Maze Whitepaper: The Super Sovereign Banking Protocol Crossing All Blockchains

lunas@mazeprotocol.com

May 2021

Abstract

Maze is a non-custodial and over-collateralized banking protocol that offers zero-interest loans. An algorithm, Stratified Harvest Regulating, replaces the classic interest model of money markets, and creates a highly dynamic yield-rate consensus mechanism. Throughout the implementation, an irreconcilable problem was discovered on the current Layer-1 blockchains.

A decentralized income distribution computing service has been developed to counter the issue. This solution comes in the form of an innovative and transparent computation middleware protocol, Farmbase. The protocol turns 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 construct a drawing right transfer protocol of cross-chain assets letting users 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 ZUSD, a new stablecoin, with blended strengths.

The whitepaper will introduce Maze abstractly and linearly — the principles, implementation 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 on the shoulders of giants. We appreciate Maker, Compound, and Aave's efforts to create the core infrastructure of decentralized finance.

Contents

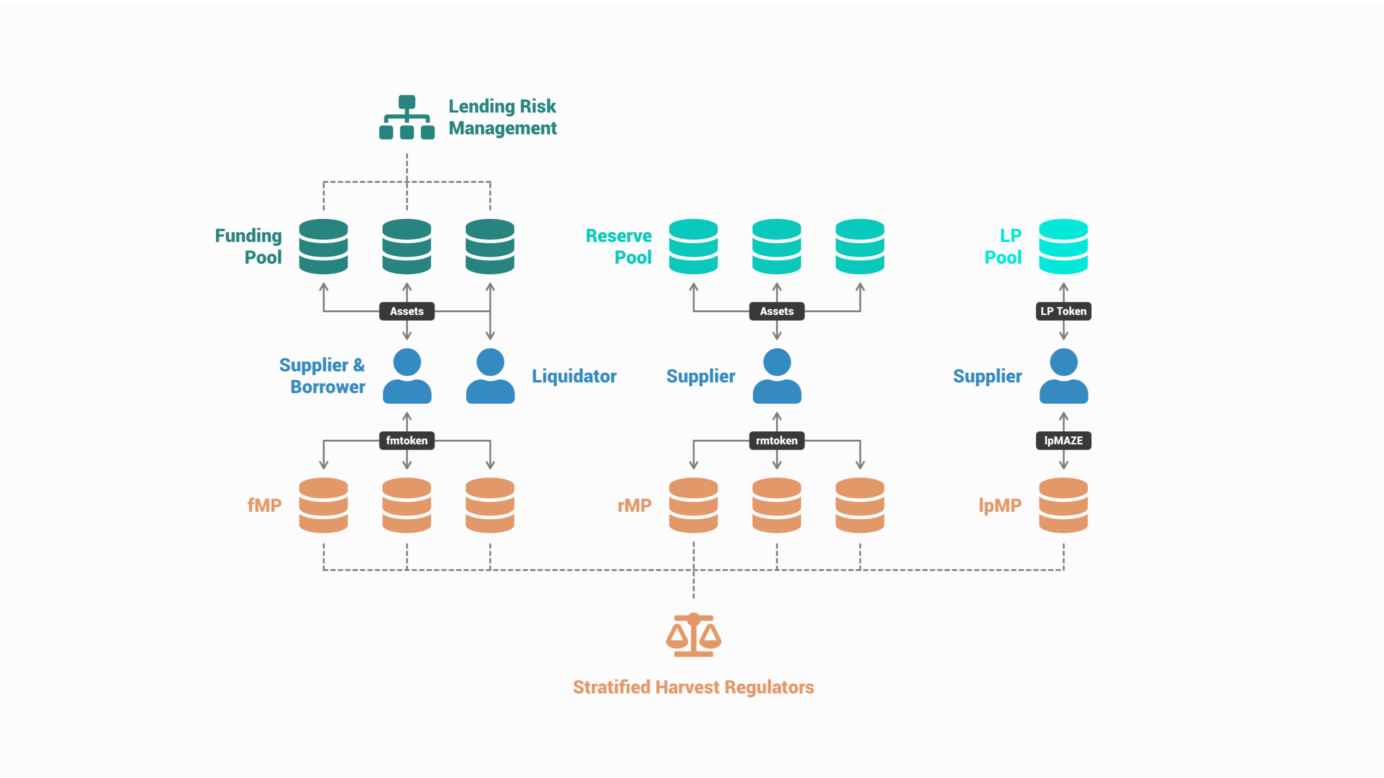
1 Maze Core Protocol: Zero-Interest Lending System

This chapter introduces the components of Maze Protocol, 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 a brief explanation of critical working flows.

1.1 Maze Protocol Structure Overview

Since Maze does not have an interest spread system, suppliers do not obtain direct income by putting assets into asset pools. 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. At the same time, delegation tokens rebuild the economic connection between asset pools and mining pools — this is the inherent self-composability of Maze.

Below is the simplified sketch of Maze core protocol:



We can see from the sketch that Maze is an extension of an Aave-like typical lending protocol. From the view of technical implementation, Maze can be deemed as a non-custodial, synthetic system with fixed zero-interest borrowing rates, 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 smart contracts that allow users to deposit assets and receive delegation tokens in return. All four interactions with the pools — supply, withdraw, borrow, repay — mint or burn corresponding number of delegation tokens. There are three types of asset pools: Funding Pools, Reserve Pools, and LP Pools.

Funding Pool

Every asset permitted by the protocol is configured with a Funding Pool.

Funding Pools are asset pools involved in over-collateralized lending. The assets stored in the Funding 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 suppliers' borrow credits.

Reserve Pool

Every asset permitted by the protocol is configured with a Reserve Pool.

Reserve Pools are safe saving pools. The assets stored in the Reserve 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 pool for savings that accepts 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 mining pool (MP). MP collects users' farming delegation tokens as the basis of shares and thus distributes income for users via the Stratified Harvest Regulating algorithm.

For convenience, the MP of the 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 uses the over-collateralized lending model, so it has an identical lending risk management system like Aave. It operates by four concepts: Loan To Value (LTV) Calculations, Liquidation Threshold, Liquidation Penalty, and Health Factor, together with a self-constructed oracle accessing on-chain price data.

Loan To Value (LTV)

The Loan to Value (LTV) ratio defines the maximum amount of currency that can be borrowed with specific collateral. For instance, if LTV is 75%, the $100 worth of collateral can support up to $75 worth of the 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 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 reference the asset prices on the local blockchain's main decentralized exchange. The prices are reported on a time-wighted-average-price (TWAP) basis.

1.2 Delegation Tokens

Compared with Compound's ctoken and Aave's atoken, the delegation token system of Maze is more complicated. The protocol mints a number of mirrored delegation tokens equal to the assets supplied into an asset pool, and burns an equal number of mirrored delegation tokens when users withdraw assets from the pool. 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 receives an underlying asset, it mints ftokens and fmtokens in a 1:1 ratio.
- When a Reserve Pool receives an underlying asset, it mints rtokens and rmtokens in a 1:1 ratio.
- When the LP Pool receives MAZE-LP Token, it mints pMAZE and lpMAZE in a 1:1 ratio.

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 balances based on asset delegation tokens. All rtoken and pMAZE are ERC-20 standard tokens, transferable between accounts. However, we've disabled the transfer function of ftoken so that users cannot move them, as the lending system must prevent the statistical data related to borrow credits from 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 initiates a loan, the protocol will mint an equal amount of debt tokens. Like ftoken, 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 are not complicated, similar to the standard lending protocols.

