Orphic Finance:
Financial Ecosystem

Version 0.2
May 2021

Authors
Orphic Finance Team
https://orphic.finance

Abstract
In this paper we introduce a series decentralized programs comprising the Orphic Finance eco-system enabling users to create derivative stake tokens, frictionlessly sell them through a liquidity pool, sell them through a market palace leveraging a token registry and providing money markets through algorithmically set interest rates based on supply and demand through collateralized assets.

1.   Introduction

This whitepaper is a view of the Orphic Finance ecosystem based on the current state of Solana's ecosystem and a series of assumptions that have been made by the team. This paper will be updated to a newer version in the near future to capture lessons learned and optimal path forward.

The Orphic Finance ecosystem's original purpose was to provide users with the flexibility to borrow against their staked tokens. The way that this can be provided is through the Cosmic Eggs protocol through derivative staking tokens, Archeio became necessary to calculate the value of the tokens, Protogonos to have the flexibility to sell to other users in a series of different ways and finally Orphic Finance Protocol to provide users with liquidity pools that ultimately enable the ability to borrow against provided collateral (including Cosmic Eggs, OFIN tokens and SOL). This was the base case that was defined by the Orphic Finance development team, however during the development process and the evolution of the ecosystem, the vision and mandate has grown to cover many additional use-cases.

2.   Orphic Finance Protocol
2.1.   Introduction

The Orphic Finance protocol is a system that enables users to lend or borrow a series of different tokens. The lending and borrowing rates are determined algorithmically for the baseline rate (the referential rate comes from Archeio) and then take into account the forces of supply and demand. We plan on beginning with Solana and OFIN Tokens as the first pools.

2.2.   Orphic Finance Protocol Process

The workflow for the Orphic Finance Protocol is the end to end process is two fold, one workflow process for the Lender and one workflow process for the Borrower. The process of lending begins with joining a liquidity pool. A user can choose a liquidity pool to join, either a specific token based pool or a value based liquidity pool. All lending returns are split amongst the participants of the liquidity pool. When a user wants to withdraw from a liquidity pool, the user is put in a queue and once the borrowers have paid their loans back, the user will be able to withdraw their principal and interest. Users can select to receive interest to their wallet (or associated wallet) or can automatically compound the interest and re-lend it. The process of borrowing begins with providing collateral against which to borrow.
When the user borrows against a liquidity pool, they agree to the following conditions: they will pay the daily interest to keep the loan active, if they fail to pay interest their collateral may (certain amount of leeway may be provided) be sold to meet their interest obligations and if the user's collateral drops below the LTV % outlined by the protocol the collateral will be liquidated to meet the borrowing obligation.

Users can either select to pay via netting (assuming the collateral is tuned for it), can prepay or pay as they go. Days are rounded up and thus the start of a day means the user is responsible for paying that day's interest. The user can close the loan at any time pursuant to paying the interest required and once obligations are met can withdraw the collateral posted (or what is left with it).

2.3. Liquidity Pools

There are three flavors of liquidity pools:
- Defined Asset Liquidity Pools - These liquidity pools are single asset based and the liquidity provider (lender) can expect to get back the same asset they put in plus any interest earned. *Example: A user (liquidity provider) provides 50 SOL to the Solana Liquidity Pool and earns .0275% per day*
- Relative Value Liquidity Pools - These liquidity pools are based on the value of the account provided and may include shifts into other assets using atomic swaps (solutions are currently being explored and considerations such subscribe to Liquidity Pools are being considered)
- Paired Liquidity Pools - These are used for swaps between the group of coins and are common to Lending protocols such as Uniswap.

2.4. Lending Detail

The Liquidity Provider transfers tokens into lending pools and receives a representational token to denote the users relative share. The LP receives daily interest from the Borrower, either via netting or via direct payment or via metered prepayment. Interest is directly re-added to the LP, until the time at which the Lender wants to leave the Liquidity Pool. Although unlikely, it is possible that the pool will be unable to enable withdrawals until after borrowers have paid back their obligations. The representational token will therefore grow in value. Day 1 the token for the SOL pool will be 1:1, however as interest is earned, the price will rise and adding money to the pool at a later date will yield less than 1 LP token.

