Maze Whitepaper: The Super Sovereign Banking Protocol Crossing All Blockchains

Maze

lunas@mazeprotocol.com

May 2021

Abstract

Maze is a non-custodial and over-collateralized banking protocol initiated from the non-interest loan service. It originally created an algorithm, Stratified Harvest Regulating, to replace the classical interest model of money markets, and thus obtained a highly dynamic yield-rate consensus mechanism. Throughout the implementation, an irreconcilable problem was discovered on the current Layer-1 blockchains. Therefore, a decentralized income distribution computing service has been specially developed, which is the innovative transparent computation middleware protocol, Farmbase. Farmbase has turned cross-chain mining pool merging into reality, allowing Maze to build branches with synchronized yield-rates on any blockchain. A particular mutation of Farmbase - Farmbase Hawala Thread, collaborates with Maze to build a drawing right transfer protocol of cross-chain assets, so that users can almost unlimitedly save or retrieve funds on different networks. Last but not least, the combination of Maze's lending protocol and its Stratified Harvest Regulating algorithm has brought the dawn of a new decentralized stablecoin, ZUSD, with blended strengths.

The whitepaper will try to introduce Maze in an abstract and linear way - the principles, the implementation ways of critical components, the relations of sub-systems and the potential influence of a super sovereign banking protocol crossing all blockchains.

Maze is a financial experiment standing above the giant's shoulder. We appreciate Maker, Compound, Aave's efforts to create the core infrastructures of decentralized finance.

Contents

1 Maze Core Protocol: Zero-Interest Lending System

1.1 Maze Protocol Structure Overview

1.2 Delegation Tokens

1.3 Working Flows

2 Maze Tokendynamics: Stratified Harvest Regulating

2.1 The Capital Market Driven by Yield-Rate Contrast

2.2 Maze Offering Distribution Algorithm

2.3 Objective Effects

2.4 Difficulties of Implementation

3 Implementing The First Dynamic Farming Pool: Farmbase Middleware Protocol

3.1 Foremost Targets

3.2 Introducing Farmbase

3.3 Farmbase-Maze Collaboration

3.4 The Abilities of Farmbase

4 Crosschain Synchronized MP

4.1 Farmbase Degeneracy of Cross-chain Identical MP

4.2 Maze Branches on All Blockchains

5 Implementing Drawing Right Transfer Protocol of Cross-chain Assets: Farmbase Hawala Thread

5.1 Hawala: Transfer of Drawing Right Instead of Assets

5.2 Reserve Pools: High-Liquidity Drawing Right Pools Distributed on All Blockchains

5.3 Farmbase Hawala Thread

5.4 The Asset Stargates Connecting All Metaverses

6 A Stablecoin Combining CDP & Algorithm

6.1 Forging ZUSD with CDP

6.2 ZUSD-MAZE Tokendynamics under SHR

7 Conclusion: A Super Sovereign Banking Protocol Crossing All Blockchains

8 Maze's Experimental Network - Neko

8.1 Overview

8.2 Neko V1 Function Configurations

8.3 Parameters

8.4 The Ignition Procedure

9 Glossary

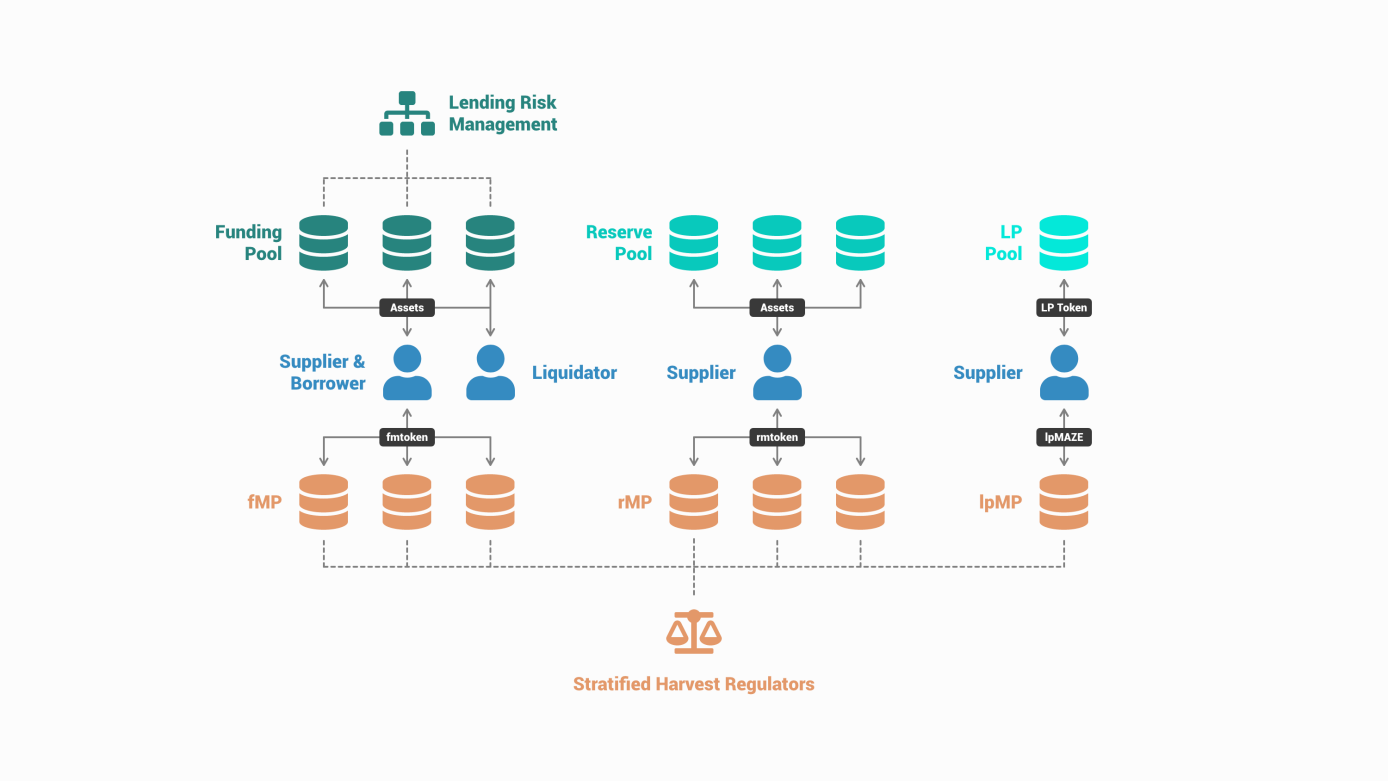
1 Maze Core Protocol: Zero-Interest Lending System

This chapter introduces the principle of Maze Protocol's functioning, including a set of asset pools for various functionalities, a structured mining pool, a typical decentralized lending risk management system, a group of newly designed delegation tokens, and the brief explanation of key working flows.

1.1 Maze Protocol Structure Overview

Since Maze does not have an interest spread system, suppliers do not get direct income by putting assets into asset pools. The delegation tokens generated from the deposits must be staked into the corresponding mining pools in order to generate pool-distributed rewards for suppliers. In fact, the asset pools and mining pools are decoupled, while the delegation tokens rebuild the economic connection between asset pools and mining pools - this is the congenital self-composability of Maze.

Below is the simplified sketch of Maze core protocol:



We can see from the sketch that Maze is an extension from an Aave-like typical lending protocol. From the view of technical implementation, Maze can be concluded as a non-custodial, synthetic system assembled by a zero-stable-rate Aave, a group of extra asset pools which are not involved in lending services, and a distinctive mining pool.

1.1.1 Asset Pools

Asset Pools transfer and store users' underlying assets. They are delegation-token-oriented smart contracts. All four behaviors (supply, withdraw, borrow, repay) regarding the pools must mint or burn corresponding delegation tokens.

Funding Pool

Every protocol-permitted asset is configured with a Funding Pool.

Funding Pools are a type of asset pools involved in over-collateralized lending businesses. The assets stored in the pools are:

- Automatically taken as the liquidity of the lending system, allowed to be borrowed by other users.
- Automatically taken as the collaterals of the accounts, changing the borrow credits.

Reserve Pool

Every protocol-permitted asset is configured with a Reserve Pool.

Reserve Pools are a type of safe saving pools. The assets stored in the pools are:

- Not taken as the liquidity of the lending system.
- Not taken as the collaterals of the accounts, nor changing the borrow credits.

LP Pool

LP Pool is a safe saving pool accepting only MAZE-LP Token. It is regarded as a special reserve pool.

1.1.2 Mining/Farming Pools (MP)

Each Funding Pool, Reserve Pool and LP Pool is set up with a corresponding MP. MP collects users' farming delegation tokens as the basis of shares and thus distributes income for users via Stratified Harvest Regulating algorithm.

For convenience, the MP of Funding Pool will be marked as fMP, Reserve Pool as rMP, and LP Pool as lpMP.

1.1.3 Lending Risk Management

Maze's lending protocol module continues to use the over-collateralized lending model, so it has an identical lending risk management system like Aave. It applies Loan To Value (LTV), Liquidation Threshold, Liquidation Penalty and Health Factor, together with a self-constructed oracle accessing on-chain price feeding.

Loan To Value (LTV)

The Loan to Value (LTV) ratio defines the maximum amount of currency that can be borrowed with a specific collateral. For instance, if LTV is 75%, the $100 worth of collateral can support up to $75 worth of loan.

Then the borrow credit of an account is calculated as:

$$V_{borrowcredit} = \sum_{i=1}^{n}(V_{collateral(i)} * LTV_{(i)})$$

When any collateral's price changes, the the borrow credit of the account is changed.

Liquidation Threshold

The Liquidation Threshold is the percentage at which a loan is defined as under-collateralized. For example, a Liquidation threshold of 80% means that if the value rises above 80% of the collateral, the loan could be liquidated.

Liquidation Penalty

The Liquidation Penalty is the ratio of the bonus that a liquidator can take from the collateral during liquidation execution.

Health Factor

The Health Factor is to examine if an account should be liquidated as a whole. As there is no interest, the equation is simpler. It calculates this way:

$$R_{healthfactor} = (\sum_{i=1}^{n}(V_{collateral(i)} * R_{liqthreshold(i)}))/V_{borrowedbalance}$$

When the Health Factor is under 1, the account could be liquidated.

Oracle

Maze branches will use the local blockchain's main DEX price reports as references. The report is using TWAP.

1.2 Delegation Tokens

Compared with Compound's ctoken and Aave's atoken, the delegation token system of Maze is more complicated. An equal amount of mirrored delegation tokens will generate when any underlying asset is supplied into an asset pool, while the protocol must burn the equal mirrored delegation tokens if a user withdraws an underlying asset. The mirrored delegation tokens have their own functions and characters.

1.2.1 Mirrored Delegation Tokens

Mirrored delegation tokens consist of asset delegation tokens and farming delegation tokens. Asset delegation tokens include rtoken, ftoken and pMAZE. Farming delegation tokens include rmtoken, fmtoken and lpMAZE.

- When a Funding Pool accepts an underlying asset, it will mint 1:1-mapping ftokens and 1:1-mapping rmtokens.
- When a Reserve Pool accepts an underlying asset, it will mint 1:1-mapping rtokens and 1:1-mapping rmtokens.
- When the LP Pool accept MAZE-LP Token, it will mint 1:1-mapping pMAZE and 1:1-mapping lpMAZE.

Please note that, as Maze has no interest spread system, the delegation tokens are not the interest derivative tokens of Maze.

Asset Delegation Tokens

The Maze front-end application can identify accounts' deposit balance based on asset delegation tokens. All rtoken and pMAZE are ERC-20 standard tokens, allowed to be transferred out of accounts. However, we've disabled the transfer function of ftokens so that users cannot move them, as the lending system must prevent the statistical data related to borrow credits from occurring faults.

Farming Delegation Tokens

Farming delegation tokens are accepted by their corresponding MPs. Their relation is:

Delegation Token	Target MP
rmtoken	rMP
fmtoken	fMP
lpMAZE	lpMP

For example, rmMAZE is only allowed in MAZE-rMP.

All farming delegation tokens are ERC-20 standard tokens, allowed to be transferred, meaning that a farming account might not be the account depositing funds in asset pools.

1.2.2 Debt Token

Once a borrower makes a loan, the protocol will mint an equal amount of debt tokens. Like ftokens, debt tokens do not have the transfer function, so that the users cannot move them. They will reflect the debt status of users in Maze's front-end application. The protocol will burn the equal debt tokens after the borrower repays the loan.