1.3.1 Supply & Withdraw Funding Assets

When a user supplies funds to Funding Pools, the account's borrow credit will rise, and the borrow-limit-used ratio (user value borrowed / user borrow credit) will fall to lower the liquidation risk. In contrast, if the user withdraws funds from Funding Pools, the borrow credit will decrease, but the borrow limit will go up, making the liquidation risk rise. For every user, the ability to withdraw from a Funding Pool depends on:

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

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 participate in the lending services, users' supply and withdrawal movements will not affect their lending account. For every user, the ability to withdraw from a Reserve Pool depends on:

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

1.3.3 Borrow & Repay Assets

A user must supply assets in any Funding Pool to acquire the borrow credit before borrowing any assets. Initiating or repaying loans changes the user's borrow limit used and liquidation risks. For every user, the ability to obtain loans from the protocol depends on:

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

1.3.4 Liquidation

If the Health Factor of an account is under 1, it will face liquidation, which 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 a bonus based on the liquidation penalty parameter.

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

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 in the distribution of MAZE. Based on the parameters of Stratified Harvest Regulators, the rewards are distributed to all MPs every 10 minutes.

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 for asset suppliers. This mechanism creates a a new question: what algorithm should be used to distribute incentives to suppliers of different assets while keeping or even surpassing the ability to balance supply and demand of the hundred-years-old interest rate system.

2.1 The Capital Market Driven by Yield-Rate Contrast

During the evolution of the banking industry, the spread between interest rates for borrowing and lending has always been the fundamental business model on which every bank relies. In lending markets, the interest rate serves two major purposes:

- i) it helps to transfer renting costs of money from borrowers to lenders.
- ii) it adjusts the liquidity supply and demand in each asset pool.

If the asset is in demand, the funding rate will rise. The rising rate encourages suppliers to provide more funds to make more earnings and borrowers to reduce the loans, otherwise paying more costs. Besides balancing the supply and demand inside a single asset pool, interest rates can create competition among different assets.

As rates vary across multiple assets, suppliers will shift to higher-rate assets, while borrowers will shift to lower-rate assets. Lastly, a money market also faces competition against external interest rates. If the external market is more attractive to suppliers, the capital will migrate out, and if it is more attractive to borrowers, the capital will not be lent out as much.

To the supply and demand sides of capital, the interest rate is the yield rate — suppliers get a positive yield rate, while borrowers get negative. The participants in the capital markets compare every option's yield rate all the time to optimize their asset management strategies. The risk-free rate is the origin of this comparison, which serves as the benchmark for comparison with other yield rates. For specific risk levels, capital is constantly seeking the best yield-rates.

Since no loan-interest charge exists in Maze, the yield rates for borrowers are no longer negative, indicating that they have an incentive to never pay back the loans. In other words, they can use non-cost assets to seek further rates of returns. Following through on this line of reasoning, certain questions arise focusing on the supply side:

- If the demand for assets of one specific asset goes up, how to attract more supply of funds?
- If one asset is much popular while another is not, inside one certain money market, how to adjust the supply distribution 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 differential-driven nature of capital markets, one thing remains certain: 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 flexible incentive distributions based on all users' current asset management strategies.

2.2 Maze Emission Algorithm

MAZE is the primary token of the protocol, without a hard cap designed. It's mainly generated via the periodical distributions by the Stratified Harvest Regulating algorithm (shortened 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. At the end of the tube system is the personal water tank of each farmer. It is to be noted that all pump stations' parameters will shift with each cycle, and the shifting is mostly based on the devotion in everyone's water tank.

Value As Share

During the statistical procedures of the protocol status, the supplied value (not just amount) of crypto assets are taken as the mining shares by SHR. Doing so enables shares to be compared across asset types. The supply values are counted in USD. The permitted stablecoins are marked continuously at $1.

Cycle

Unlike 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. 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, accordingly calculating all farmers' income within it.

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 ratio 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 tokens associated with stablecoin asset pools' MP, namely rMP and fMP, and MAZE asset pool's MAZE-rMP and MAZE-fMP.

- Instablecoin Supply Pool (ISP)

  ISP contains all tokens associated with unstable asset pools' MP, namely rMP and fMP.

- LP Pool (LPP)

  LPP is lpMP.

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

R_svrb is a function of P_maze:

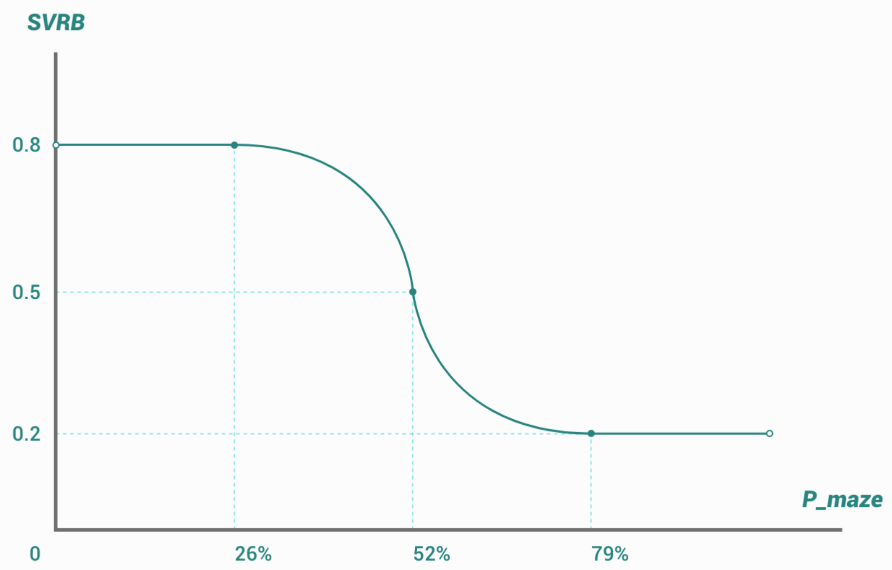
$$R_{svrb} = 0.8 \qquad P_{maze} \in (0, \pi/12)$$

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

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

Please note: the regulating mechanism is untested in a real-world application. During the product operation, the function will be iterated on.

The function's curve:



As evident on the graph, under the regulation of SVRB, when the value of MAZE drops, the yield rate will be higher for 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 cycle T, it is known that: USDT's supplied value is $300,000,000 and BUSD's supplied value is $200,000,000, so USDT's MPs should get 3/5 of MAZE yield and BUSD'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 differential.

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:

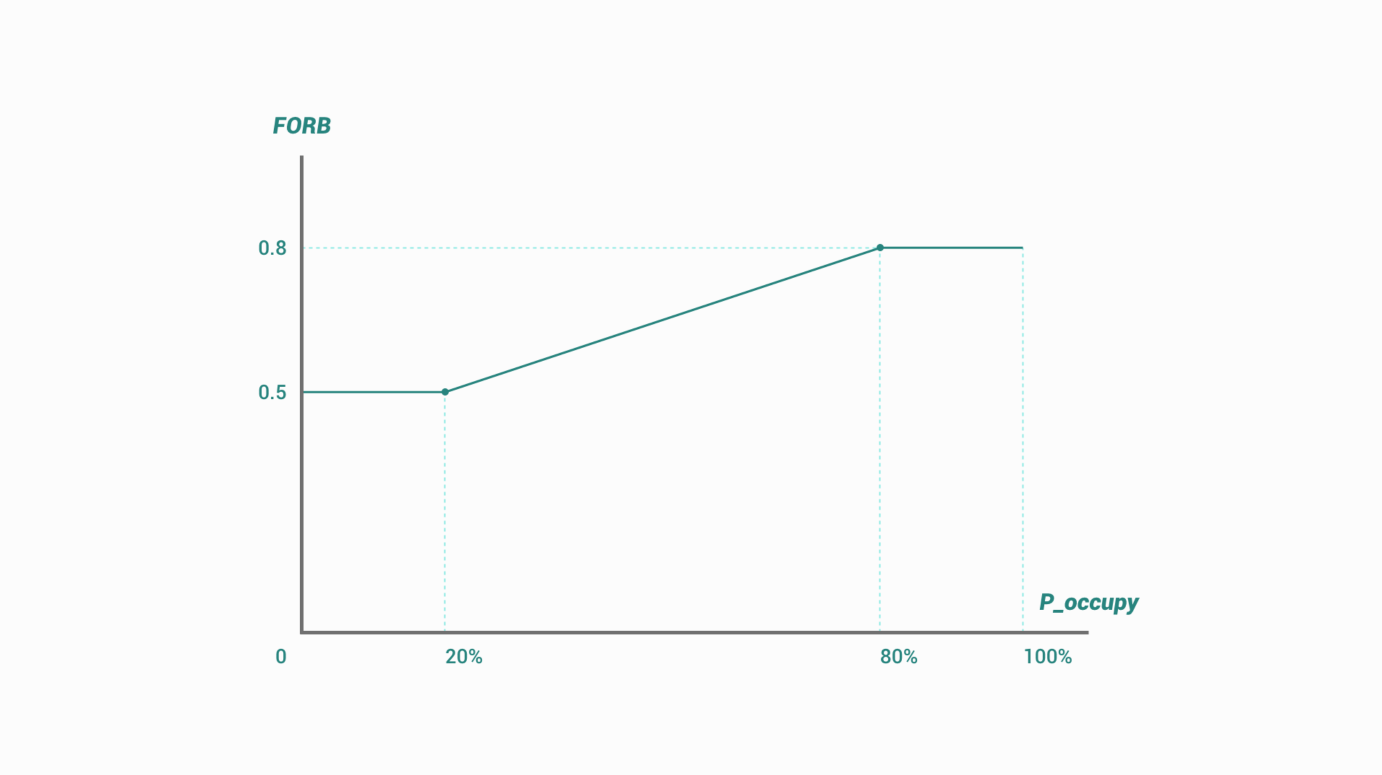
$$R_{forb(i)} = 0.5 \qquad P_{occupy(i)} \in [0, 0.2]$$

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

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

Please note: the regulating mechanism is untested in a real-world application. During the product operation, the function will be iterated on.

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

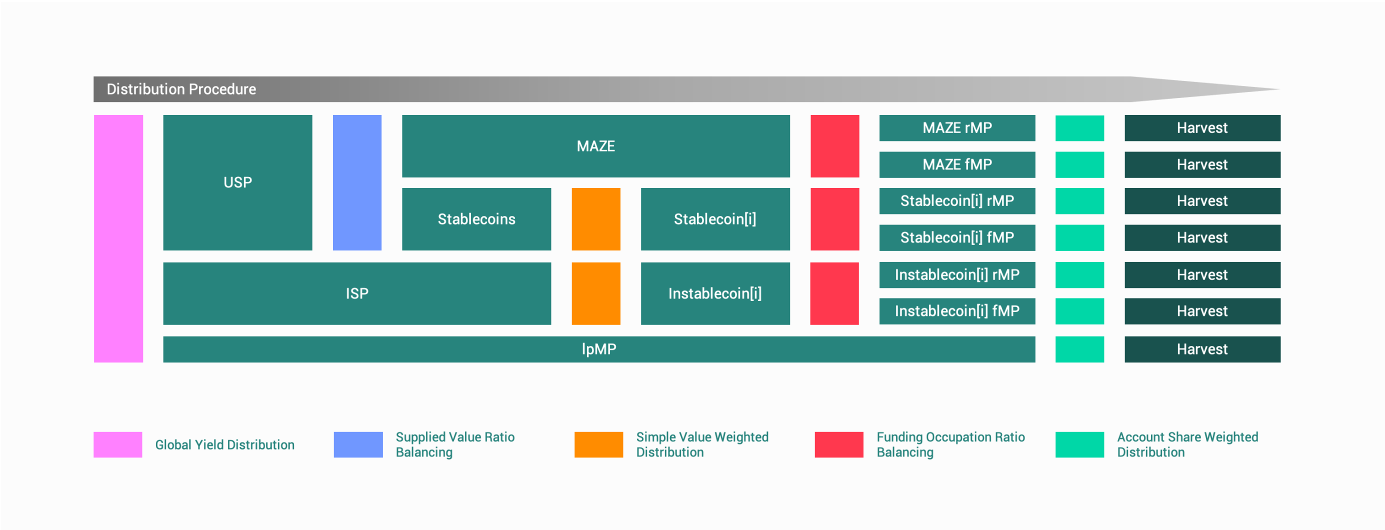
Considering an extreme condition: if there is no supply in an asset's Funding Pool or Reserve Pool, the MP's FORB computation unit will be ineffective. The MP's yield will all be taken by the supplied side.

Account Share Weighted Distribution (ASWD)

ASWD is the bottom-class distribution, distributes the rewards according to the proportions of the staking delegation tokens in the MP by the accounts. In particular, 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 SHR, MAZE distribution mechanism periodically computes the yield reaching each rMP, fMP and IpMP, and finally distributes rewards to every farmer using ASWD. It is equivalent to asking 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 feed for the farming computation system is inherited from the one of the lending protocol modules.

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

  Below are the variable names of the delegation tokens in MPs and meaning of each variable.

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

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

- 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

$$
\begin{gathered}
P_{stablevalue(i)} = V_{stable(i)} / V_{stable} \\
D_{stable(i)} = D_{stable} * P_{stablevalue(i)}
\end{gathered}
$$

  - 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

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

- 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 is to keep reporting yield-rate contrast data to asset suppliers. This is done to guide the capital to the most liquidity-needed asset pool at the time. Such an effect is identical to the performance of the interest rate system. In addition, SHR creates 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 adjustments between 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 calculating shares and unifies the weights of all assets with SVWD so that the isolation of asset pools is completely mitigated.

