Datamall Chain

A Decentralized Storage Exchange Network

D M C

Technical Yellowpaper

V 1.0

DMC FOUNDATION

August 2022

1. Project Overview

1.1. Project Overview

DataMall Chain puts forward a decentralized storage trade algorithm and a network mechanism to connect data storage needs with data storage services, and uses the Proof of Storage Service (PoSS) algorithm for consensus, which can effectively converge various decentralized storage resources and present them to data storage demanders, ensuring that both the data storage supply side and the data storage demand side provide real data storage transactions. At the same time, the ecosystem governance token DMC as a media and a value carrier of storage service transactions can fully mobilize and leverage the resource aggregation advantages of the decentralized storage trading market.

As the governance token of Datamall Chain, Datamall Coin (DMC) can be obtained through mining (i.e., providing storage space and consuming storage space). The total supply of DMC is 1 billion, with 4 decimal places. To make DMC allocation more reasonable and better respond to various extreme situations, two auxiliary tokens (i.e. PST and RSI) are introduced in the process of the output, allocation and storage service transaction of DMC, which makes the storage space transaction market more stable. PST and RSI can only be used inside DataMall Chain and will not circulate in the market.

PST is introduced as Proof of Service Token, which refers to the proof of the MP'S storage service capability. Based on the quantity of PSTs minted by staking DMC, the PoSS (Proof of Storage Service) consensus algorithm generates a corresponding amount of voting power in proportion and ranks the nodes accordingly, so as to continuously select a certain number of miners who proactively provide storage services as consensus nodes.

Because the number of miners and the scale of storage delivery are dynamically changing, there is no fixed DMC reward for storage trading and delivery efforts. Therefore, a Real Storage Incentive (RSI) is introduced and a trading relationship between RSI and DMC is brought in. RSI can be swapped for DMC through continuous repurchase.

RSI is designed as an auxiliary token to measure community contribution, which is similar to the proof of work. The reason why community contribution is not rigidly linked to rewards mainly lies in the consideration to resist the impact of DMC price fluctuations on community incentives, so as to form a more stable reward mechanism. For example, in a certain period, DMC price fluctuations may result in a sharp increase in the number

of PSTs, and the total amount of community contribution will increase too. If community contribution is rigidly bound to rewards, the distributable DMC for unit community contribution will fall substantially. This will directly harm the enthusiasm of the community. When the non-real time distributed RSI is adopted for reward distribution, miners may choose to exchange their rewards for DMC at an appropriate time, which will have a positive effect on stabilizing the RSI/DMC exchange price.

At the same time, because RSI and PST are not freely circulating tokens, their amount of circulation is also directly linked to community contribution. Therefore, although third parties may still do evil, the risk of being sniped at is nevertheless relatively controllable. However, the risk of the arbitrage by miners for more profit still exists, which may be conducive to price stabilization.

Free trading will not increase the opportunity of doing evil. On the contrary, free trading will fragment arbitrage attempts, which helps to detect and eliminate risks early. By comparison, the distribution model of a fixed design will be solidified and accumulate risks, and finally be sniped at single points.

1.2. Role Description

The project includes the following main roles. To facilitate communication and understanding, these terms are explained as follows:

Miner the consumer (MC): MC is the consumer of storage capacity by purchasing PST on the platform and is also the verifier who initiates storage challenge during the transaction of storage service.

Miner the provider (MP): MP is the provider of storage capacity by selling PST on the platform to earn DMC and is also the service provider who accepts storage challenge during the transaction of storage service.

Limited Partner (LP): The DMC investor invests a certain amount of DMC on the MP. The MP mints PSTs by staking the invested DMC. When LP claims the staked DMC, the smart contract will calculate the profit based on the proportion of investment.

1.3. Interpretation of Terms

1.3.1. Storage Challenge

Challenge handling charges: The MC and the MP need to pay 1:1 challenge handling charges during the storage challenge and the challenge handling charges are paid into the repurchase account.

Default fine: If an MP fails to respond within the specified time and there is no arbitration, it will be deemed as default by the MP, in which case the MP needs to pay a default fine.

Default fine handling charges: The part of the default fine that is paid to the repurchase account.