1.3 Working Flows

The working flows of the Maze product is not complicated, similar to the common lending protocols.

1.3.1 Supply & Withdraw Funding Assets

When a user supply funds to Funding Pools, the account's borrow credit will rise, and the borrow limit used will fall, so that the liquidation risk is lower. In contrast, if the user withdraws funds form Funding Pools, the borrow credit will descend, but the borrow limit will go up, making the liquidation risk rises. The limitation of a user's withdrawal from a Funding Pool is depending on:

- The amount of ftoken and fmtoken in the wallet.
- The health factor, which should not be under 1.
- The available liquidity of the Funding Pool itself.

The price change of an asset will impact the account's borrow credit and the borrow limit used in real time.

1.3.2 Supply & Withdraw Reserve Assets

Since Reserve Pools do not take part in the lending services, users' supply and withdraw movements here will not affect their lending account. The limitation of a user's withdrawal from a Reserve Pool is depending on:

- The amount of rtoken and rmtoken in the wallet.
- The available liquidity of the Reserve Pool itself.

1.3.3 Borrow & Repay Assets

A user must supply assets in any Funding Pool to acquire the borrow credit before borrowing any assets. Getting or repaying loans will change his borrow limit used and liquidation risks. A user's limitation of acquiring loans from the protocol is depending on:

- The account's borrow credit and limit used.
- The available liquidity of the Funding Pool itself.

1.3.4 Liquidation

If the Health Factor of an account is under 1, it will face liquidation. The liquidation is gradual. The liquidator can repay up to 50% of loans in one liquidation incident, aiming to recover the Health Factor back to over 1. The liquidator can take some extra collateral as bonus based on the liquidation penalty parameter.

Once the liquidation occurs, the ftokens pointing to the liquidated collaterals and the debt tokens pointing to the repaid debts will be burnt by the contract. However, as the fmtokens can be transferred, unable to be burnt automatically by the contract, they are to be repaid manually by the borrowers. Before repaying fmtokens, the account's lending service functions will be suspended.

1.3.5 Earn Incentive for Asset Supply (Farming)

The underlying asset deposits are not bringing any direct reward for the suppliers. Users have to stake their farming delegation tokens into MPs accordingly to participate the distribution of MAZE. Based on the parameter of Stratified Harvest Regulators, the reward is distributed in all MPs at a 10-minute interval.

2 Maze Tokendynamics: Stratified Harvest Regulating

In Maze's money market, the traditional interest spread model has been replaced completely by an incentive distribution mechanism unilateral for asset suppliers. It leaves a new question in front of us: what algorithm should we use to distribute incentives for different assets' suppliers on the protocol, in order to realize the adjustment power of the hundred-year-old interest system, even surpassing?

2.1 The Capital Market Driven by Yield-Rate Contrast

During the hundred years of banking industry development, the interest spread has always been the fundamental business model which every bank relies on. On lending markets, the interest system's general tasks can be concluded to two major ones:

- First, it helps to transfer renting costs of money from borrowers to lenders.
- Second, it adjusts the liquidity supply and demand in each asset pools.

If the occupation of an asset goes intensive, the funding rate will rise. The lifting rate guides suppliers to provide more funds in order to make more earnings, or guides borrowers to reduce the occupation, otherwise paying more costs. Beside driving the supply and demand inside a single asset pool, the interest rates can create competitions among different assets. When rates change across multiple assets, suppliers

will shift to higher-rate assets, while borrowers will shift to lower-rate assets. At last, a money market also faces competitions against external interest rates. If the external market is more attractive to suppliers, more local capital will migrate out, and if more attractive to borrowers, more local capital will be released.

To the supply and demand sides of capital, the interest rate is the yield rate - suppliers get positive yield rate, while borrowers get the negative. The entire capital market's participants are comparing every option's yield rate all the time to make their asset management strategies. The risk-free rate is the origin of this comparison, which forms a huge contrast list with other types of yield rates. On specific risk levels, capitals are always seeking the best yield-rate offers.

From the cross-section view, on Maze, since no loan-interest charge exists, the yield rates for borrowers are no longer negative, indicating that they have eternal drive to occupy funds (they can use non-cost assets to seek further rate of returns). The questions are now focusing on the supply side:

- Inside one certain asset type, if the occupation goes intensive, how to attract more supply of funds in?
- Inside one certain money market, if one asset is much popular, while another one is not, how to adjust the distribution of supply to mitigate the liquidity imbalance?
- In the whole economy's yield-rate contrast competition, how to ensure the long-term attractiveness of the local money market's overall rate of return?

While we investigate these questions and the yield-rate contrast driven nature of capital markets, there is one thing for sure: a new capital supply drive algorithm's target is to realize a rational and differentiated yield-rate guiding mechanism. Starting from this thinking, Maze has provided a Stratified Harvest Regulating algorithm, to let codes execute transparent and highly guide-capable incentive distributions based on all users' current asset management strategies.

2.2 Maze Offering Distribution Algorithm

MAZE is the primary token of the protocol, without hard cap designed. It's mainly generated via the periodical farming minting controlled by Stratified Harvest Regulators (shortened as SHR).

2.2.1 The Principle

SHR can be imagined as a tap-water system. The initial mint contract is the central water plant, which sends water at the permitted largest current to the first-class pump station. The pump station allocates water current to its lower-class tube branches according to its distribution parameters, and there are several layers of such pump stations and branches. The ending of the tube system is the personal water tank of each farmer. It is to be noted that all pump stations' parameters will shift at each cycle, and the shifting is mostly based on the devote in everyone's water tank.

Value As Share

During the statistical procedures, SHR takes supply value other than just amount as mining share. Doing so is to enable the share comparison across asset types. The supply values are counted in USD in realization. The permitted stablecoins are marked constantly at $1.

Cycle

Different from common mining pools, Maze's reward generation interval is not one block. SHR's minimal computing period is 10 minutes (e.g., 40 blocks on Ethereum, 200 blocks on BSC).

The rule of cycle raises the extra concern on the validity of farming share registration. SHR requests every farming share to endure a complete cycle before being valid for the cycle's reward distribution rights. It leads to the following rules:

- Stake: T+1 effectiveness

  The current cycle of delegation token staking transaction does not offer farming share for this cycle T, and the token will be on a waiting sequence. The share is accepted from the cycle T+1.

- Unstake: T+0 effectiveness

  The current cycle of delegation token unstaking transaction revokes the farming share, as the share is not 'attending'.

At the end of every cycle, SHR will take a snapshot for the valid share and price feeding data of the cycle, and thus calculate all farmers' income on this cycle accordingly.

Global Yield Baseline

Global Yield Baseline is the baseline amount of MAZE that the mint contract can issue on each cycle.

Global Yield Speed Factor

Global Yield Speed Factor is the ratio between baseline and actual production. It is for controlling the production cutdown of MAZE.

Global Yield Distribution

Global Yield Distribution is the ratios of MAZE distribution toward the first-class logical structure from the mint contract. The first-class logical structure consists of three main partitions, including

- United Supply Pool (USP)

  USP contains a set of permitted stablecoin asset pools' corresponding rMP and fMP, and MAZE asset pool's MAZE-rMP and MAZE-fMP.

- Instablecoin Supply Pool (ISP)

  ISP contains all permitted instablecoin asset pools' corresponding rMP and fMP.

- LP Pool (LPP)

  LPP is lpMP.

In the practical product, we use nicknames to call the main partitions' corresponding asset pools. They are Tabby Cattery to USP, Siamese Cattery to ISP, Ragdoll Cattery to LPP.

Supplied Value Ratio Balancing (SVRB)

SVRB is in charge of the ratios of MAZE yield distributed from USP toward stablecoin MPs and MAZE MPs. SVRB is a function, where R_svrb represents the ratio of MAZE yield distributed from USP toward MAZE MPs, and P_maze represents the proportion of MAZE supplied value in the entire USP.

Below is the function:

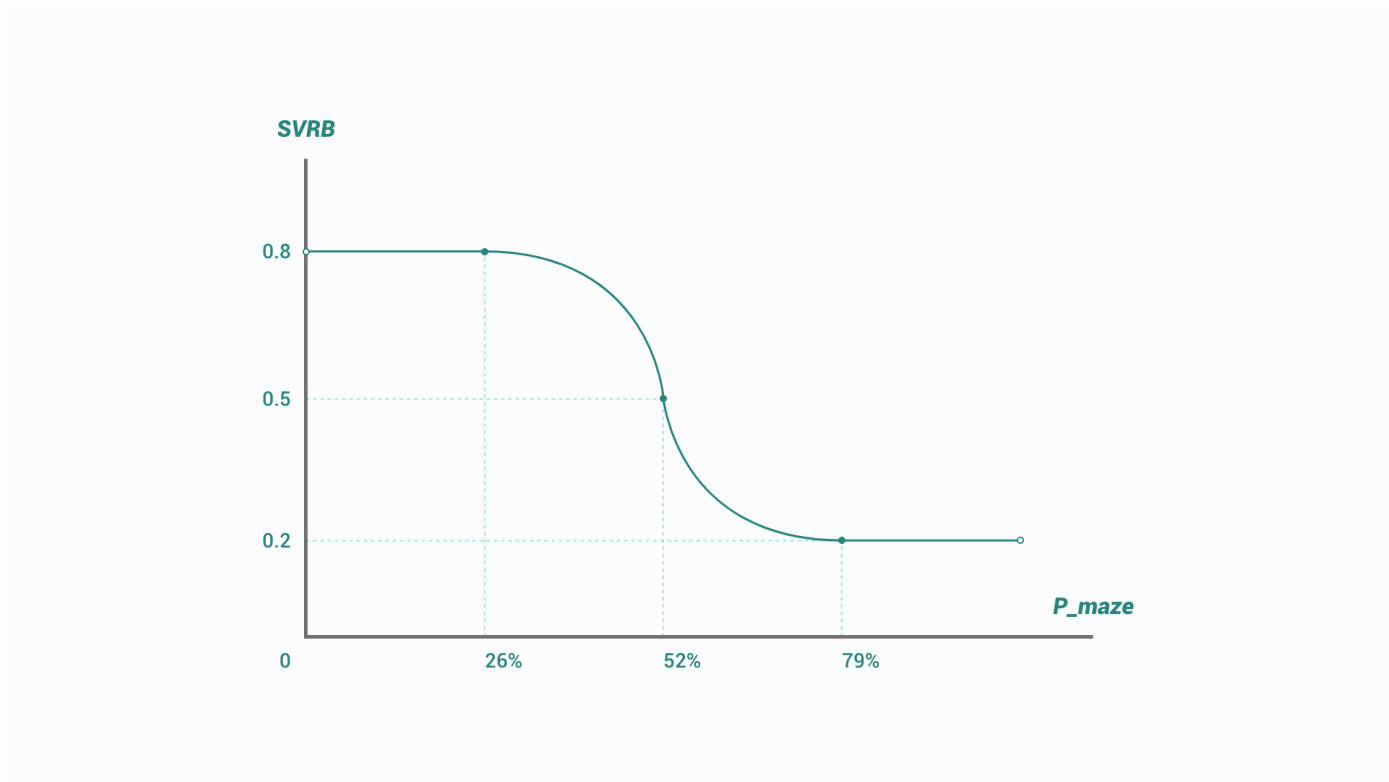
$$R_{svrb} = 0.8 \qquad P_{maze} \in (0, \pi/12)$$

$$R_{svrb} = 0.3 \sin 6x + 0.5 \qquad P_{maze} \in [\pi/12, \pi/4]$$

$$R_{svrb} = 0.2 \qquad P_{maze} \in (\pi/4, 1)$$

Please note: the regulating mechanism is not yet tested in real application. During the product operating, the function could iterate.

The function's curve:



As we can see intuitively, under the regulating of SVRB, when the value of MAZE drops, the yield rate will lean to MAZE MPs and vice versa.

Simple Value Weighted Distribution (SVWD)

SVWD means that, among all same-class asset types, MAZE yield is distributed according to the weights of the different assets' supplied values inside the class.

For example, in the stablecoin MP structure in USP, there are USDT's MPs and BUSD's MPs. In the cycle T, it is known that: USDT's supplied value is $300,000,000 and BUSD's supplied value is $300,000,000, so USDT's MPs should get 3/5 of MAZE yield and USDT's MPs should get 2/5. Among the MPs in ISP, they follow the same weighted rule.