2.5. Borrowing Detail

The Borrower begins the process by choosing a market they want to borrow from. The user then, next locks in collateral (collateral will need to be validated as liquid and have a defined margin call path - this will involve a swap or other tools) into an escrow account where the amount to be borrowed will be based on an LTV % (LTV will be a governance voted percentage - however there will likely be separate pools that support various LTVs, as the higher the LTV the higher the yield should be as there is risk that the principal will not be paid back).
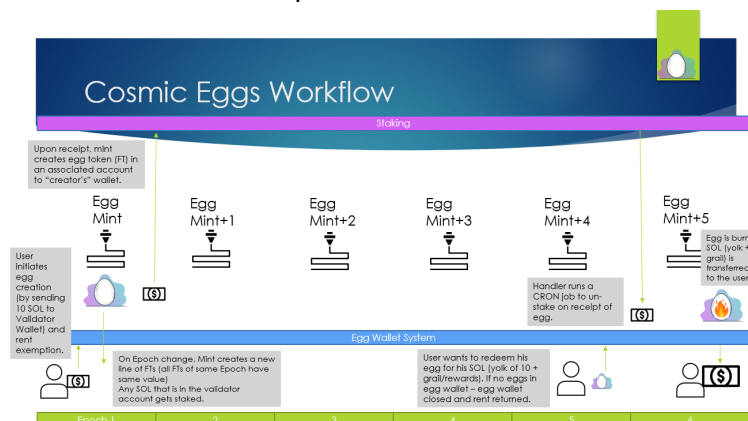
3. Cosmic Eggs

3.1. Introduction

Cosmic Eggs enables liquid staking through "defined amount" stakes providing partial fungibility. This is achieved by allowing users to stake through Cosmic Eggs which are representational of amounts staked on a validator. The current sizing is 10 SOL, however through the OFIN governance token voting mechanism we will potentially list a series of varied sizes; (100 SOL, 1000 SOL etc.). Cosmic Egg users get value from being able to resell their Cosmic Egg tokens to other users, use it to borrow additional funds or being able to Eggcinerate them without having to wait for the Epoch change.

3.2. Cosmic Eggs Process

The workflow for Cosmic Eggs is the end to end process which is started by the "creation" of an egg. This process is kicked off by the Cosmic Eggs user interface (this may be exposed at a later date via API), prompting the user to send 10 SOL and including rent exemption to create an associated wallet to the Cosmic Eggs program. The program, on confirmation of the receipt of the Solana, will create a mint and proceed to mint a token for the user to their newly created associated account. During the epoch any additional users who create a Cosmic egg, will have a token minted from the mint. All solana sent to the program in a given Epoch is put in an Epoch based staking account. The mint and epoch based staking account information are logged to Archeio in order to calculate rewards per epoch and "know" the value of the underlying eggs. Cosmic Eggs can be sold to other users, used as collateral or redeemed. Redemption of an egg requires a process through the Cosmic Eggs user interface (this may be exposed at a later date via API) prompting the user to send their Egg(s) back to the program for redemption. Once the egg is confirmed via the Cosmic Eggs program, Archeio is used to verify the correct account and identify the relative share that the user has using the circulating supply. The program then splits the Epoch based stake account, "withdrawing" the relative portion, then proceeds with stake de-activation and sending the equivalent SOL to the user. The Egg is then burnt and the associated wallet is no longer required and rent exemption is returned to the user.

A visual representation of the end to end process is below:

3.3.   Staking Detail

Cosmic Eggs will have it's own validator node that it will use to host all of the stakes (tied to Eggs). Every Epoch a new stake account will be created (this account will be written to Archeio) and will comprise all the underlying shares (relative to the number minted). Every time an egg is created, a new staking account will transfer into the current Epoch stake account, which is then delegated at the end of the epoch.

For example:
If 10 Cosmic Eggs are created in Epoch 100, the aggregate Stake account from Epoch 100 will have 100 SOL. The sol will be staked on the change of epoch and the first reward will be earned on the subsequent epoch change.

On redemption, the opposite occurs; the egg is burnt and based on the egg that is sent to the program, Archeio fetches the correct epoch stake account and splits off the relevant share (using the current token supply) and then un-delegates the newly created account, this takes until completion of the epoch. The withdrawal permission is then updated to the holding account of the egg burner and once the epoch completes the SOL will be available for withdrawal.

For example:
In Epoch 105, the aggregate Stake account from Epoch 100 has 123 SOL, with 10 Epoch 100 Cosmic Eggs minted. A user redeems an egg; the share is calculated based on circulating supply and in parallel the egg is burnt (9 remain) and the relative share, 12.3 SOL in this case, is split into a new account (there is rent exemption required in this process) - the Epoch 100 Stake account now contains 110.7 SOL. At the conclusion of the epoch the user's funds are available to be transferred to the user's main Solana account.