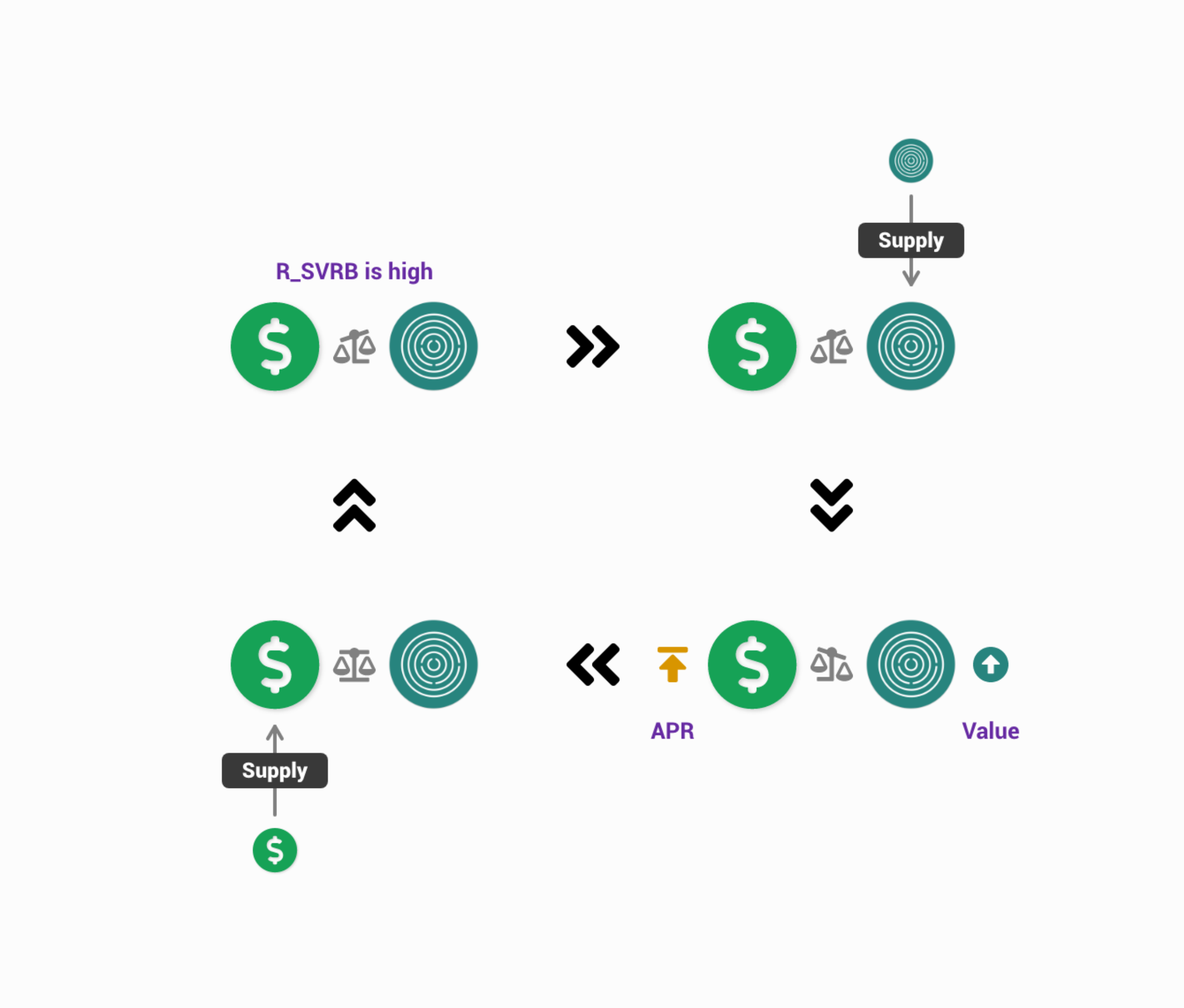
In Maze, the market price of one asset rising will cause its pool's weight to rise against the other asset pools automatically. The asset suppliers thus automatically get the yield-rate advantage brought by its principal's growth. Such a feature does not exist in traditional money markets, e.g., bitcoin suppliers usually get a lower yield rate when bitcoin's price goes up.

Based on SVWD, the market price performance will directly influence the suppliers' willingness to deposit this asset into Maze. An increase in price of a permitted asset will effectively drive the token holders to lock the asset into Maze. It could be said that SVWD is a hidden mechanism to support the race for yield on capital.

2.3.2 MAZE as Core Drive

Based on the Value As Share and SVRB mechanisms, and the USP structure, MAZE token drives the protocol and creates friction with stablecoins. The value of MAZE supplied directly affects competitive status of the yield rates in USP. The factors include the supplied amounts of MAZE and stablecoins, and the market price of MAZE.

The goal of SHR is to create the following self-reinforcing tokendynamics between MAZE and stablecoins.



Commonly, the risk to stablecoin suppliers is the lowest, and their yield rates can be regarded as the risk-free rates. 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 increases the willingness of MAZE suppliers to add MAZE to the asset pool 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 rises, the deposited extra MAZE will cause R_svrb to drop, making the risk-free rate higher. More stablecoin supply will be attracted, which leads USP to again create a shortage for MAZE tokens..

In this loop, a collateral effect will be generated: as the MAZE price climbs, the protocol's full yield rate advantage will rise against other external money markets, including the yield rates of instablecoin supply, which will continue 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

Regardless of how well the loan risk management performs, a lending protocol has inevitable hidden risks. Generally, suppliers face the risks like unavailability of pool liquidity or a deficit in the pool. Under a zero-interest lending environment, the asset pools are more likely to be intensively occupied for a long time, which causes difficulties on redeeming funds. For this reason, Maze provides a Reserve-Funding Pool structure for all permitted assets to give users a choice between different risks. .

In the meantime, FORB is capable of guiding supply strategies based on each asset's occupation rate. If a user wants to use lending services or approach higher yield rates, he can actively choose Funding Pools and take the corresponding risks. The other approach is not to take risks but still generate income. In that case, a user can choose Reserve Pools, becoming a member of the yield-rate contrast group.

The concept seems like the deposit-reserve ratio, but the users with lower risk tolerance determine the reserve. The other uses of Reserve Pools will be addressed later.

2.3.4 High-Frequency Yield-Rate Consensus Algorithm

To the world's banks, interest rates are very passivated parameters and infrequently change, which is essential to the stability of the traditional money markets, especially for borrowers. Frequency interest rate fluctuations cause trouble to their business. However, in 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 the 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: a bank that generates a certain interest on your demand deposits every 10 minutes. The computation of interest depends on the market prices of assets as well as the behaviors of every user in this bank — Maze is such a bank.

2.4 Difficulties of Implementation

According to the SHR principle, in every computation cycle, the status of the pools is different and it does not depend on the parameters of earlier distributions. 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 rule. It cannot be predetermined in objective MPs via any a priori algorithm, so the farming computation system of Maze is a posterior one. Thus, we design Maze with those concepts in mind to ensure a correctly functional program.

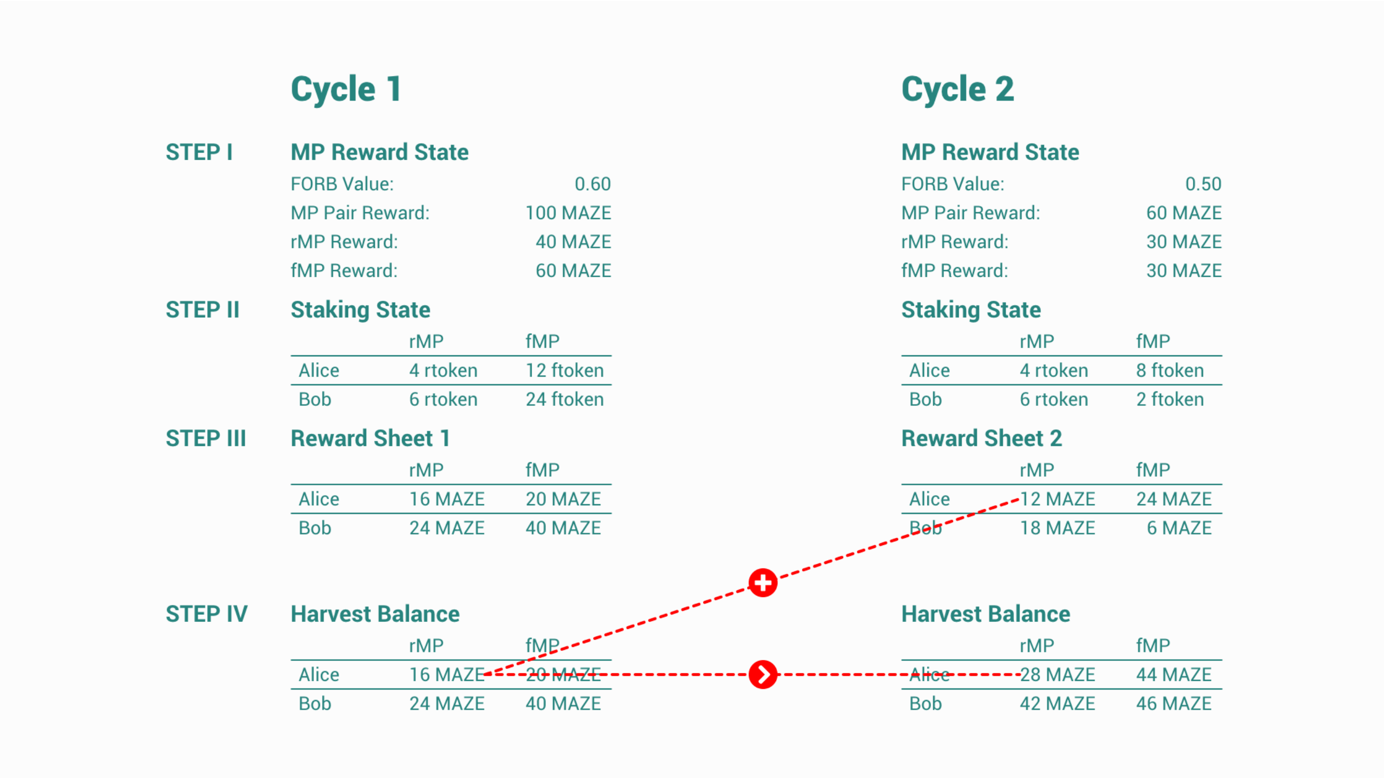
2.4.1 Posterior Stratified Yield Delivery

Implementing Maze's farming system is not a simple task due to the computational complexity of the yield calculation. These computations are expensive to do on-chain. 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. 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 finalized cycles.