SVWD makes all asset suppliers unified to accept their assets' USD values as the basis of yield-rate contrast.

Funding Occupation Rate Balancing (FORB)

FORB is in charge of the ratios of MAZE yield distributed between an rMP and an fMP of each asset type. It means that there is a FORB computation unit in each asset's MPs. FORB is a function, where R_forb(i) represents the ratio of an asset(i)'s MAZE yield distributed to fMP(i), and P_occupy(i) represents the loan occupation rate of the Funding Pool(i).

Below is the function:

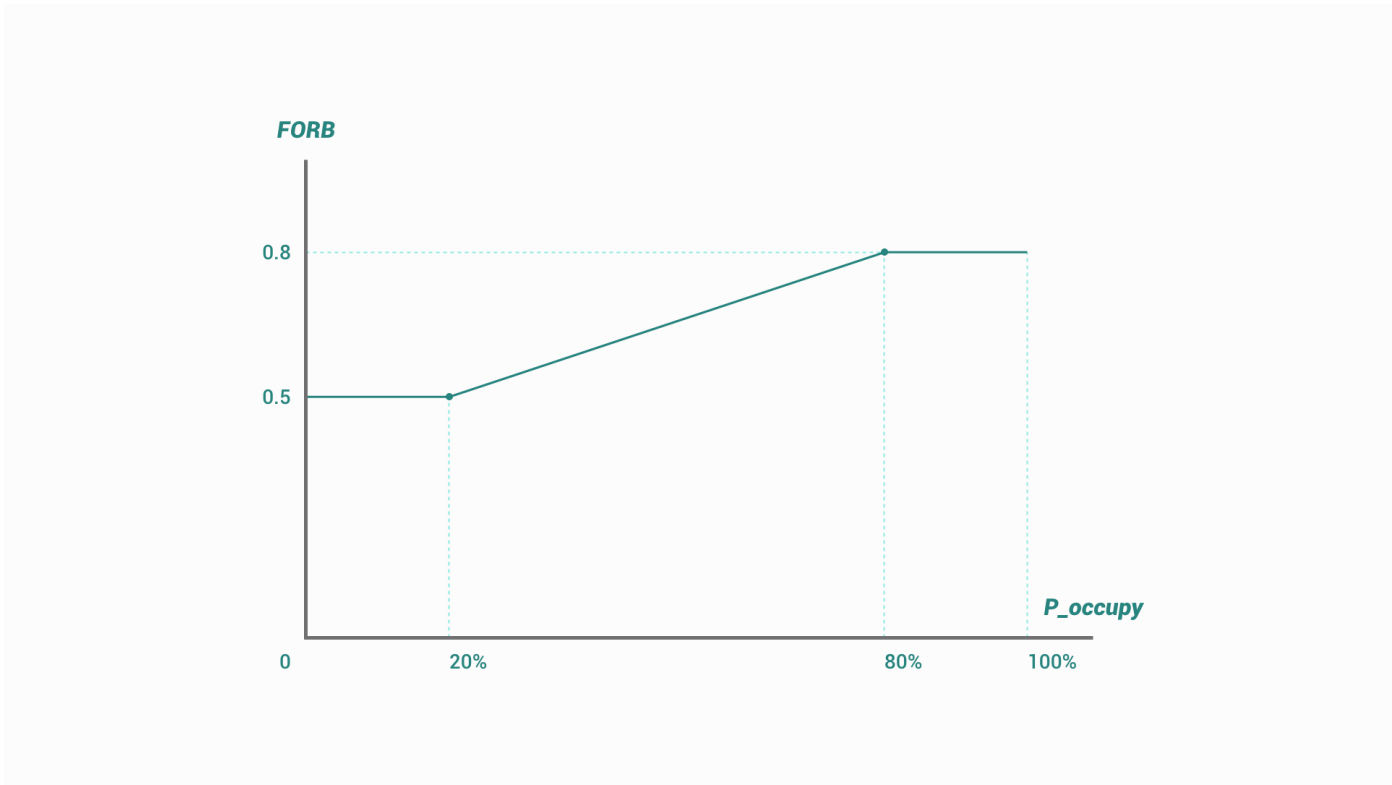
$$R_{forb(i)} = 0.5 \qquad P_{occupy(i)} \in [0, 0.2]$$

$$R_{forb(i)} = 0.5 * P_{occupy(i)} + 0.4 \qquad P_{occupy(i)} \in (0.2, 0.8)$$

$$R_{forb(i)} = 0.8 \qquad P_{occupy(i)} \in [0.8, 1]$$

Please note: the regulating mechanism is not yet tested in real application. During the product operating, the function could iterate.

The function's curve:



As we can see intuitively, under the regulating of FORB, when the occupation rate of the Funding Pool rises, the yield rate will lean to its fMP and vice versa.

Besides, as no occupation rate is in lpMP, no FORB computation unit is needed by it.

FORB Exception

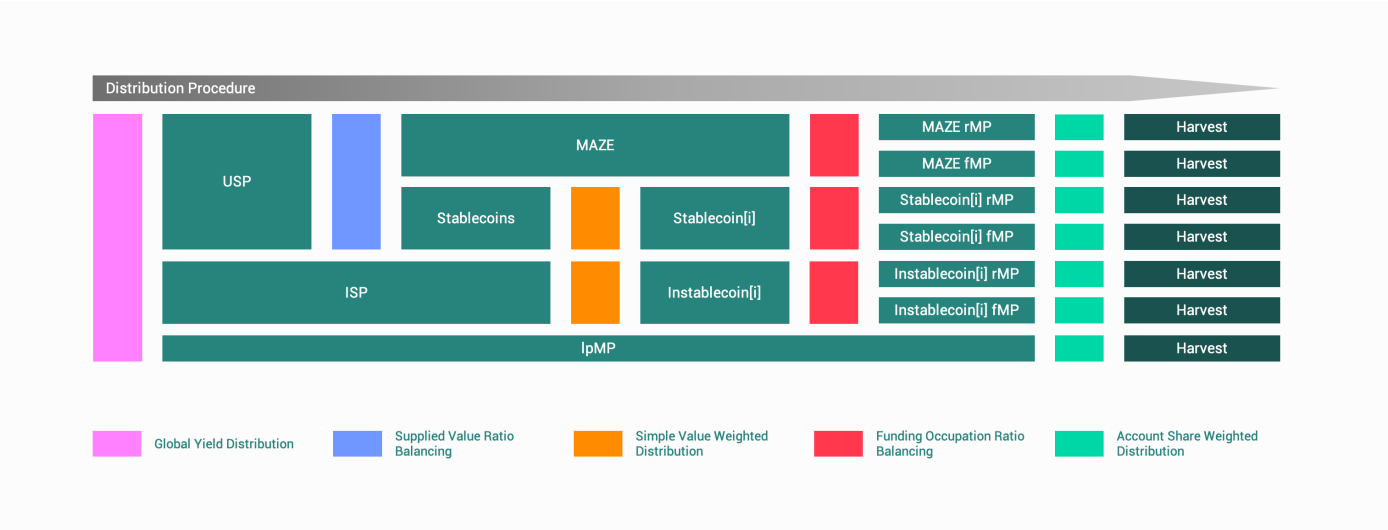
Considering an extreme condition: if there is no supply in an asset's Funding Pool or Reserve Pool, the MP's FORB computation unit will be ineffective. The MP's yield will all be taken by the supplied side.

Account Share Weighted Distribution (ASWD)

ASWD is the bottom-class distribution, which is the common way of all mining pools' contracts. It is to eventually distribute the rewards according to the proportions of the staking delegation tokens in the MP by the accounts, when the MAZE yield is allocated in a specific rMP, fMP or lpMP.

2.2.2 SHR Visualization

The chart below reveals SHR's way of passing yield from left to right.



2.2.3 Modeling Farming Computation System

According to the principle of SHR, the nature of MAZE distribution mechanism is periodically computing the yield reaching each rMP, fMP and lpMP, and finally distribute reward for every farmer using ASWD. It is equivalent to ask for the objective functions of calculating the MPs' yield.

Presets

- Cycle

  One cycle is 10 minutes. SHR collects the status of all MPs in cycles.

- Validity Of Shares

  Only the full-cycle endured farming shares are the valid shares.

Parameters

- Global Yield Baseline

  The first parameter is the global MAZE yield speed's baseline, which is the root of the farming protocol. It is a protocol-preset goverance parameter.

Variable	Source	Range	Definition
Y_baseline	Protocol Preset	500	Global yield velocity of MAZE on baseline per cycle

- Global Yield Speed Factor

  It is a factor to control the global actual yield speed.

Variable	Source	Range	Definition
K_global	Protocol Preset	0-1	Speed Factor for production adjustment

- Global Yield Distribution

  It is a set of ratios to pass incentive to three main partitions from the global yield.

Variable	Source	Range	Definition
R_usp	=1-(R_lpp+R_isp)	-	Global yield distribution ratio for USP
R_isp	=1-(R_lpp+R_usp)	-	Global yield distribution ratio for ISP
R_lpp	Protocol Preset	0.05	Global yield distribution ratio for LPP

Initial Variables

- Price Feeding

  The price feeding of farming computation system is inherited from the one of the lending protocol module.

Variable	Source	Definition
C_maze	Price Feeding	Arithmetic mean price of MAZE in the cycle
C_instable(i)	Price Feeding	Arithmetic mean price of the instablecoin in the cycle
C_stable(i)	Fixed	Value of USD

- Token Amounts

  They are the statistics of the delegation tokens in MPs and the issued tokens, taken from Reserve Data.

Variable	Source
Staking Tokens	
N_mazereserve	Total **rmMAZE** in rMP
N_mazefunding	Total **fmMAZE** in fMP
N_stablereserve(i)	Total **rmstable**(i) in rMP
N_stablefunding(i)	Total **fmstable**(i) in rMP
N_instablereserve(i)	Total **rminstable**(i) in rMP
N_instablefunding(i)	Total **fminstable**(i) in fMP
N_lpstake	Total **lpMAZE** in lpMP
Issued Tokens	
N_rmaze	Total **rMAZE** in circulation
N_fmaze	Total **fMAZE** in circulation
N_rstable(i)	Total **rstable**(i) in circulation
N_fstable(i)	Total **fstable**(i) in circulation
N_rinstable(i)	Total **rinstable**(i) in circulation
N_finstable(i)	Total **finstable**(i) in circulation
N_mazeoccupy	Total **debtMAZE** in circulation
N_stableoccupy(i)	Total **debtstable**(i) in circulation
N_instableoccupy(i)	Total **debtinstable**(i) in circulation

Computation Process

- Phase 1 Value

Variable	Definition
V_maze	Value of MAZE supplied in USP
V_stable(i)	Value of one type of stablecoin supplied
V_stable	Value of stablecoins supplied in USP
V_usp	Value of assets supplied in USP
V_instable(i)	Value of one type of instablecoin supplied
V_instable	Value of assets supplied in ISP

$$V_{maze} = (N_{rmaze} + N_{fmaze}) * C_{maze}$$
$$V_{stable(i)} = (N_{rstable(i)} + N_{fstable(i)}) * C_{stable(i)}$$
$$V_{stable} = \sum_{i=1}^{n} V_{stable(i)}$$
$$V_{usp} = V_{maze} + V_{stable}$$
$$V_{instable(i)} = (N_{rstable(i)} + N_{fstable(i)}) * C_{instable(i)}$$
$$V_{isp} = \sum_{i=1}^{n} V_{instable(i)}$$

- Phase 2 Global Yield Distribution

Variable	Definition
Y_global	The actual amount of MAZE generated per cycle

$$Y_{global} = Y_{baseline} * K_{global}$$
$$Y_{usp} = Y_{global} * R_{usp}$$
$$Y_{isp} = Y_{global} * R_{isp}$$
$$Y_{lpp} = Y_{global} * R_{lpp}$$

- Phase 3 USP Internal Distribution
  - Phase 3.1 SVRB Stablecoin-MAZE

Variable	Definition
P_maze	Proportion of MAZE value supplied in USP
R_svrb	SVRB ratio
D_maze	USP reward distributed to MAZE supply
D_stable	USP reward distributed to stablecoin supply

$$P_{maze} = V_{maze}/V_{usp}$$
$$R_{svrb} = 0.8 \qquad P_{maze} \in (0, \pi/12)$$
$$R_{svrb} = 0.3 \sin 6x + 0.5 \qquad P_{maze} \in [\pi/12, \pi/4]$$
$$R_{svrb} = 0.2 \qquad P_{maze} \in (\pi/4, 1)$$
$$D_{maze} = Y_{usp} * R_{svrb}$$
$$D_{stable} = Y_{usp} * (1 - R_{svrb})$$