3.4.   Eggcinerator

The Cosmic Eggs project will additionally include a yield farming opportunity which will be Solana based, enabling users to immediately redeem their Cosmic Eggs. The Eggcinerator will allow users to deposit Solana into a token pool (and hold Eggcinerator tokens or representative tokens of their share of the pool) and earn rewards any time a Cosmic Egg holder opts to redeem their Cosmic Eggs via Eggcinerator. Cosmic Egg holders receive the convenience of being able to convert their eggs back to Solana at the cost of a discount on their Solana, and giving up the current in process Epoch's reward. Yield farmers are rewarded by receiving the discount and the reward earned on the latest epoch (and the rent exemption that the redeemer pays to create an associated wallet for the Eggcinerator Pool). The discount rate at which the Cosmic Egg can be redeemed will initially be fixed, however the Orphic Finance team may adjust based on the volume of Eggs and the demand on the pool.

3.5. Eggcinerator Process

A Cosmic Egg holder may redeem their egg for Solana based on current value (from Archeio), less the discount as decided by the Eggecinerator program. Users will forfeit any fractional rewards of the current epoch.

For Example: User's cosmic egg is worth 14 SOL and the discount is currently 10%, the user can redeem their Cosmic Egg for 12.6 SOL.

Yield Farming participants join the Eggecutor program and are issued relative share of the pool through SPLs, when users redeem Cosmic Eggs, they are issued SOL from the pool and the Cosmic Eggs are redeemed through the standard Cosmic Egg workflow, which yields both the discount and 1 epoch worth of rewards for the pool. The team may choose to make the discount rate variable to incentivize or de-incentivize participants. When users would like to withdraw, they send their SPL tokens back to the Eggecutor program, which in turn will burn them and send the user back the relative value of their share (SPL/Total SPL). Withdrawing from the Eggecutor will exclude users from participating in any yet to be determined rewards. Withdrawals may need to be delayed until close of epoch if there is not enough SOL present to meet all requests.

For Example: A Yield Farming participant puts 10 SOL into the Cosmic Egg Eggecutor along with 90 SOL put in by other people. When the redeeming user from the initial example redeems, 12.6 SOL is transferred from the pool to the user (leaving the pool with 87.4 SOL). The Cosmic Egg they redeemed, is then transferred to the associated wallet of the pool and queued for redemption via the Cosmic Eggs protocol. The Cosmic Egg's value at time of redemption will be 14 SOL + rewards from current epoch. That amount is then transferred back to the Eggecutor program address, yielding a gain of 1.4 SOL + new reward for the pool users, or .14 SOL + 10% of the reward for our yield farmer.

4. Archeio - Token Registry
4.1. Introduction

Archeio works partially as a token registry and partially as a pseudo oracle. The purpose is to be able to store a series of metadata that are important to the operation of various programs and ultimately to the development of the blockchain ecosystem. The registry will enable a few usage patterns; provide a verifiable value of a specific token whether it be a semi-static (only changes on epoch) or fully dynamic (market driven), time series data will be stored for use in charting. Furthermore all data in Archeio will be immutable and any changes to entries will be public and visible to all users.

4.2.   Archeio Interactions

Users and programs can interact with Archeio via API or graphical user interface (archeio.app). For token creation a user must pay a fee to create the registry entry and additionally stake tokens to keep the record active. A token creator must provide a series of the relevant metadata and can also decorate their tokens with additional fields. Any attribute or field modifications requires paying a fee and all changes will be immutable (history of changes will exist perpetually); if a different user than the original lister is modifying the entry, collateral must be posted by the modifier. A user can additionally pay to delist (this will render the record inactive - but due to immutability it will not be fully removed) the token and the collateral will be returned to them. Any record that is delisted becomes archived and is no longer reachable via API, however it is browsable through the Archeio GUI.

4.3.   Metadata

Archeio captures metadata provided by a Token creator (or Mint Authority), which can be a user or a program. Archeio users can add their own custom fields to the token registry or they can add the predefined values in archeio (this list may be expanded over time and driven by attributes provided by users). Fields can either be static (provided via input) or calculated (semi-static or fully dynamic).