The tables below intuitively show the reason:



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 to accumulate the reward for further cycles. If many users are farming in multiple MPs, the memory will cache a huge amount of data, which is impractical. Otherwise, the contract must store the income record to the blockchain ledger after each cycle, making the gas costs unreasonably high 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.

Therefore, the system is constrained by many trade-offs:

- 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 high 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 requests based on off-chain records, the system will lose its transparency and decentralization.

To remove the constraints, the Maze team has developed an innovative technology named **Farmbase**.

3 Implementing The First Dynamic Farming Pool: Farmbase Middleware Protocol

Farmbase Protocol is a by-product of development of the SHR algorithm. It is a middleware protocol that reads the farm status from a targetchain periodically performs computations and then updates the reward balance ledger for the users on the targetchain accordingly.

3.1 Foremost Targets

Farmbase is a transparent computation middleware service that works in the background, decentralized, independently tasked, and flexible. It can be deployed for many potential applications with structured income-distribution demands.

3.1.1 Works in the Background

Users will not be aware of interactions with the Farmbase middleware 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 transparent to users.
- Store and allow retracing of the 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 computations, which do not need to be deployed on the main network, to Farmbase. In 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 of the main task network. Accordingly, there will be no unnecessary gas costs for on-chain computations and no occupation of mempool and VM performance on the underlying blockchains.

3.1.4 Easy Iteration of Algorithms

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

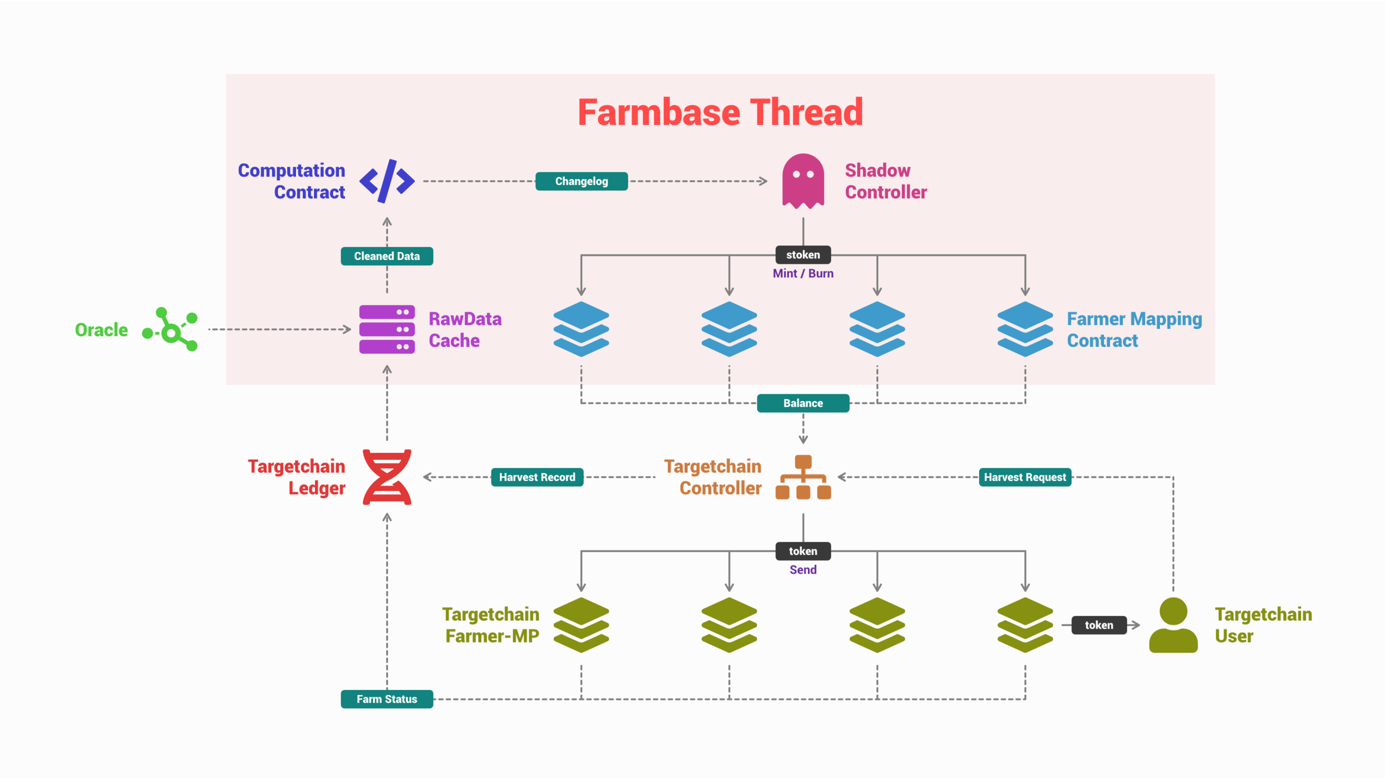
3.2 Introducing Farmbase

Farmbase Protocol is a multi-chain consensus network built with a large number of configurable use cases. A single-member blockchain is called a Farmbase Thread. A Farmbase Thread ought to serve one target protocol only. The reason for this is that, 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	The blockchain on which the underlying protocol is deployed.
RawData Cache	Collects and cleans the data from a targetchain before the computations.
Computation Contract	A set of smart contracts that compute reward distribution based on transparent functions.
stoken	Shadow token, a token on Farmbase Thread to map the reward asset on targetchain.
Shadow Controller	A smart contract that 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.

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 into this step. Rawdata Cache submits the pre-processed data to the Computation Contract. The customized income distribution algorithm will run on Computation Contract to generate the ledger of this cycle's income. 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. A new mapping contract will be deployed for every new farmer..

- Claiming Calls

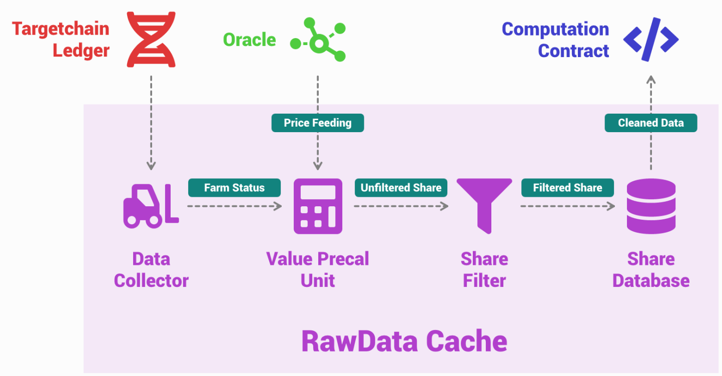
  When the user claims a harvest on targetchain, Targetchain Controller will read the stoken balance in the Farming Mapping Contract to verify the valid claiming amount, and then release the token to the user account.

  - 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. The release does not happen until the Targetchain Controller confirms the stoken burning. During the confirmation period, no subsequent claim requests can be made on the same Targetchain Farmer-MP address pair.