    - Phase 3.1.1 FORB MAZE-rMP/fMP

Variable	Definition
P_mazeoccupy	Proportion of MAZE lent
R_forbmaze	MAZE FORB ratio
D_mazefunding	Reward distribution to MAZE funding supply
D_mazereserve	Reward distribution to MAZE reserve supply

$$P_{mazeoccupy} = N_{mazeoccupy}/N_{fmaze}$$
$$R_{forbmaze} = 0.5 \qquad P_{mazeoccupy} \in [0, 0.2]$$
$$R_{forbmaze} = 0.5 * P_{mazeoccupy} + 0.4 \qquad P_{mazeoccupy} \in (0.2, 0.8)$$
$$R_{forbmaze} = 0.8 \qquad P_{mazeoccupy} \in [0.8, 1]$$
$$D_{mazefunding} = D_{maze} * R_{forbmaze}$$
$$D_{mazereserve} = D_{maze} * (1 - R_{forbmaze})$$

- Phase 3.2 SVWD Stablecoin

Variable	Definition
P_stablevalue(i)	Value proportion of one type of stablecoin supply
D_stable(i)	Reward distribution to one type of stablecoin supply

$$P_{stablevalue(i)} = V_{stable(i)}/V_{stable}$$
$$D_{stable(i)} = D_{stable} * P_{stablevalue(i)}$$

  - Phase 3.2.1 FORB Stablecoin-rMP/fMP

Variable	Definition
P_stableoccupy(i)	Proportion of the stablecoin lent
R_forbstable(i)	The stablecoin FORB ratio
D_stablefunding(i)	Reward distribution to the stablecoin funding supply
D_stablereserve(i)	Reward distribution to the stablecoin reserve supply

$$P_{stableoccupy(i)} = N_{stableoccupy(i)}/N_{fstable(i)}$$
$$R_{forbstable(i)} = 0.5 \qquad P_{stableoccupy(i)} \in [0, 0.2]$$
$$R_{forbstable(i)} = 0.5 * P_{stableoccupy(i)} + 0.4 \qquad P_{stableoccupy(i)} \in (0.2, 0.8)$$
$$R_{forbstable(i)} = 0.8 \qquad P_{stableoccupy(i)} \in [0.8, 1]$$
$$D_{stablefunding(i)} = D_{stable(i)} * R_{forbstable(i)}$$
$$D_{stablereserve(i)} = D_{stable(i)} * (1 - R_{forbstable(i)})$$

- Phase 4 ISP Internal Distribution
  - Phase 4.1 SVWD Instablecoin

Variable	Definition
P_instablevalue(i)	Value proportion of one type of instablecoin supply
D_instable(i)	Reward distribution to one type of instablecoin supply

$$P_{instablevalue(i)} = V_{instable(i)} / V_{isp}$$
$$D_{instable(i)} = Y_{isp} * P_{instablevalue(i)}$$

  - Phase 4.2 FORB Instablecoin-rMP/fMP

Variable	Definition
P_instableoccupy(i)	Proportion of the instablecoin lent
R_forbinstable(i)	The instablecoin FORB ratio
D_instablefunding(i)	Reward distribution to the instablecoin funding supply
D_instablereserve(i)	Reward distribution to the instablecoin reserve supply

$$P_{instableoccupy(i)} = N_{instableoccupy(i)} / N_{finstable(i)}$$
$$R_{forbinstable(i)} = 0.5 \qquad P_{instableoccupy(i)} \in [0, 0.2]$$
$$R_{forbinstable(i)} = 0.5 * P_{instableoccupy(i)} + 0.4 \qquad P_{instableoccupy(i)} \in (0.2, 0.8)$$
$$R_{forbinstable(i)} = 0.8 \qquad P_{instableoccupy(i)} \in [0.8, 1]$$
$$D_{instablefunding(i)} = D_{instable(i)} * R_{forbinstable(i)}$$
$$D_{instablereserve(i)} = D_{instable(i)} * (1 - R_{forbinstable(i)})$$

Objective Functions

$$D_{mazefunding} = D_{maze} * R_{forbmaze}$$
$$D_{mazereserve} = D_{maze} * (1 - R_{forbmaze})$$
$$D_{stablefunding(i)} = D_{stable(i)} * R_{forbstable(i)}$$
$$D_{stablereserve(i)} = D_{stable(i)} * (1 - R_{forbstable(i)})$$
$$D_{instablefunding(i)} = D_{instable(i)} * R_{forbinstable(i)}$$
$$D_{instablereserve(i)} = D_{instable(i)} * (1 - R_{forbinstable(i)})$$
$$D_{lpp} = Y_{lpp} = Y_{global} * R_{lpp}$$

2.3 Objective Effects

SHR gives an adaptive nature to the farming system. Its main effect can be concluded as: to keep reporting yield-rate contrast data to asset suppliers, in order to guide the capitals to migrate to the most liquidity-needed asset pools all the time. Such effect is identical to the performance of the interest system's duty. In addition, SHR has created some effects that a traditional interest system cannot generate.

2.3.1 Ending The Isolation of Asset Pools

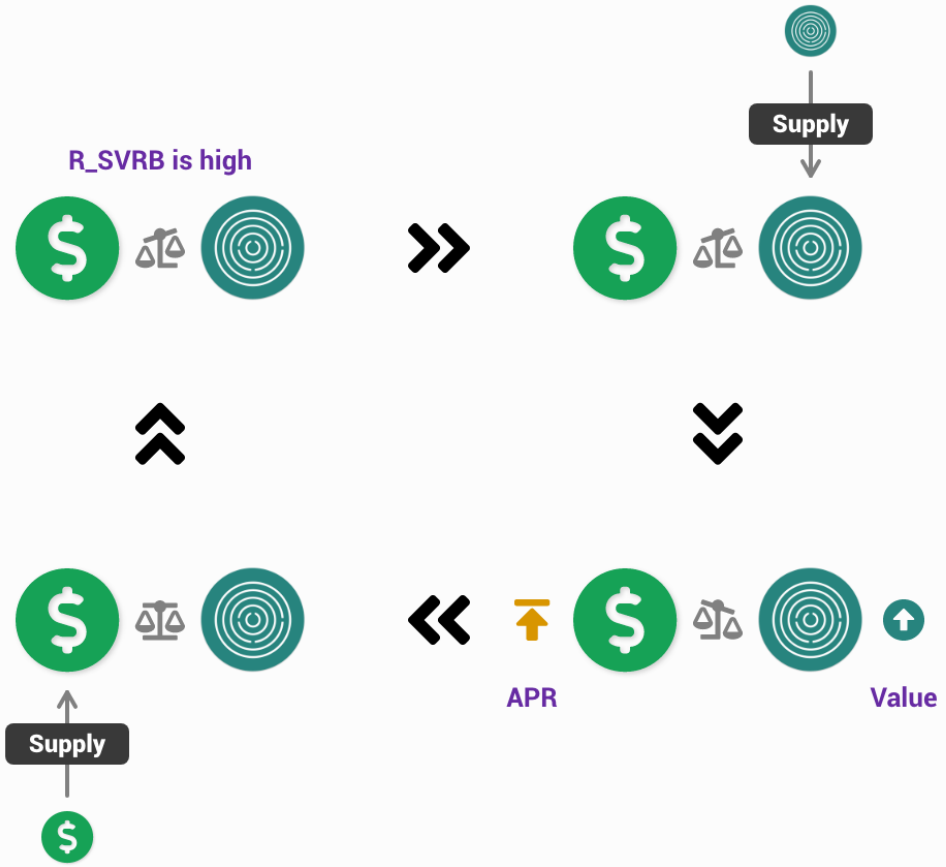
In common money markets, the yield-rate adjustment of different asset types are mostly isolated. It forces asset suppliers to consider the changes of exchange price of various assets when they impact the actual rates of return, no matter what price standard is used. SHR uses supplied value as the basis of share, and unifies the weights of all assets with SVWD, so that the isolation of asset pools is completely broken.

In Maze, the market price of one asset rising will cause its pool's share weight automatically rises against the other asset pools. The asset suppliers thus automatically get the yield-rate advantage brought by its principal's growth. Such a feature does not exist in traditional money markets, e.g. bitcoin suppliers usually get a lower yield rate, and even the price increase is not helpful to their supply income rate. Now, based on SVWD, the market price performance will directly influence the suppliers' willingness to deposit this asset into Maze. To the project team of a permitted asset, boosting the price of the asset will effectively drive their token holders to lock the asset using Maze's service. It could say that SVWD is a hidden mechanism to support a capital race.

2.3.2 MAZE as Core Drive

Based on Value As Share and SVRB mechanism, and the USP structure design, MAZE is applied as the protocol driving factor creating friction with stablecoins. MAZE's supplied value is directly affecting the competitive status of the yield rates in USP. The complex factors include the supplied amounts of MAZE and stablecoins, and the market price of MAZE.

SHR's goal is to form the following tokendynamics loop between MAZE and stablecoins.



Commonly, the risk on stablecoin suppliers is the lowest, and their yield-rates can be regarded as the risk-free rates which is the yield-rate contrast start point for all protocol users. Because the supply of stablecoins is relatively large, it will easily decrease the value proportion of MAZE in USP, causing the rise of R_svrb and a more preferential yield rate for MAZE suppliers. This lifts the willingness of MAZE suppliers to add their supply, to rebalance SVRB. However, more MAZE deposits will exacerbate the liquidity squeeze of MAZE on external markets, bringing a tendency of MAZE price climbing. If the price rising happens, the deposited extra MAZE will cause R_svrb dropping, making the risk-free rate better again. More stablecoin supply will be attracted which leads USP to return to a MAZE shortage situation.

In this loop, a collateral effect will generate: as the MAZE price climbs, the protocol's whole yield rate advantage will rise against other external money markets, including the yield rates of instablecoin supply, which continues to increase the attractiveness of the system. As the system grows, MAZE, which carries the governance function of the protocol, will be more appealing.

2.3.3 Providing Distinct Options

No matter how the loan risk management performs, a lending protocol has inevitable hidden risks. Generally suppliers are facing the risks like unavailability of pool liquidity or deficit in the pool. Under zero-interest lending environment, the asset pools are more likely to be intensively occupied for a long time which causes difficulties on funds redeeming. For this reason, Maze provides a Reserve-Funding Pool structure for all permitted assets to ensure users' rights of choosing risks. In the meantime, FORB is capable of guiding supply strategies based on each asset's occupation rate. If the user wants to use lending services or approach to higher yield rates, he can actively choose Funding Pools and take the corresponding risks. If not wanting to take risks but still having some profits, he can choose Reserve Pools, and in fact becomes the member of the yield-rate contrast group.

The concept seems like the deposit-reserve ratio, but the reserve is determined by the anti-risk users themselves. The other uses of Reserve Pools will be discussed later.

2.3.4 High-Frequency Yield-Rate Consensus Algorithm

To most banks in the world, interest rates are very passivated parameters and change infrequently, which is important to the stability of the traditional world, especially the borrowers. Too much rate fluctuations will cause trouble to their business. However, on crypto markets, the high volatility is widely accepted, and if there is no interest on the debt side, no one will be troubled by the volatility of costs.

In Maze, SHR plays a role of the high-frequency yield-rate consensus coordinator. It realizes a highly complex adjustment of dynamic rates on a 10-minute time granularity in the true sense. Imagine it, a bank generates a certain interest for your demand deposit every 10 minutes, and the computation of interest is depending on the market price performances of abundant assets with every user's behavior in this bank - such a bank is Maze.

2.4 Difficulties of Implementation

According to the SHR principle, in every computation cycle, the status of the pools is different, so it is unlikely to decide the distribution parameters earlier. The computation can only be proceeded based on the status snapshot after one cycle is over. The yield must be passed across layers based on each distribution rules, and cannot be pre-determined in objective MPs via any priori algorithm, so the farming computation system of Maze is a posterior one. Only knowing such concepts can ensure a correctly functional program.

2.4.1 Posterior Stratified Yield Delivery

Implementing Maze's farming system is unlike a simple task composing a MasterChef.sol. The high-dynamic character of SHR may make it impossible to land on a layer-1 blockchain easily. This part will explain the reason intuitively.