- Name - Token name
- Ticker - Token ticker symbol (most likely will have some guidance on defined length)
- Icon - an image file that is used to represent the token
- Normalized Version - if name or ticker are not unique provides a unique identifier
- Mint Address - the address of the mint for the token (this is a point of validation, if the user is not the mint authority - the record should not be created and the collateral should be returned ; the charged fee will not be returned)
- Mint State - the state on the mint (active, inactive, frozen); pollable from mint address
- Circulating supply - is a calculable value which will poll the mint address
- Value Method - this will either be epoch based or dynamic; in the situation of dynamic multiple pipelines are preferable and the team will build these to cover DeFi and CeFi options and provide weights and discount data that is standard deviations from the mean
- Historical Information - both circulating supply and total value or account value
- Mint Authority - this will double as the editor of the metadata as the same entity who can mint tokens should be the same entity who can modify the token registry entry
- Collateral Posted - the amount and current value of the collateral posted, when the record is set to be deleted this is the amount that will be returned to the account of origination.
- Account Originator - address that posted the collateral to keep the entry active
- Project URL - the link to the project that is creating the token registry entry
- Listed Location - the location, CeFi, DeFi and marketplace where the token can be interacted with
- Base borrow rate - using both CeFi and DeFi a blended baseline rate at which the asset can be borrowed

- Base lending rate - using both CeFi and DeFi a blended baseline rate at which the asset can be lent
- Tags - these will be for categorization and discoverability
- Parentage - not required, but if a group of tokens are all connected, a parent entry can be created without certain normally required fields and then one child minimum should be provided with the required fields (for example: must contain a mint)

4.4.   Dynamically Calculated Fields

In the instance of dynamically calculated fields for external prices or rates, the more integrations, the more accurate a value is possible. Adjustments via TWAP (time weighted average price) or VWAP (volume weighted average price) will be supported in further iterations. A mean price should be calculated and any data points that are 2 standard deviations from the mean should be ignored, and the mean should be recalculated. This will prevent using stale data and ultimately creating inaccurate values.

4.5.   History

For many tokens in the Solana ecosystem, it is impossible to get historical value of tokens. Archeio will provide users the ability to view values over time or be able to create actions from the Archeio API. Based on the type of token, different time keeping strategies will be employed. Archeio does not expect to provide tick history for all tokens it covers, but will instead boil them down into candles based on defined slices of time (potentially as small as 1 min and rolling upwards based on desired setting).Likewise, for all tokens that get updates on epoch change, a data point will be recorded for at the beginning of a new epoch. Users can also choose to have additional variable fields stored, such as the circulating supply.

4.6.   Authorization

Users who create or modify records should be the same users that hold the mint authority, this will prevent other users from putting erroneous data. This should be kept in mind when users transfer mint authority and would like to remove the Token Registry entry, as either the user will need to remove the entry prior to transferring Mint authority or the user will need to ask the new authority holder to modify their record and replace their posted collateral.

4.7.   Archeio Creation Process

A Token Registry entry creator, will first click create from the GUI or initiate a create via the Archeio API. The user will be asked to connect their wallet and pay a fee to create the entry (from initial creation, there will be a grace period of 3 epochs before the user is considered to be modifying the records and must re-pay the fee). The creator must then provide the relevant metadata to their project and ultimately hit submit. Next the user will be prompted to post their collateral to keep their Archeio entry active - based on a defined SOL value that is not overly

prohibitive but enough to make users carefully consider creating records (in the case of Cosmic Eggs, an entry may only be needed for the parent - Cosmic Eggs, however the team is carefully considering how to handle this situation).

When a user transfers mint authority from their control, they should either deactivate the Archeio entry ahead of time or have the transferee modify the entry or post collateral to get their collateral back, otherwise they will be beholden to either raising a governance driven vote or the whims of the new token authority holder.

When a user wants to deactivate an entry, they pay a fee, in line with the creation fee, and then receive their posted collateral back.

4.8.   Replication

One of the most important features of Archeio's design is that it is replicated across multi-regions and cloud providers in order to provide resiliency from outages, as it will be core to providing the necessary functionality to operate the Orphic Finance ecosystem. In the event of geographically localized or single cloud partner disruption, Archeio will be able to route requests to one of the other nodes on it's network (this may cause some latency issues, but not down time).

4.9.   Future Considerations

Archeio is considering a series of add-ons to continue to push Solana ecosystem adoption and the ones that we have actively discussed our captured below:

4.9.1.   Archeio Node Operation

Users can run a node of the Archeio database and receive OFIN tokens as a reward for doing so, alternatively the team is also considering distributing a portion of the fees generated in SOL. Decentralizing the Token Registry would reduce the risk of DDoS attacks or other malicious activities intent on disruption.