Value Precal Unit (VPU)

A farmer no longer active leaves a 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 a problem set up at RawData Cache's entrance. 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. Maximum or Minimum balance can both be used as filter rules. Min is often used to filter dust balance.

3.2.2 Farmbase Technical Design

Farmbase Protocol is an independent middleware project, and Maze is its first client. Farmbase Whitepaper will elaborate on its design and implementation in detail. In this part, the essential technical designs are briefly outlined.

- 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 Threads, deployed by targetchain project teams. The data of a Farmbase Thread is self-contained, and not synchronized in Global Domain or otherwise.

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

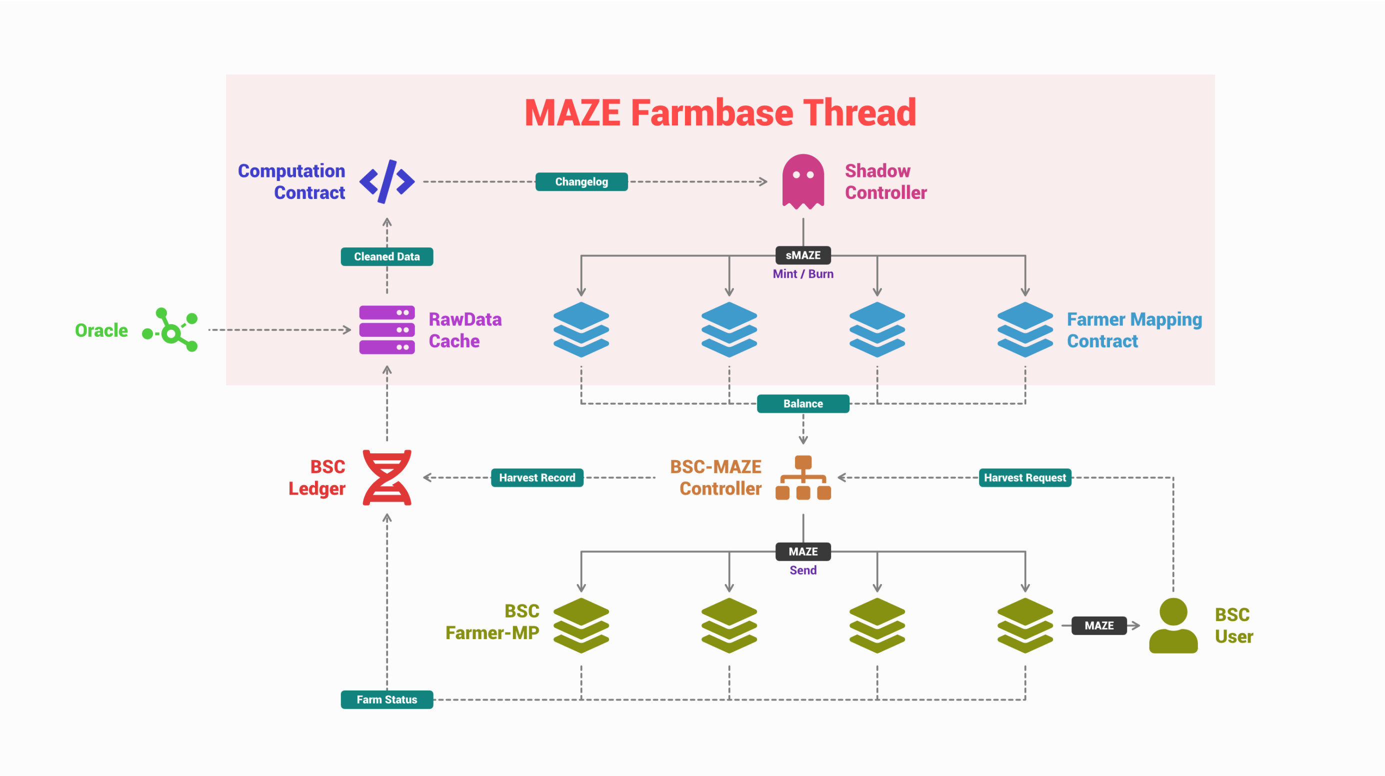
    To distinguish Global Domain and dedicated task domains, and to manage assets on different chains.
- Optional Storage Engine

  An internal PostgreSQL can assist with supermassive 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 — the experimental network named Neko — will be launched on BSC. Correspondingly, we take the Maze-BSC pair as an example to explain the collaboration flow of Farmbase and Maze.



3.3.1 Maze-BSC-Farmbase Collaboration Flow

This part outlines the working flow of farming income distribution on Maze when the Farmbase middleware is used.

- 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 lpMP 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 the income.

3.4 The Abilities of Farmbase

Farmbase's abilities are not limited to transparently computing farming income distributions. Farmbase also has potential to push the evolution of DeFi business ventures.

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 with the income distribution systems of the projects. The interference may cover:

- Quickly adding a speed factor of distribution for a selected MP fulfills the demand of enlarging or shrinking the reward.
- Filter the shares with rules, such as adding an invitation system that 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 attacked or facing an 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. Yield farming has emerged relatively recently, lacking infrastructure maturity. Still, we believe that as DeFi is widely adopted, farming will be the common financial business model in crypto markets. Therefore, it will be exceedingly productive 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, simplifying the construction of a visualized interface related to farming for project founders and users. It is easy to predict that as farming becomes a daily activity in the near future, every user can intuitively get analysis reports via the Farmbase data interface, thus making more intelligent asset management decisions.

3.4.3 Realizing The Scalability of Mining Pools

Besides 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 the 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

How to implement a special mining pool that requires users to stake the assets on two blockchains into two MPs respectively? Using Farmbase Thread to combine two blockchains' ledger data-feeding, the cross-chain farming shares can be easily joined logically.

In fact, it is this functionality of Farmbase that will allow Maze to operate across many blockchains at once.

Cross-chain Cycle Recalibration

Although the block interval is averagely controlled, 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 cross-chain lending protocols we see run their pools independently on each blockchain. In other words, though these are the same projects' various branches, they are separate. If Maze runs simultaneously on Ethereum, BSC and Polkadot parachains, will the SHR systems on the different protocol's economies break the curse of lonely islands? The answer is a definite yes. Farmbase was developed to facilitate the fusion of such economies.

4.1 Farmbase Degeneracy Of Cross-chain Identical MP

As the multi-chain ecosystem grows, more projects will issue their tokens across various blockchains. When Maze runs on different blockchains, it will definitely deal with underlying assets on different chains (typically USDT). Without Farmbase, the MPs for the assets on each chain would run separately with independent SHRs. Farmbase aims to change this imbalance.

Using Farmbase to merge the MPs of cross-chained identical assets is very simple — one only has 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 embedded on all blockchains. No matter how many networks are expanded upon, only one SHR is needed, and all networks share the same economy driven by one MAZE token.

Users don't have to know about these processes on the Farmbase middleware, but everyone can see that the yield rates of supplying identical assets are synchronized on every blockchain's Maze software.

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

4.2 Maze Branches On All Blockchains

When Maze merges MPs with Farmbase's help, it can deploy on blockchains in different development stages in a flexible way.

Each blockchain's assets, available liquidity and the development status are different. Building separate lending protocols on some not-so-developed blockchains or function-focused networks may cause problems. One of them is the lack of incentive feedback and liquidity shortage. Fortunately, Maze can easily deploy branches on these chains, build asset pools and their MPs, and then provide synchronized yield rates for these protocols' farmers through merged mining pools. Correspondingly, the protocols can instantly own a usable and incentive-synchronized banking system.