The Paradox

The common way of implementing a farming contract is to validate the revenue when the farmer calls a claim function. The contract will calculate the balance based on the information of the latest block height. However, this solution does not work in Maze's design, because under the demand of SHR, you apparently should not use the latest ledger status to finalize the income value for all the passed cycles.

The tables below intuitively show the reason:

Cycle 1

STEP I — **MP Reward State**

FORB Value:	0.60
MP Pair Reward:	100 MAZE
rMP Reward:	40 MAZE
fMP Reward:	60 MAZE

STEP II — **Staking State**

	rMP	fMP
Alice	4 rtoken	12 ftoken
Bob	6 rtoken	24 ftoken

STEP III — **Reward Sheet 1**

	rMP	fMP
Alice	16 MAZE	20 MAZE
Bob	24 MAZE	40 MAZE

STEP IV — **Harvest Balance**

	rMP	fMP
Alice	16 MAZE	20 MAZE
Bob	24 MAZE	40 MAZE

Cycle 2

MP Reward State

FORB Value:	0.50
MP Pair Reward:	60 MAZE
rMP Reward:	30 MAZE
fMP Reward:	30 MAZE

Staking State

	rMP	fMP
Alice	4 rtoken	8 ftoken
Bob	6 rtoken	2 ftoken

Reward Sheet 2

	rMP	fMP
Alice	12 MAZE	24 MAZE
Bob	18 MAZE	6 MAZE

Harvest Balance

	rMP	fMP
Alice	28 MAZE	44 MAZE
Bob	42 MAZE	46 MAZE

As we move from one cycle to the next, the parameters will change due to price or amount fluctuation, and thus the reward to each MP is altered. With the latest status, the contract can precisely calculate the income for one cycle, but it will need to keep the data in the memory in order to accumulate the reward for further cycles. If there are a large number of users farming in multiple MPs, the memory will cache a huge balance sheet which is unpractical. Otherwise, the contract must store the income record to the blockchain ledger after each cycle, costing much gas and performance spending in the long term.

Assuming there are 1,000 users each farming in 10 MPs, generating 10,000 income record tasks in one cycle, which will be 43,200,000 operations in a month.

So we have a paradox here:

- If the contract only uses the final status of the block height where the user calls the claim, the reward value will be completely incorrect.
- If the contract caches the income value of each cycle in the memory, the network will eventually break.
- If the contract updates the income value of each cycle to the blockchain ledger, the transaction will cause great gas cost and performance waste.
- If we host a back-end service to compute the reward balance and assign a smart contract to execute users' harvest request based on off-chain records, the system will lose its transparency.

To crack the paradox, the team of Maze has developed an innovative technology named **Farmbase**.

3 Implementing The First Dynamic Farming Pool: Farmbase Middleware Protocol

Farmbase Protocol is a byproduct born during the development of SHR algorithm. It can be concluded into a simple idea: it is a middleware protocol which reads the farm status from a targetchain periodically, and then computes to keep updating a reward balance ledger for the users to accordingly harvest on their targetchain.

3.1 Foremost Targets

Farmbase is a transparent computation middleware service which is unconscious, decentralized, independently and concentratedly tasked, flexible-configured. It can be deployed on all potential projects with structured income distribution demands.

3.1.1 User Unconsciousness

As a middleware, Farmbase will not be perceived by users in Maze's key working flows.

3.1.2 Transparent Verifiable Dedicated Computation

Farmbase must be a decentralized service, which needs to be:

- Maintained by a decentralized underlying protocol.
- Collect and clean data from decentralized systems.
- Make the income computation process and results public to users.
- Store and allow retracing of computation process.
- Provide result data for the smart contracts on target systems.

3.1.3 Not Spending Gas & Performance of Main Task Network

We need to migrate all farming yield computation missions which are not necessarily deployed on the main task network, to Farmbase. In the other words, from the change of MP status by users to the harvest in MP by users, all of the middle segments are not in charge by the main task network now, so there will be no unnecessary gas costs and no occupation of mempool and VM performance on the original protocols.

3.1.4 Easy Iteration of Algorithms

Deploying the income distribution algorithm in the Farmbase contracts, means that the developers can adjust the algorithms' parameters easily and abstractly, or even replace the entire ones.

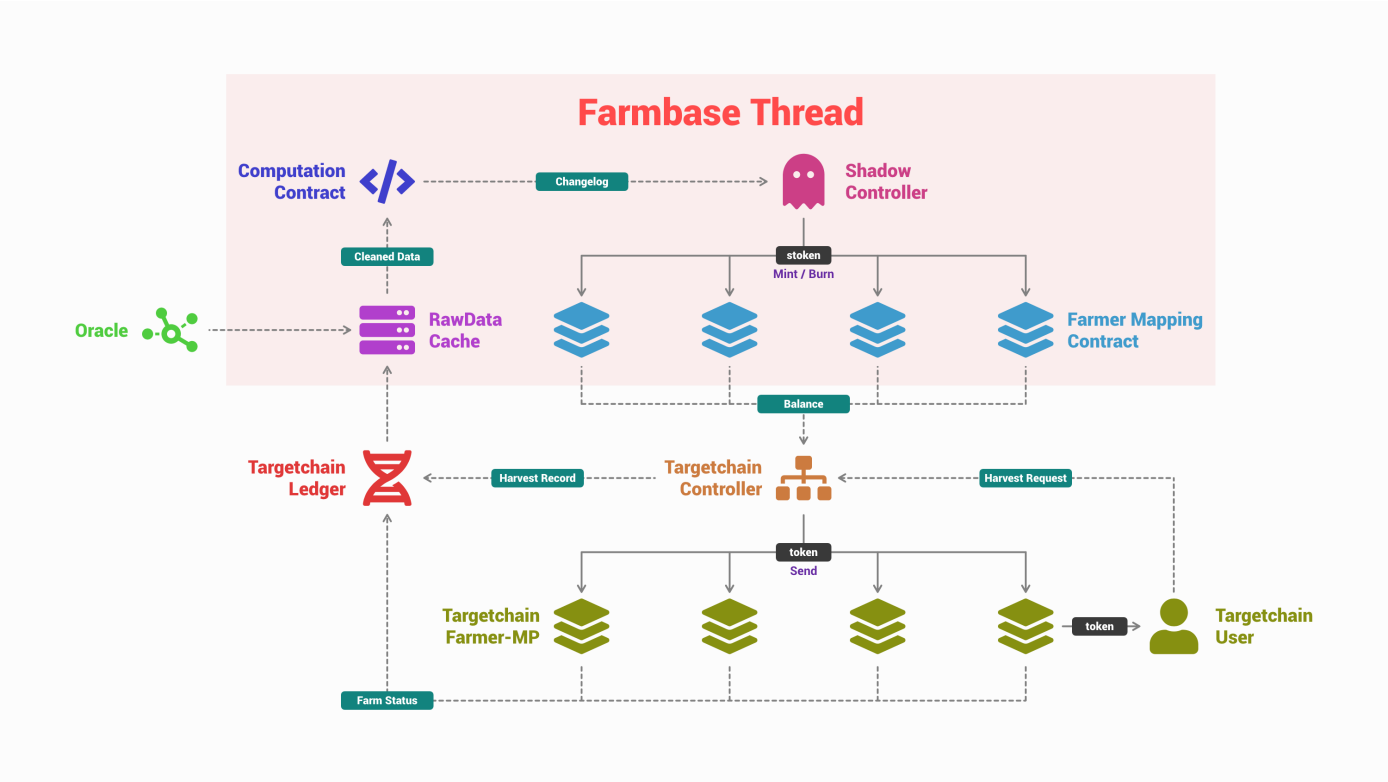
3.2 Introducing Farmbase

Farmbase Protocol is a multi-chain consensus network built with a large number of dedicated task domains. A single member blockchain is called a Farmbase Thread. A Farmbase Thread ought to serve one target protocol only. This is because, in our view, every mining pool project's computation resources should be constructed by the developers and community members and secured by themselves. There is no need to share computation performance across projects.

3.2.1 The Abstract Principle

This part will take one Farmbase Thread as an example to introduce the middleware's principle.

When a Farmbase Thread builds collaboration with its targetchain, the following flow structure will be formed.



Concepts

Item	Description
Farmbase Thread	A Farmbase blockchain which dedicates to one farming business.
Targetchain	A blockchain where the underlying business lands.
RawData Cache	Collects and cleans the data from a targetchain for the computation mission.
Computation Contract	A set of smart contracts which compute reward distribution based on transparent functions.
Shadow Controller	A smart contract which receives balance updates and accordingly mints or burn stoken in farmer mapping contracts.
Farmer Mapping Contract	Stores stoken as a farmer delegation account pointed to a targetchain Farmer-MP address pair on a targetchain.
Targetchain Farmer-MP	The parameter to identify the farmer's share location on a targetchain.
Targetchain Controller	Acquires the latest reward balance data from Farmbase to execute a farmer's harvest request.
stoken	Shadow token, a token on Farmbase Thread to map the reward asset on targetchain.

The Universal Working Flow

- Read & Compute

  To a Farmbase-integrated Dapp, its MP data is periodically read by RawData Cache. If needed, the oracle can be integrated in this step. Rawdata Cache submits the pre-processed data to Computation Contract. The customized income distribution algorithm will run on Computation Contract to generate the cycle's income ledger. If needed, the relational database engine can be integrated to assist with the final balance ledger processing in this step. Then, Shadow Controller mints Shadow tokens to Farmer Mapping Contracts according to the income balance ledger. If it is a new farmer, new mapping contract will be deployed.

- Call Claiming

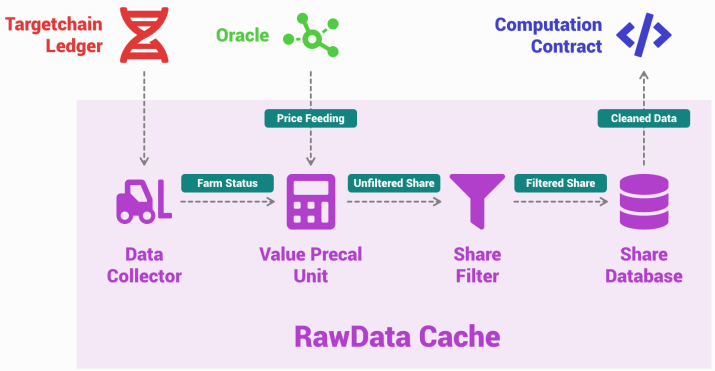
  When the user calls a claiming on targetchain, Targetchain Controller will read the stoken balance in the Farming Mapping Contract to verify the valid claiming amount, and then release the token to the user account.

  - The Atomic Closure Of Harvest Claiming

    Before the income is released, Farmbase Thread needs to task Shadow Controller to burn the corresponding stoken balance in the Farmer Mapping Contract. Before Targetchain Controller confirms the stoken burning, no release will be done, nor permitting the next claim request of the same Targetchain Farmer-MP address pair.

Value Precal Unit (VPU)

A farmer no longer active leaves dust balance in MPs from time to time. If not filtered, dust balance will cause waste on the computation performance and ledger space of the Farmbase Thread. VPU is the module to mitigate such problem, which set up at the entrance of RawData Cache. The flow structure is below.



The MP developer can set up filter parameters in VPU to determine the farming share threshold accepted by the cache. If needed, the oracle can be planted in this step to filter useless share based on prices or more. (Max or Min can both be used as filter rules. Min is often used to filter dust balance while Max can control the whale's behavior.)

3.2.2 Technical Design Facts

Farmbase Protocol is an independent middleware project, and Maze is its first client. The detailed design and implementation of Farmbase will be elaborated in Farmbase Whitepaper. In this part, the essential technical designs are briefly given.

- Network Structure

  The isomorphic network built with data shards. The nodes are divided into different domains, isolated with NetworkID, and 0 is the Global Domain. The data in Global Domain will be synchronized in all nodes. The dedicated task domains are called Farmbase Thread, deployed by targetchain project teams. The data of a Farmbase Thread are not synchronized in Global Domain or others, but only in itself.

- Basic Function

  It decouples the income computation and business logic of a mining pool.

- Consensus Algorithm

  The main chain is PoS, and threads choose their own.

- Encryption Algorithm

  Curve25519.

- Virtual Machine

  WebAssembly.