4.9.2.   Token Creation

Enable non technical users to create tokens on Solana using a UI and connecting their wallet. The token creation program will maintain authority over the mint and can pass on functionality to the UI. This alone is a very low hanging fruit and will allow direct entry into Archeio. Users will be able to leverage the Token Sale Framework.

4.9.3.   Token Sale Framework

Support either fixed rate price (set value of token and sell defined amount) or variable price (set token amount and raise into escrow wallet to determine price). Multiple distribution patterns could be supported, including different lock-in periods or discounts related to certain conditions. We can also support conditional requirements, such as a soft cap (cancel auction and send back to users if threshold not met), reduction in time of raising if caps met (example: at 10,000

SOL reduce time by 1 week), hard caps (instant end). Users will be able to choose between representational tokens (raise without defined governance token) or governance tokens, both have been added to the roadmap for Archeio.

4.9.4.   Ticker Name Reservation

Much like a DNS service, users can reserve tickers for token / projects. Users will be able to pay in Solana, OFIN token to reserve a name and post collateral to keep it active. This will be paired with Protogonos to allow users to sell names to one another and can be released as auctions as well.

5.   Protogonos Marketplace
5.1.   Introduction

The Protogonos Marketplace allows users to create a series of different ways to sell tokens. It will be fully integrated with Archeio thus allowing for reward bearing tems to be sold and bid on via relative price. This includes also providing the ability to create direct listings, entertain offers and create time and epoch based auctions. More detail to follow on Protogonos as we see how things develop.

5.2.   Integrations

Protogonos will baseline integrate with both Archeio and the native Solana token registry. For auctions based on Archeio, token sellers will be able to dynamically update the value of their token based on the value definition set - enabling users to sell a token that grows in relative value based on a % over or under value.

6.   OFIN Tokenomics
6.1.   Introduction

The OFIN token is the native token of the Orphic Finance ecosystem. It is a governance based token with a % of the fees from all the Orphic Finance services being used to buy on a randomized basis, creating perpetual buy pressure on the token and incentivizing users to continue to hold. The wallets where the community holding fees are generated will be posted on the Orphic Finance site, and the protocol will randomly purchase the OFIN token in an unpredictable manner, to prevent spikes and crashes in the OFIN token price. Changes to the Orphic Finance ecosystem will be added as voting based and all OFIN token holders will be able to vote on topics based on their token holdings at time of vote.

6.2.   Buy and Burn

A % of the funds generated from the Orphic Finance ecosystem will be posted into a public account and used to buy and burn OFIN tokens. The activities in the Orphic Finance ecosystem that are expected to generate funds that go to the public account are:
1. Cosmic Eggs operation fees (staking fees)
2. Eggcinerator fees (LP fees)
3. Archeio fees (LP fees)
   a. Listing fees
   b. Token Creation fees
   c. ICO handling fees
   d. Ticker Name Service fees
4. Protogonos auction fees
5. Orphic Finance (LP fees)

The percentage that will be used to buy and burn the OFIN tokens will likely be calculated net to the operating cost of servers and validators for the ecosystem. We expect that the bulk majority (~75+%) of all funds generated will go back to the Orphic Finance token holders.

6.3.   Distributions

Although we have not finalized our plans for the OFIN token system; we do plan on distributing a percentage of tokens to users of the ecosystem in the form of rewards. This includes all LP activities, holders of cosmic eggs, holders of OFIN tokens, Archeio users / node operators and Protogonos users.
We roughly expect the following distributions:
- 5% to Advisors
- 30% to Orphic Finance Founders Pool
- 15% to Orphic Finance Foundation
- 40% to Public ICO
- 10% to Market Making / Listings

6.4.   OFIN Token Sale

The Orphic Finance team is currently planning a sale of the initial OFIN token sometime between June 2021 and July 2021. This token will likely be able to be swapped for the full fledged OFIN Governance Token at a later date, as the Governance features in Solana are all fairly undeveloped. We will post updates in the near future through our social sites, discord and the Orphic Finance website.

7.   Summary

This whitepaper is intended to cover the Orphic Finance ecosystem at a high-level, based on a number of assumptions the Orphic Finance team has made and what is known about the underlying Solana ecosystem. As both are currently in active development, this paper will likely become out of date fairly quickly and should be used as a theoretical guide of how the Orphic Finance team wants to implement its' services.