4.2.1 Supporting Fragment Branches

More importantly, similar to how a bank's branches run different business models in different regions, Maze can build fragment branches on different blockchains. Alongside supporting only permitted assets on some blockchains, Maze can provide only Reserve Pools (rMPs) or Funding Pools (fMPs) for specific assets on selected networks. SHR will distribute the all-chain-crossed MAZE tokens to these MPs based on the merged ledger of shares on Farmbase.

The cross-chained scalability of MAZE income distribution and the deployment of fragment branches is a unique value proposition of Maze. If the blockchain money markets are the large colonized star systems spread across the galaxy, creating a federated, galactic civilization then Maze is the star fleet that can sail to the most distant regions 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. Is there a fast and unlimited way of transferring such assets from one chain to another? Such cross-chain activity is still uneasy. If a user wants to make the migration, usually it is better to rely on centralized exchanges.

We will face such issues when deploying Maze across many blockchains. The users with assets deposited in many Maze pool branches will need to withdraw funds from the current branch and attempt to migrate via other means if heading to another blockchain.

Thanks to the combination of Maze and Farmbase, we are only one step away from eliminating such awkwardness.

5.1 Hawala: Transfer of Drawing Rights Instead of Assets

Hawala is an ancient way of transfer that 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 conducted in the following manner:

- Alice, needs to transfer her funds to the receiver Bob but has no direct pathway. Instead, she can only approach 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 correctly.
- David gives the funds to Bob. The transfer is completed.

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 transferring funds to Bob, which requires an evolved infrastructure. It often needs the development team of the underlying assets to build an atomic swap channel between specific blockchains. To ordinary users, the better cross-chain transfer is what hawala flow represents, transferring drawing rights. The real migration of underlying assets is exactly the work of reserve asset rebalance between two hawala brokers.

So to fulfill the goal, we need to build a system that transfers drawing rights, with asset pools spread everywhere.

Note: the hawala mode is illegal under many countries' financial regulation frameworks. On Maze, hawala is used as a concept that illustrates the principle of a drawing-right transfer protocol for 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 represent an essential infrastructure. From then on, Reserve Pools are no longer just a yield-rate contrast group, but they fulfill the liquidity demand of cross-chain withdrawal on each blockchain.

A user deposits a cross-chain underlying asset into the Reserve Pool, getting rtoken and rmtoken, and thus adds liquidity to the local pool. Doing so, a user also obtains the rights to withdraw funds on the other blockchains' Reserve Pools using the rtoken and rmtoken. The drawing rights of cross-chain assets are rtokens and rmtokens.

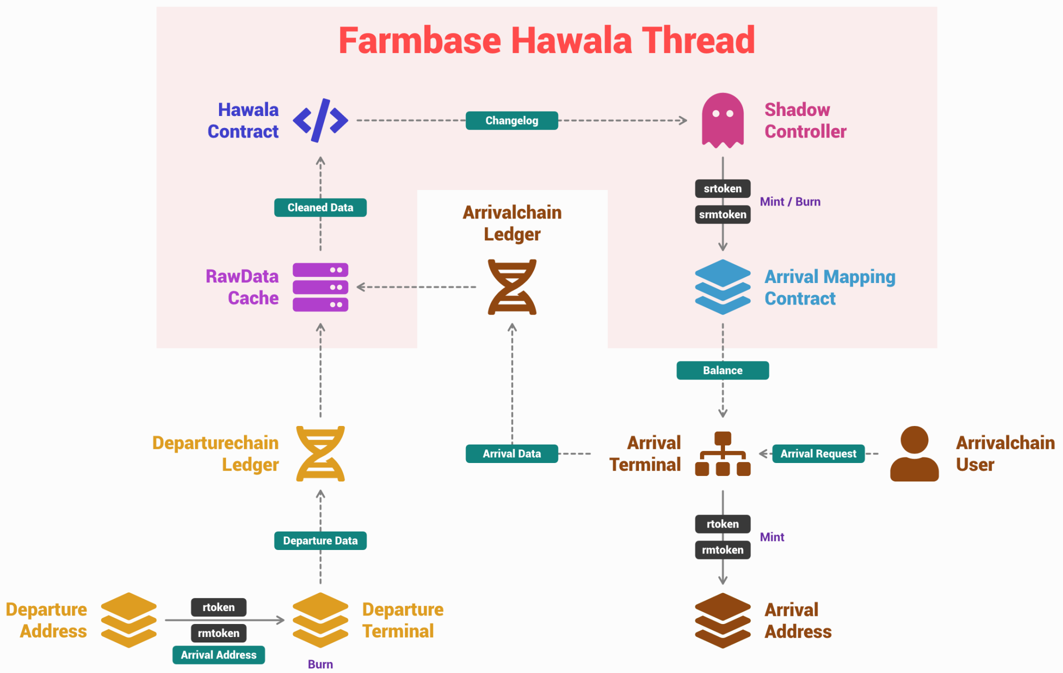
Then, the remaining issue to be resolved is: how to realize the cross-chain movement of rtoken and rmtoken? The answer is simple — using Farmbase.

5.3 Farmbase Hawala Thread

Farmbase Hawala Thread is a special Farmbase Thread. It is built separately, not running any farming computation tasks, but 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	Collects and cleans 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 are transfered to.
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

To understand Farmbase Hawala Thread better, we can imagine a transfer of drawing rights as taking a flight. The cross-chain asset drawing rights, rtoken and rmtoken, is your luggage, while the address you have on the other blockchain is the destination of this flight trip.

- Step 1

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

- Step 2

  You need to send your transport luggage to the check-in counter (send the rtoken and rmtoken into Departure Terminal). The airport personnel 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 dashboard, you just press 'Send').

- Step 4

Enjoy the flight and then get off (wait for a couple of seconds and open the dashboard with Arrival Address).

- Step 5

  Go to the carousel of Arrival Terminal and fetch your luggage (on the dashboard, you 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. The migration will not be done until the Arrival Terminal contract confirms the stoken burning.

Withdrawing with Cross-chain Drawing Rights

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

However, 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 incentivize on-chain hawala brokers' rebalance job of underlying asset liquidity. e.g., a higher portal fee can be charged on the transfer direction from low liquidity to surplus liquidity, and transferred to rebalance liquidity.

5.4 The Asset Stargates Connecting All Metaverses

Maze's drawing right transfer protocol has brought an alternative solution to requesting all project teams to deploy their own token's cross-chain swap channels over different networks. 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 Reserve Pools on different chains 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. The MAZE Reserve Pools in all branches can ensure the users can comfortably enjoy MAZE-related business on all chains. Additionally, they can 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. As Farmbase Hawala Thread must be integrated with homomorphic hidings, the drawing rights transferred between chains hold inherently fortified privacy. In the meantime, any third-party developer can build mixers supporting rtoken and rmtoken on local blockchains.

Please note: when a user transfers 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 other words, a user who is transferring the drawing rights remises his short-term income to other farmers.