- System-level Contract
  - Farmbase Validator Contract

    For PoS implementation, including the economic model's interface.
  - Farmbase Gateway Contract

  For limiting access authority. Global Domain is not open for contract deployment. The dedicated task domains' contract deployment are open for developers with keys.
  - Farmbase Domain Management

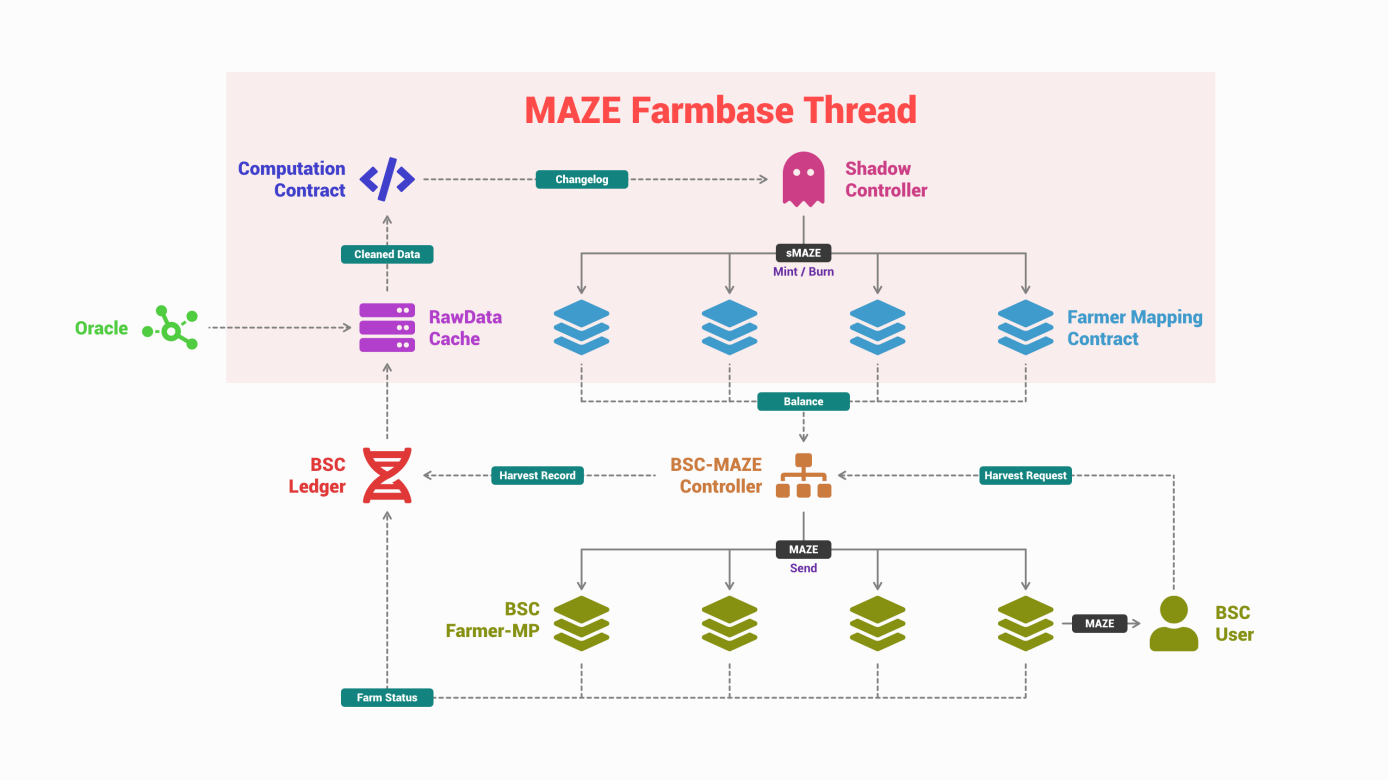
    To distinguish Global Domain and dedicated task domains, and to manage assets on different chains.
- Optional Storage Engine

  An internal PostgreSQL can assist with super massive ledger computation.
- Cross-chain Security Protocol

  A security interface is planted on the business layers of both targetchain and Farmbase Thread. A user-targetchain-farmbase signature chain is used to prevent attacks of fake data.

3.3 Farmbase-Maze Collaboration

Due to the dynamic nature of SHR, Maze has to depend on a Farmbase Thread to power its token economic system. The first mainnet version of Maze will be launched on BSC, so we take Maze-BSC as the example to explain the collaboration flow of Farmbase and Maze.



3.3.1 Maze-BSC-Farmbase Collaboration Flow

This part will elaborate the working flow of farming income distribution with the middleware as abstractly as possible.

- Step 1

  RawData Cache reads the new cycle's status from the BSC ledger. VPU filters the dust balances based on a minimum threshold. The original data is now pre-processed into a format with Farmer-MP address pairs and valid token balances.

- Step 2

  Computation Contract inputs the pre-processed data to calculate the income toward each rMP, fMP and dMP for this cycle based on objective functions of SHR.

- Step 3

  The PostgreSQL engine assists with the computation of all Farmer-MP address pairs' final income ledger.

- Step 4

  Shadow Controller inputs the final income ledger and mint sMAZE to Farmer Mapping Contracts. For the new Farmer-MP address pairs, new contracts are deployed first.

- Step 5

  BSC's user calls a claim on Maze software, and then BSC-MAZE controller checks the sMAZE balance of the Farmer Mapping Contract on Farmbase.

- Step 6

  BSC-MAZE controller temporarily suspends the claim authority of the account and submits claiming execution data to Farmbase.

- Step 7

  Shadow Controller burns the sMAZE in Farmer Mapping Contract according to claiming execution data.

- Step 8

  The Maze contract on BSC releases MAZE to the farmer's account based on the burnt sMAZE balance.

- Step 9

  BSC-MAZE controller reopens the claim authority of the account.

By now, Farmbase Thread has finished the entire process from computing to releasing of the income.

3.4 The Abilities of Farmbase

Farmbase's abilities is not limited in basic farming income distribution's transparent computing. It also has many potentials to push the evolution of DeFi business.

3.4.1 The Mining Pool Manager to Fulfill Governance Demands

Farmbase allows customized configurations in any location of RawData Cache, Computation Contract, Shadow Controller and Farmer Mapping Contracts, which means the operators of mining pools can easily interfere the income distribution systems of the projects. The interference may cover:

- Quickly adding a speed factor of distribution for a selected MP to fulfill the demand of enlarge or shrink the reward.
- Filter the shares with rules, such as adding an invitation system which allows permitted farmers in specific pools.
- Control the balance of specific Farmer Mapping Contracts to make a different reward release curve from the other farmers. It could impact some whale farmers.
- Rapidly shut down or adjust rules when being attacked or facing emergency, without affecting the mainnet's business logic.
- ...

There could be many possible MP management strategies to be invented after Farmbase is widely used, but all managers should note that it is important to respect the community's interest when using this tool.

3.4.2 The Data Aggregating Interface of Income Distribution Systems

Today, there are few DeFi projects building a data visualization service such as a farming income aggregating curve for their farmers. The yield farming is just born lately, and infrastructures are not mature, but we believe that as DeFi is becoming a widely adapted scenario, farming will be the common financial business in crypto markets. Therefore, it will be quite significant to create a financial data system of farming business for users.

Every Farmbase Thread saves a detailed history of the income distribution for all users, which simplifies the construction of a visualized interface related to farming for project founders. It is easy to predict that in the near future, as farming becomes a daily matter, every user can intuitively get analysis reports via Farmbase data interface, and thus make more intelligent asset management decisions.

3.4.3 Realizing The Scalability of Mining Pools

Beside the technical solution of decoupling complex income distribution algorithms, Farmbase also provides privacy protection and cross-chain scalability strengths for mining projects.

Privacy Protection

Since Farmbase cross-chain communication has the homomorphic hiding feature naturally, MP developers can use reformed Farmer Mapping Contract to allow users to redirect their income to other addresses or even other blockchains to hide their traces.

Cross-chain Scalability

If there is a special mining pool which requires users to stake the assets on two blockchains into MPs respectively to form a liquidity pair so it could be taken as a valid share, how can it be implemented? Using Farmbase Thread to combine two blockchains' ledger data feeding, the cross-chain farming shares can be easily joined logically.

In fact, on the way to becoming a protocol crossing all blockchains, it is such Farmbase cross-chain scalability which helps Maze.

Cross-chain Cycle Recalibration

Although the block interval is averagely controlled, but there are always errors. From the view of multi-chain data collecting, the calibration of cycles must be considered. Farmbase includes a virtual clock to coordinate the ledger data collection range on all targetchains.

4 Crosschain Synchronized MP

Nowadays the lending protocols we see are independently running on each blockchains. In the other words, though these are the same projects' various branches, they are separate. If Maze is running simultaneously on Ethereum, BSC and Polkadot parachains, then will the SHR systems on the different protocol's economies can break the curse of lonely islands? The answer is yes for sure. Farmbase is born to bring the condition for the fusion of economies.

4.1 Farmbase Degeneracy Of Cross-chain Identical MP

As the multi-chain ecosystem grows, more projects are issuing their tokens across various blockchains. When Maze is running on different blockchains, it will definitely meet underlying assets which land on multiple chains (typically USDT). If Farmbase doesn't exist, the MPs of the assets on their chains would have to separately run on the parameters of themselves, with independent SHRs. Farmbase has shown up to change everything.

Using Farmbase to merge the MPs of cross-chained identical assets is very simple - just to accept multiple ledgers, and then to add the pool balances together when the whole share of the MP is calculated:

$$V_{token} = \left(\sum_{i=1}^{n} N_{token(chainId=i)}\right) * C_{token}$$

Maze uses such a way to merge the shares of MPs lying on all blockchains. Any time, no matter how many networks are expanded on, there is only one SHR needed, and all networks share the same economy driven by one MAZE token.

After the merging, users are still unconscious about Farmbase middleware, but everyone can see that on every blockchain's Maze software, the yield-rates of supplying identical assets are synchronized.

A field will be included to locate different targetchains in Farmer Mapping Contracts' data format to help the controller on different targetchains to identify the reward which belongs to the chain.

4.2 Maze Branches On All Blockchains

When Maze merges MPs with Farmbase's help, it can distribute itself to blockchains of all development stages in a more flexible way.

The blockchains' development status is different, so be their asset types and liquidity scales. If building separate lending protocols on some not-so-developed blockchains or function-focused networks, we may encounter problems like lack of incentive feedback and liquidity shortage. Fortunately, Maze can easily deploy branches on these chains, building asset pools and their MPs, and then provide synchronized yield-rates for these protocols' farmers through merged mining pools. In this way, the protocols can instantly own a usable and incentive-synchronized banking system.

4.2.1 Supporting Fragment Branches

The more significant point is that, like a bank's branches may operate different businesses in different regions, Maze can build fragment branches on different blockchains. Beside supporting only permitted assets on some blockchains, Maze can provide only Reserve Pools or Funding Pools for specific assets on selected networks and their rMPs or fMPs. SHR will distribute the all-chain-crossed MAZE tokens for them with these MPs merged on a whole ledger of shares by Farmbase.

The cross-chained scalability of MAZE income distribution and the deployment of fragment branches will strengthen Maze protocol with unique appeal. If we make such a metaphor that abundant chain systems are the stars spread across the galaxy, then Maze is the star fleet which can sail to the most distant dimensions and expand the federation economy to every civilization.

5 Implementing Drawing Right Transfer Protocol of Cross-chain Assets: Farmbase Hawala Thread

More and more assets are becoming cross-chain assets (the same token of one protocol issued on different blockchains), like USDT. Regarding these types of assets, if a user wishes to transfer his deposit from one chain to another, can he find an intuitive, fast, unlimited way of 'sending'? Now such behavior on-chain is still uneasy. If he wants to make the migration, usually it is better to rely on a CEX's transform.

We will obviously face such awkwardness following the Maze blueprint to deploy branches on all blockchains: The users with assets deposited in so many Maze pool branches, if wanting to go to another blockchain, he will need to withdraw funds from the current branch and then attempt to migrate via some other plans.

In fact, due to the combination of Maze and Farmbase, we are only one step away from eliminating the awkwardness above.

5.1 Hawala: Transfer of Drawing Right Instead of Assets

Hawala ia an ancient way of transfer originated in India, also called 'money transfer without money movement'. It needs to be done via Hawala brokers at two locations. A hawala transaction flow is like this:

- The transferer Alice needs to send her funds to the receiver Bob but has no direct ways. She can only approach to the broker Charlie. Bob knows another broker David, and David knows Charlie.
- Alice gives the funds to Charlie and sets a secret code.
- Charlie tells David that he has received Alice's funds with the secret code.
- Alice tells Bob the code.
- Bob tells David the code, verified correct.
- David gives the funds of equal amount. The transfer is done.

