MB-based StaFi DAO development solution

1 Technical requirements

1.1 Reference library

https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/contracts/utils

1.2 Contract Data Layer

https://doc.subquery.network/

1.3 Related technical documents

Node running instructions:

https://docs.moonbeam.network/cn/node-operators/networks/full-node/

Collector Description:

https://docs.moonbeam.network/cn/node-operators/networks/collator​​​or/

Pledge Description:

https://docs.moonbeam.network/en/tokens/staking/stake/#how-to-nominate-a-collator​​​

Precompiled interface description:

https://docs.moonbeam.network/cn/tokens/staking/stakingprecompile/

Precompiled interface code:

https://github.com/PureStake/moonbeam/blob/master/precompiles/parachain-staking/StakingInterface.sol

https://github.com/PureStake/moonbeam/blob/master/precompiles/author-mapping/AuthorMappingInterface.sol

Reference Code:

https://github.com/hyd628/OneBlockCourseScripts/blob/main/Week11/NominationDAO.sol

2 factories

2.1 System parameters

Governance Template Address

Pool Template Address

Faucet Template Address

Airdrop template address

Reward Template Address

2.2 Implementing functions

Initialize DAO

1. Generate DAO governance

2. Generate DAO pledge pool (including borrowing stkToken)

Define symbols and names, refer to DAO governance, and set the pool address (stkToken) for DAO governance

3. Generate DAO airdrop pool (including retToken)

Define symbols and names, refer to DAO governance, and set the airdrop address (retToken) of DAO governance

4. Generate DAO reward pool

Set the reward address for DAO governance

Templates can be upgraded

3 Governance

3.1 System parameters

DAO technical service reward ratio: default 2%, minimum 1%

Collector's technical service ratio: default 12%, minimum 10%

Pool investment lower limit: the default is 100, when the investment lower limit is reached, Pool starts timing and governance

Pool investment upper limit: default 500000

Minimum investment per person: default 1

The minimum number of votes for voting: 51% by default, one-person multiple-vote system

Minimum allocation of rental income: default 1

Time limit for starting calculation of rental income and airdrop: default 2

Pool minimum redemption time limit: default 2

Time limit of zero income for collectors and principals: 3 days by default

Collector and client retToken margin ratio: default 1:0.1

stkToken (pool): voting rights

3.2 Implementing functions

sudo manually set system parameters

Automatically remove sudo and transfer governance rights to the DAO community when the set conditions are met

Investors have governance proposal rights

Need to set start and end time

Investors with a certain investment time limit have the right to vote on governance proposals: for example, 30 days

One vote per governance

Governance classification:

1. Internal governance, modify DAO parameters, and execute automatically

2. External governance, the proposal does not modify DAO parameters, but can be associated with external proposal files

If the minimum number of votes is met and the proportion of affirmative votes reaches the set proportion, it is considered to have passed the vote.

4 Pools

4.1 Issuing Credentials

stkToken

proof of ownership

transferable

detachable

Destructible

4.2 System parameters

Governance address

4.3 Implementing functions

sudo manually set system parameters

Pledge Token, mint OCToken and ECToken 1:1

Reach the lower limit of investment and the lower limit of the number of people Pool to open the leasing function

lease

1. Generate a collector faucet

Set the lease time limit and the collector's technical service reward address, calculate the required retToken (including a certain percentage of the deposit), and obtain the pledge vote by destroying the retToken at 1:1

2. Generate a client faucet

Set the collector address and lease time limit, calculate the required retToken (including a certain percentage of the deposit), and obtain the pledge vote by destroying the retToken at 1:1

redemption

Set the time limit for redemption and unlock, and investors can redeem as required

The stkToken:Token trading pair can be established in the third-party swap to solve the liquidity shortage problem of the pool pledge pool Token

5 Airdrop

5.1 Issuing Credentials

retToken

certificate of use

transferable

detachable

Destructible

5.2 System parameters

Governance address

Airdrop time unit: such as 2, 1 airdrop unit every 2 days

ertToken airdrop ratio: default 1:0.2

Calculated based on the unleased Token, airdrop to the address holding stkToken

5.3 Implementing functions

Meet the investment time limit and airdrop the corresponding proportion of retToken

6 Reward

6.1 System parameters

Governance address

Technical Party Dividend Address

6.2 Implementing functions

Receive Collector Rewards and Delegator Rewards

Start to distribute rental income when the minimum reward amount is reached

The rental income is automatically converted into stkToken and issued to the corresponding account

7 Faucet

7.1 Collectors

System parameters

Reward address

Set the lease time limit: such as 30, 60, 90

retToken pledge ratio: default 1:1

OwnerAddr: Collector's technical service reward address

authorAmount: Token required for author ID mapping binding

implement function

Reserve authorAmount

Burn and mortgage retToken to activate collectors

Mapping and binding wallet rewards

Zero revenue penalty, destroy retToken and force lease return

Rental income automatically enters Reward

When the lease time expires, the leased votes are automatically returned to the Pool, and the mortgaged retToken is returned to the user

Ballots can be rented multiple times

7.2 Principal

System parameters

Reward address

Set the lease time limit: such as 30, 60, 90

ECToken pledge ratio: default 1:1

Collector's address

implement function

Burn and mortgage retToken to activate the delegator

Zero revenue penalty, destroy retToken and force lease return

Rental income automatically enters Reward

When the lease time expires, the leased votes are automatically returned to the Pool, and the mortgaged retToken is returned to the user

Ballots can be rented multiple times