment is to be accepted based on this Whitepaper. Any agreement governing to any sale and purchase of Tune Tokens is to be governed by only the Purchase Agreement or such other agreement or document as Tune Software may require and no other document (including this Whitepaper). In the event of any inconsistencies between the Purchase Agreement and this Whitepaper, the Purchase Agreement shall prevail.

No regulatory authority has examined or approved of any of the information set out in this Whitepaper. No such action has been or will be taken under the laws, regulatory requirements or rules of any jurisdiction. The publication, distribution or dissemination of this Whitepaper does not imply that the applicable laws, regulatory requirements or rules have been complied with.

RISKS AND UNCERTAINTIES

Prospective purchasers of Tune Tokens should carefully consider and evaluate all risks and uncertainties associated with Tune Software and their respective plans, businesses and operations, and the Tune TDE, including all information set out in this Whitepaper and any Purchase Agreement prior to any purchase of Tune Tokens. If any of such risks and uncertainties develops into actual events, the business, financial condition, results of operations and prospects of Tune Software could be materially and adversely affected. In such cases, you may lose all or part of the value of the Tune Tokens.