We can see from the hawala flow that only a drawing right is transferred between Alice and Bob, never the actual funds.

On different blockchains, the cross-chain transfer of underlying assets is like Alice directly transfering funds to Bob, which requires much more on infrastructures. It often needs the development team of the underlying assets to build an atomic swap channel between specific blockchains. To the ordinary users, the better cross-chain transfer is exactly the hawala flow, that is, transferring drawing rights. Regarding the real migration of underlying assets, it is exactly the work of reserve asset rebalance between two hawala brokers.

So to fulfill the goal, what we need to do is to build a system which transfers drawing rights, and asset pools spread everywhere.

Note: the hawala mode is illegal under many countries' real financial regulation frameworks. On this protocol, hawala is merely a concept for fast understanding the principle of drawing right transfer protocol of cross-chain assets. It has no connection to the international asset movement activities in the real world.

5.2 Reserve Pools: High-Liquidity Drawing Right Pools Distributed on All Blockchains

To implement a drawing right transfer protocol of cross-chain assets, Maze's Reserve Pools spread on different blockchains are the most essential infrastructure. From now on, Reserve Pools are no longer just a yield-rate contrast group, but they will fulfill the liquidity demand of cross-chain withdrawal on each blockchain.

A user deposits a cross-chained underlying asset into the Reserve Pool, getting rtoken and rmtoken, and thus adds liquidity for the local pool. In the meantime, he has also obtained the rights of withdrawing funds on the other blockchains' Reserve Pools using rtoken and rmtoken. The drawing rights of cross-chain assets are rtoken and rmtoken exactly.

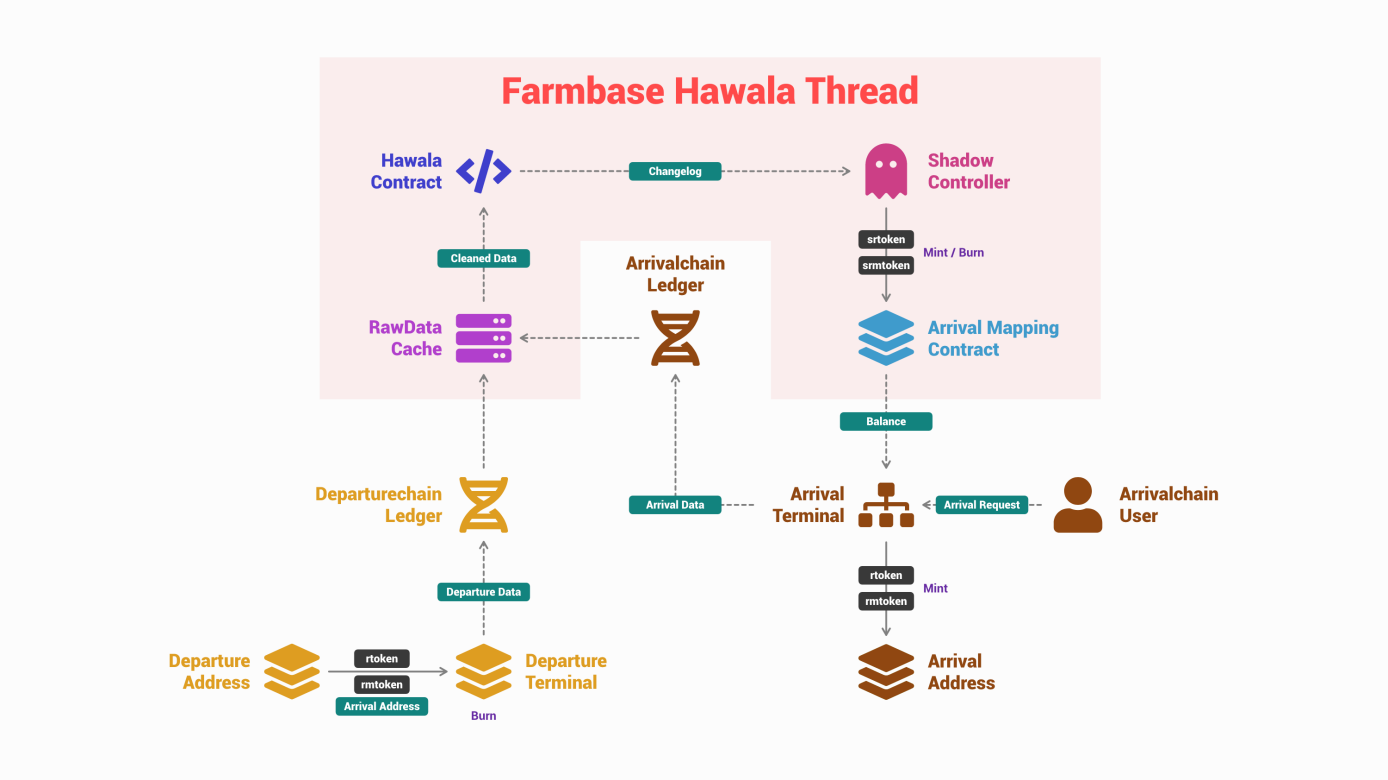
The last question left is: how to realize the cross-chain movement of rtoken and rmtoken? The answer is still using Farmbase.

5.3 Farmbase Hawala Thread

Farmbase Hawala Thread is a special Farmbase Thread. It is separately built, no running any farming computation tasks, but it is focusing on another: to build rtoken and rmtoken's cross-chain migration channel between any two blockchains.

5.3.1 The Abstract Principle

Below is the working flow of Farmbase Hawala Thread.



Concepts

Item	Description
Farmbase Hawala Thread	A blockchain focusing on transfering drawing rights between two blockchains.
Departurechain	The blockchain where the rtoken and rmtoken originally stays.
Departure Address	The account calling the cross-chain transfer of rtoken and rmtoken.
Departure Terminal	A contract deployed on the departurechain, for burning the rtoken and rmtoken which are standing by to migrate.
RawData Cache	Collect and clean data from the departurechain and the arrivalchain.
Hawala Contract	Provide instructions to Shadow Controller on cross-chain transfer.
Shadow Controller	Mint or burn stokens based on the instructions by Hawala Contract.
Arrival Mapping Contract	A contract account pointed to an arrival address, with stokens stored.
Arrivalchain	The blockchain where the rtoken and rmtoken depart for.
Arrival Address	The final account receiving cross-chain transfered rtoken and rmtoken.
Arrival Terminal	A contract deployed on the arrivalchain, to mint rtoken and rmtoken to Arrival Address.
stoken	Shadow tokens, including srtoken and srmtoken, the token on Farmbase Hawala Thread, mapped from rtoken and rmtoken.

The Universal Working Flow

In order to help understanding Farmbase Hawala Thread better, we can imagine a transfer of drawing rights as taking a flight journey. The cross-chain asset drawing rights, rtoken and rmtoken, is your transport luggage, while the address you have on the other blockchain is the destination of this fight trip.

- Step 1

  You need to head to Departure Terminal. Here, you must buy a flight ticket with your destination on it (Arrival Address on the other chain).

- Step 2

  You need to send your transport luggage to the checkin-counter (send the rtoken and rmtoken into Departure Terminal). The ground duty will scan your ticket, sticking a label on your luggage and promise you that the luggage will be returned to you after landing.

- Step 3

  Board the airplane (on the software, you just press 'Send').

- Step 4

Enjoy the flight and then get off (wait seconds and open the software with Arrival Address)

- Step 5

  Go to the carousel of Arrival Terminal and fetch your luggage (on the software, you just press 'Receive').

  - The Atomic Closure Of Receiving Drawing Rights

    Before the drawing rights are accepted, Farmbase Hawala Thread needs to task Shadow Controller to burn the corresponding stoken balance in the Arrival Mapping Contract. Before Arrival Terminal contract confirms the stoken burning, no migration will be done, nor permitting the next receiving request of the same Arrival Terminal.

Withdrawing with Cross-chain Drawing Rights

On the other chain, you can use rtoken and rmtoken to redeem underlying assets in the corresponding Reserve Pools. It is indifferent to your Maze experience on your original blockchain.

It needs to be noted that if the current Maze branch does not have sufficient liquidity in its Reserve Pool, your withdrawal will be hindered. To resolve the problem of liquidity imbalance in Reserve Pools of branches, a new balancer needs to be added into future SHR, which provides preferential yield rates to spent Reserve Pools' rMPs of different branches. Also, a new protocol can be developed to incentive on-chain hawala brokers' rebalance job of underlying asset liquidity. e.g., a higher portal fee can be charged on the transfer direction from spent liquidity to surplus liquidity, and reward such revenue to the liquidity rebalance work.

5.4 The Asset Stargates Connecting All Metaverses

Compared to requesting all project teams to deploy their own token's cross-chain swap channels over different networks, Maze's drawing right transfer protocol has brought an alternative solution. It can be duplicated rapidly on all kinds of blockchains, without costing anything from the team of the underlying assets.

The drawing right transfer protocol connects Maze's all complete or fragment branches' local Reserve Pools into a huge interstellar asset network. Via a quick and abstract Dapp, the capital movement efficiency of all crypto users will be lifted to an unprecedented level. To Maze itself, the MAZE Reserve Pools in all branches are able to ensure the users to comfortably enjoy MAZE-related business on all chains, and spread or gather their assets quickly with the drawing right transfer protocol.

5.4.1 The Mixer of Drawing Rights

Providing mixing strategies for rtoken and rmtoken will turn Reserve Pools into private liquidity pools in all intents. As Farmbase Hawala Thread must be integrated with homomorphic hidings, the drawing rights transferred between chains are naturally protected for privacy. In the meantime, any third-party developers can build mixers supporting rtoken and rmtoken on local blockchains.

Please note: when a user transfer the drawing rights in any form, as the rmtoken must be removed from the rMP, it means that he has to give up the funds' share in SHR for a short moment. In the other word, a user who is transferring the drawing rights remises his short-term income to the other farmers.

6 A Stablecoin Combining CDP & Algorithm

The zero-interest lending system and SHR has brought a unique condition of forging a high quality stablecoin for Maze. The suppliers can choose to forge non-cost ZUSD with their assets. ZUSD is a stablecoin naturally combines CDP and algorithm adjustment.

6.1 Forging ZUSD with CDP

Similar to Maker, Maze supports ZUSD forging by staking assets in CDP. In Maze, a CDP is called a Forge Pool. In nature, a Forge Pool is an isolated Funding Pool created by a user. It has the following characters:

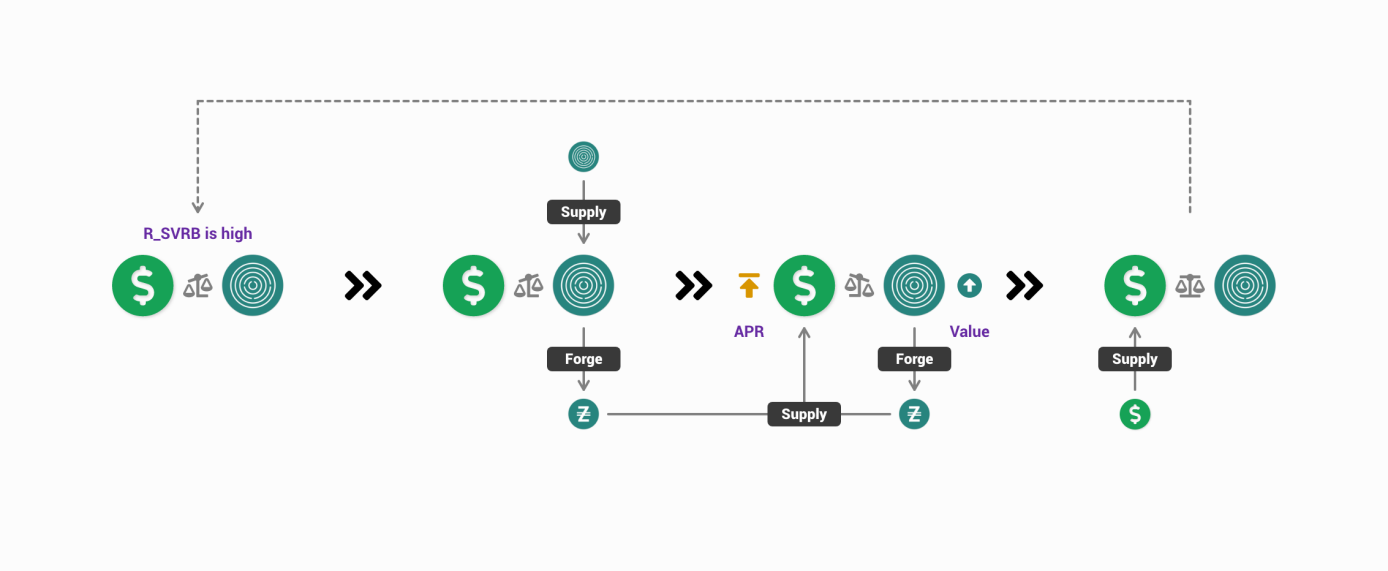
- Every Forge Pool is an over-collateralized zero-interest lending system.
- Every Forge Pool accepts only one type of asset staked.
- Every Forge Pool's liquidity is only for the creator.
- Forge Pools use the same loan risk management system like the lending protocol, including LTV, liquidation threshold, liquidation penalty and health factor.
- Every Forge Pool is liquidated independently.
- Supplying funds into Forge Pools will generate fmtoken which can be staked in fMPs.