6 A Stablecoin Combining CDP & Algorithm

The zero-interest lending system and SHR have brought a unique condition of forging a high-quality stablecoin for Maze. The suppliers can choose to forge ZUSD with their assets as collateral. ZUSD is a stablecoin that 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:

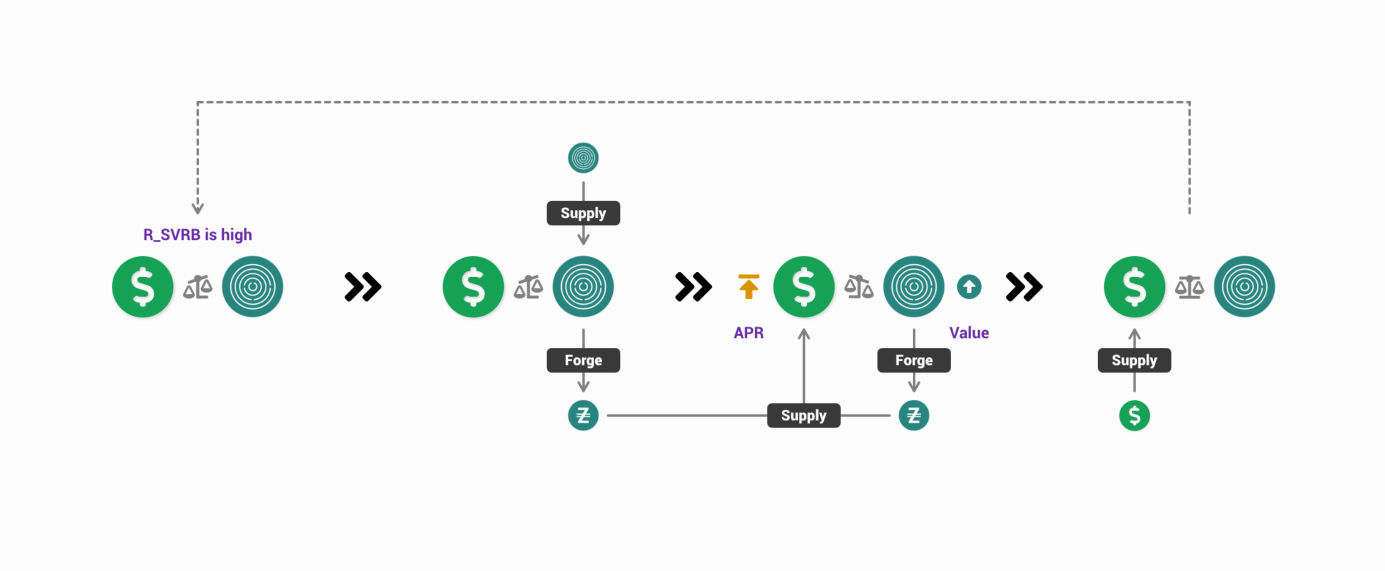
- Every Forge Pool is an over-collateralized zero-interest lending system.
- Every Forge Pool accepts only one type of asset staked.
- Every Forge Pool's liquidity is only for the creator.
- Forge Pools use the same loan risk management system like the lending protocol, including LTV, liquidation threshold, 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. 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 forging 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, the tokendynamics of MAZE become more complicated.



Foremost, MAZE's original tokendynamics loop still exists. With the launch of Forge Pools, MAZE will likely exit the activities in common lending markets. MAZE's Funding Pool supply should migrate to Forge Pools as much as possible. Supplying MAZE in Forge Pools carries several advantages:

- The liquidity is only for the Forge Pool itself, so the supply does not risk the high occupation rate in the pool.
- As every Forge Pool is liquidated independently, more meticulous risk management is possible.
- Supplying MAZE in Forge Pools generates fmMAZE, so the original yield can still be captured.
- 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)

Moreover, as Forge Pools need to lock large amounts of MAZE, the price of MAZE is driven up due to a liquidity squeeze. If the MAZE price goes up, the existing Forge Pools will have additional forge credit,and thus provide more ZUSD liquidity. Because the risk-free rate increases with the rising price of MAZE, the best choice for suppliers is to keep supplying ZUSD.

Since the borrowed funds cost no interest, the better option for suppliers is to 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 assets. Its goal is to make a large proportion of MAZE quit circulation and turn it into a cornerstone reserve on all blockchains. SHR will maintain the ZUSD-MAZE tokendynamics loop to ensure fair value of ZUSD.

It is no secret that any decentralized stablecoin has inevitable systemic risks. If the value of Forge Pools falls radically in a drastic market fluctuation, or the market price of ZUSD shows abnormal deviation, severe financial incidents could happen. Therefore, a careful plan for mitigating these risks needs to be in place before ZUSD can become a vibrant part of Maze.

7 Conclusion: A Super Sovereign Banking Protocol Crossing All Blockchains

Maze is a protocol pioneering the zero-interest lending business model. We have expanded on the concept and proposed the world's first completely decentralized high-frequency yield rate consensus mechanism.

Maze is built with a dedicated, transparent, computing middleware service, an infinite cross-chain open banking model, a drawing right transfer protocol of cross-chain assets, a CDP-based, algorithmic stablecoin, and a non-linear tokendynamics system.

With this in mind, a fully developed Maze will perform this way: its branches operate non-custodial zero-interest lending and decentralized stablecoin issuing business inside each domain. Furthermore, they provide user-friendly, privacy-protected cross-chain deposit and withdrawal services across the entire cyberspace. Lastly, they unite all asset suppliers in the same yield-rate management system. Thus a Super Sovereign Banking Protocol Crossing All Blockchains is born.

8 Maze's Experimental Network - Neko

Akin to Polkadot's testing platform Kusama, before mainnet launching, Maze will deploy on an experimental network, named Neko. Neko will launch 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 the real world given controllable risks, and being a long-term trial platform for new functions in the future, which means that Neko always has a faster version evolution than Maze. Please note, as an experimental platform, the operation of Neko could be relatively centralized.

8.1 Overview

- Protocol Environment

  Thanks to the low-latency and low-cost blockchain offered as Binance Smart Chain, and the solid support of mainstream underlying assets in its ecosystem, we plan to deploy the Neko Experimental Network on BSC. Neko 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 thoroughly tested, which reduces the security risks of asset pool smart contracts, and the difficulty of an audit. However, as the Farmbase middleware is original, it will need to be thoroughly audited to ensure safety of the protocol.

8.2 Neko V1 Function Configurations

- Main Business Functions

  Include Zero-interest Funding Pools, Reserve Pools, an 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 Considerations

    In the initial 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 a 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.

    In the future, a zkp-crosschain component will replace the off-chain relay service.

- Underlying Asset Types Initially Supported

  BUSD, USDT, BNB, ETH, BTCB on BSC mainnet, and the primary BEP-20 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/farm

8.4 The Ignition Procedure

- Prerequisites For Launching

  Since SHR takes the value ratio of stablecoins and NEKO in USP as reference, before launching the MP, some initial NEKO deposit must be made. 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

- Initial SHR Offering of NEKO

  Initial SHR Offering (ISO) is the innovative launching procedure exclusively operating on Maze/Neko core protocol.

  Before starting the NEKO ISO, the founder locks up initial NEKO-BNB liquidity in Pancakeswap, which provides price feeding to SHR. The team must also deposit a certain amount of the new token into Neko Network and turn into the first farmer of NEKO.

During this ISO, all money suppliers on the protocol will be rewarded with NEKO based on the SHR algorithm, and they must compound the rewards. The founder's NEKO deposit will be automatically burnt as users' rewards are re-staked. Once the users fully replace the founder's deposit, the ISO is completed, and NEKO's withdrawal starts opening.

9 Supplements

9.1 Disincentive of Funding Pool Occupation

Regarding the borrow function, a disincentive measure is suggested to help reducing the extreme occupation ratio in Funding Pools, which is a borrow fee.

Borrow fee is charged for every loan operation when it is activated. The borrow fee rate remains zero when the occupation ratio in a Funding Pool is low, and activates when the ratio reaches 90%. The borrow fee rate will climb from 1% to 10%, as the occupation ratio goes from 90% to 100% (parameters adjustable by governance).

For example, if Alice attempts to borrow 100 USDT when the USDT funding occupation ratio is at 90%, she will only get 99 USDT after requesting the loan but carry a 100 USDT debt. The 1 USDT borrow fee is collected by the protocol vault instantly.

10 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		A metric to examine if an account should be liquidated.
Oracle		A middleware mainly used to feed price data.
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 Regulators	SHR	MAZE's offering distribution algorithm, an adaptive yield-rate 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 loan initiation mechanism invented by Maker, managed by smart contracts running on Ethereum.
Forge Pool		CDP on MAZE, a type of isolated Funding Pools to forge ZUSD.