From the last feature we can see that the MP of Forge Pools are merged with fMP, and such merging is via the local farming delegation tokens' natural composition, not using Farmbase. In this way, MAZE Forge Pool will generate fmMAZE, and users can get MAZE-fMP share after forge ZUSD.

Since Maze can deploy branches with Forge Pools on multiple networks, ZUSD is a cross-chain asset too. ZUSD will be accepted by USP.

6.2 ZUSD-MAZE Tokendynamics under SHR

With ZUSD's joining, MAZE's token dynamics will evolve into a more complicated pattern.



First of all, MAZE's original token dynamics loop still exists. With the launching of Forge Pools, it is likely that MAZE will exit the activities in common lending markets. MAZE's Funding Pool supplier should migrate to Forge Pools as many as possible. Supplying MAZE in Forge Pools carries several advantages:

- The liquidity is only for the Forge Pool itself, so the supply does not take the risk of high occupation rate in the pool any longer.
- As the Forge Pool is liquidated alone, more meticulous risk management is possible to be made.
- Supplying MAZE in Forge Pools generates fmMAZE so the original yield is not missing at all.
- ZUSD can be supplied as a value of USP, generating rmZUSD and fmZUSD, which means an extra risk-free yield rate for suppliers. (The MAZE suppliers who do not fully use Forge Pools are actually remising yield for ZUSD suppliers)

As Forge Pools need to lock large amount of MAZE, leading to liquidity squeeze, which means the new price climbing drive of MAZE. If the price of MAZE goes up, the existing Forge Pools will have additional forge credit, and thus provide more ZUSD. Because the risk-free rate is climbing with the price rising of MAZE, the best choice for suppliers is to keep supplying ZUSD.

Since the occupation of funds cost no interest, the better option of suppliers is apparently supply underlying assets all the time and borrow some on demand. At any moment, the suppliers should keep the deposit to get rewards at no less than risk-free yield rates.

Generally, ZUSD is the conversion of MAZE asset. Its target is to make a large proportion of MAZE to quit the circulation and turn into a cornerstone reserve on all blockchains. SHR will keep in charge of maintaining the ZUSD-MAZE Token Dynamics loops to ensure the fair value of ZUSD.

It is true that any decentralized stablecoin has inevitable systematic risks. If the value of Forge Pools falls radically in a drastic market fluctuation, or the market price of USD shows abnormal deviation, severe financial incidents could happen. Therefore, before improving safety guarantee measures, ZUSD is not a product to be presented immediately.

7 Conclusion: A Super Sovereign Banking Protocol Crossing All Blockchains

Maze is a project started from the zero-interest lending business. From implementing its first objective, First Principles are always kept, so that step by step we expanded the world's first completely decentralized high-frequency yield rate consensus mechanism, a dedicated transparent computing middleware service, an infinite cross-chained open banking model, a drawing right transfer protocol of cross-chain assets, a CDP & algorithm blended stablecoin, and a non-linear token dynamics system.

A fully developed Maze will be like: its branches operate non-custodial zero-interest lending and decentralized stablecoin issuing business inside each domain, and provide user-friendly, privacy-protected cross-chain deposit and withdrawal services across the entire cyber space, and unite all asset suppliers in a same yield-rate management system - so a Super Sovereign Banking Protocol Crossing All Blockchains is born.

8 Maze's Experimental Network - Neko

Just like Polkadots having a real testing platform Kusama, before mainnet launching, Maze as a massive innovative product, will deploy an experimental network, Neko. Neko will be launched on the mainnet, which has identical functions to Maze's design, and its fundamental token is NEKO. The objectives of Neko includes testifying the feasibility of the product in real under a controllable risk, and being a long-term trial platform for new functions in the future, which means that Neko always has a faster version evolvement than Maze. Please note, as an experimental platform, the operation of Neko could be relatively centralized.

8.1 Overview

- Protocol Environment

  Due to the fluent and low-cost running environment offered by Binance Smart Chain, and the solid support of mainstream underlying assets in its ecosystem, for now we plan to deploy Neko Experimental Network over BSC. Neko also has a Neko-Farmbase Thread for itself.

- Code Reuse & Safety Concern

  The Funding Pool module of Maze core protocol reused part of Aave V2's codes. These codes are tested for long, which can reduce the security risks of asset pool key components, and the difficulty of audit. However, as the Farmbase middleware is original, whose underlying protocol is fully implemented by Maze's development team, it has lifted the workload of audit, and will affect the safety of the protocol.

8.2 Neko V1 Function Configurations

- Main Business Functions

  Non-interest Funding Pools, Reserve Pools, a LP Pool, a lending risk management system, and a farming system controlled by SHR are included. The first Farmbase Thread will be realized in this version.

  - Farmbase-BSC Cross-chain Security Essential

    At the first stage, an off-chain relay service will be applied to coordinate the reward harvest. To prevent evil nodes in BSC, two BSC full nodes are launched to compare to the official nodes of their heights periodically. If 10-block error exists, the suspicious nodes' data will be discarded. When users call the claim, the administration contract will use EIP712sig arrays to verify with Farmbase. NEKO will not be released only if more than 2/3 of signatures are verified.

    At the second stage, a zkp-crosschain component will replace the off-chain relay service.

- Underlying Asset Types Initially Supported

  BUSD, USDT, BNB, ETH, BTCB of BSC mainnet, and the primary token NEKO.

8.3 Parameters

- Parameters of Lending Protocols Module

  Please refer to https://docs.mazeprotocol.com/governance/money-markets

- Parameters of Income Distribution System

  Please refer to https://docs.mazeprotocol.com/governance/configurations

8.4 The Ignition Procedure

- Prerequisites For Launching

  Since SHR needs to take the value ratio of stablecoins and NEKO in USP as reference, before launching the MP, there must be NEKO deposit which fulfills the lowest value request. The deposit is called ignition fuel reserve. The protocol must fetch NEKO price feeding data through NEKO-BNB liquidity pool on Pancakeswap.

- Configuration Checklist & Ignition Height

  Please refer to https://docs.mazeprotocol.com/mazenomics/circulation/ignition-procedure

9 Glossary

According to the order of appearance.

Item	Alias	Description
Funding Pool		A type of asset pools involved in over-collateralized lending business.
Reserve Pool		A type of safe saving pools.
LP Pool		A safe saving pool accepting only MAZE-LP Token.
MP		Each Funding Pool, Reserve Pool and DAO Pool is set up with a corresponding MP. MP collects users' farming delegation tokens as the basis of shares and thus distribute income for users via Stratified Harvest Regulating algorithm.
fMP		Funding Pool's corresponding mining pool.
rMP		Reserve Pool's corresponding mining pool.
lpMP		LP Pool's corresponding mining pool, lpMP.
LTV		The maximum amount of currency that can be borrowed with a specific collateral.
Liquidation Threshold		The percentage at which a loan is defined as under-collateralized.
Liquidation Penalty		The ratio of the bonus that a liquidator can take from the collateral during liquidation execution.
Health Factor		To examine if an account should be liquidated as a whole.
Oracle		A middleware mainly used to feed in price reports.
Liquidator		A role to actively monitor and liquidate unhealth accounts on the protocol.
Asset Delegation Token	ftoken, rtoken, pMAZE	Mapping tokens, the basis of current deposit balance.
Farm Delegation Token	fmtoken, rmtoken, lpMAZE	Mapping tokens, accepted by their corresponding MPs as farming share.
Token Dynamics		The research on interaction and effects of tokens.
Stratified Harvest	SHR	MAZE's offering distribution algorithm, an adaptive yield-rate

Regulators		consensus mechanism.
Value As Share		SHR takes supply value other than just amount as mining share.
Cycle		The time range of a snapshot on ledger.
Global Yield Baseline		The baseline amount of MAZE that the mint contract can issue on each cycle.
Global Yield Speed Factor		The ratio between baseline and actual production.
Global Yield Distribution		The ratios of MAZE distribution toward the first-class logical structure from the mint contract.
United Supply Pool	USP	A set of permitted stablecoin asset pools' corresponding rMP and fMP, and MAZE asset pool's MAZE-rMP and MAZE-fMP.
Instablecoin Supply Pool	ISP	All permitted instablecoin asset pools' corresponding rMP and fMP.
Supplied Value Ratio Balancing	SVRB	An SHR component, in charge of the ratios of MAZE yield distributed from USP toward stablecoin MPs and MAZE MPs.
Simple Value Weighted Distribution	SVWD	An SHR component. Among all same-class asset types, MAZE yield is distributed according to the weights of the different MPs' token values inside the class.
Funding Occupation Rate Balancing	FORB	An SHR component, in charge of the ratios of MAZE yield distributed between an rMP and an fMP of each asset type.
Account Share Weighted Distribution	ASWD	An SHR component, distribute the rewards according to the proportions of the staking delegation tokens in the MP by the accounts.
Farmbase		A transparent computation middleware service which is unconscious, decentralized, independently and concentratively tasked, flexible-configured.
Middleware		Software that lies between an operating system and the applications running on it.
Farmbase Thread		A Farmbase blockchain which dedicates to one farming business.
Targetchain		A blockchain where the underlying business lands.
RawData Cache		Collects and cleans the data from a targetchain for the computation mission. Collect and clean data from the departurechain and the arrivalchain.

Computation Contract		A set of smart contracts which compute reward distribution based on transparent functions.
Shadow Controller		A smart contract which receives balance updates and accordingly mints or burn stoken in farmer mapping contracts. Mint or burn stokens based on the instructions by Hawala Contract.
Farmer Mapping Contract		Stores stoken as a farmer delegation account pointed to a targetchain Farmer-MP address pair on a targetchain.
Targetchain Farmer-MP		The parameter to identify the farmer's share location on a targetchain.
Targetchain Controller		Acquires the latest reward balance data from Farmbase to execute a farmer's harvest request.
stoken		Shadow token, a token on Farmbase Thread to map the reward asset on targetchain, srtoken and srmtoken, the token on Farmbase Hawala Thread, mapped from rtoken and rmtoken.
Atomic Closure		A transaction, finished completely or entirely undone.
Value Precal Unit	VPU	Filter of dust balance, set up at the entrance of RawData Cache.
Dust Balance		Tiny inactive balance, causing performance waste.
Crosschain Synch MP		A technology to merge MPs on different blockchains.
Fragment Branch		Maze's versions with different functions on different blockchains.
Drawing Right		A set of delegation tokens to withdraw funds from supported asset pools.
Farmbase Hawala Thread		A blockchain focusing on transfering drawing rights between two blockchains.
Departurechain		The blockchain where the rtoken and rmtoken originally stays.
Departure Address		The account calling the cross-chain transfer of rtoken and rmtoken.
Departure Terminal		A contract deployed on the departurechain, for burning the rtoken and rmtoken which are standing by to migrate.
Hawala Contract		Provide instructions to Shadow Controller on cross-chain transfer.
Arrival		

Mapping Contract		A contract account pointed to an arrival address, with stokens stored.
Arrivalchain		The blockchain where the rtoken and rmtoken depart for.
Arrival Address		The final account receiving cross-chain transferred rtoken and rmtoken.
Arrival Terminal		A contract deployed on the arrivalchain, to mint rtoken and rmtoken to Arrival Address.
Collateralized Debt Position	CDP	A way of loan invented by Maker, managed by smart contracts running on Ethereum.
Forge Pool		CDP on MAZE, a type of isolated Funding Pools to forge ZUSD.