Arbitration handling charges: Refers to the charge 100 times the challenge handling charges that one party needs to pay in circumstances such as malicious arbitration by an MP (i.e. not responding to direct arbitration) or successful arbitration due to abnormal challenges instituted by an MC. The charges will be paid to the repurchase account.

The details of storage challenge will be illustrated in chapter 3.

1.3.2. Stake Rate

m: Benchmark stake rate, the minimum stake reserve ratio that must be met for minting PST.

m': The stake rate that MP may define by themselves, which is equivalent to the ability to compensate. When setting m', the MP only needs to make it equal to or greater than the benchmark stake rate. Each modification should be ± 10%, and the modification can be made every 7 days.

r: Current stake rate. The calculation formula is:

$$r = \frac{DMC}{PST \times p}$$ (Assuming that 1 PST = p DMC)

P: the average transaction price of PST in the past seven days, the price is in DMC.

2. Smart Contract

2.1. PST and PST Maker Contract

2.1.1. PST（Proof of Service Token）

PST is the proof of service token. 1 PST represents a standard unit of storage service, that is, 1 PST corresponds to the storage service capacity of 1G for 7 days. MPs need to stake DMC through the PST Maker Contract to mint PSTs.PST cannot be transferred. And the estimated total supply of PST is 10 billion, with 0 decimal place. When trading PST, the system will provide one reference price, which is (tentatively) the average price of all transactions in the previous 7 days.

The PoSS consensus algorithm uses the quantity of the minted PST by staking DMC to proportionally generate the corresponding number of voting rights, and ranks the nodes accordingly to continuously select a certain number of MPs who actively provide storage services as the consensus nodes and give them incentives according to the reward rules.

2.1.2. PST Maker Contract

Anyone may become MPs after staking DMC and choosing to mint PST, and the system assumes that they have the storage capacity. Once they stake DMC, MPs can mint PST. Assuming that 1 PST = p DMC, the specific calculation formula is as follows:

$$PST = \frac{DMC}{m \times p}$$

Assuming that there is x staked DMC in an MP's account and that the price of PST is 1 PST = p DMC and the benchmark stake rate is m, then the maximum number of PST that the MP can mint is: $\lfloor \frac{x}{m \times p} \rfloor$

However, MPs may choose to mint partial PST according to their actual storage capacity. For example, when the DMC staked by the MP who can mint 4 PST at most, the MP may choose to mint only 3 of them for sale. In practice, the system will automatically calculate the total number of PST according to the market price of the DMC (whether it is invested by an LP or staked by an MP) based on the amount of DMC the MP can stake. And based on their needs, MPs may mint a certain quantity of PST for sale. As the PST price changes with market fluctuations, the stake rate r will also fluctuate in real time. When r is greater than m'*60% but less than m', the stake rate reaches a value at risk; When r is $<$ m'*60%, the stake rate reaches the liquidation value. The system will liquidate the current account. The specific rules of liquidation can be viewed in the liquidation mechanism.

2.2. Smart Contract and Investment Mechanism

In PST Maker Contract, all MPs can accept the investment from LPs (foundations or other investors). MPs can set the proportion of LP investment, the range is between 0-80%, and the DMC staked by MPs themselves should be at least 20% of the total.
The proportion of the LP and the MP as well as the total amount of staked DMC is recorded in the PST Maker Contract. In order to get an accurate number of reward and penalty, **the calculation formula to get the proportion of investment is as follow:**

$$\frac{asset}{staked_{old}} \times weights_{old}$$

2.3. Trading Contract and Storage Delivery Contract

When issuing a PST pending order, MPs may choose the shortest service cycle, and MCs can only choose from all the service duration MPs provide. MPs may set the amount of deposit (which should be a multiple of the order price). Once the deposit is set up, if an MC defaults, the deposit will be deducted. If the MC defaults within the service time (E.g., the MC does not have enough balance to pay expenses, the order will be cancelled by the contract). The deposit is deducted and paid to the MP (with $\frac{1}{1+m'}$ paid into the MP's balance, and $\frac{m'}{1+m'}$ paid into the MP's stake pool).

When purchasing services, MCs need to pay two expenses, which are respectively the storage service charges and the deposit. As for the storage service charges part, MCs must pay the charges of at least one cycle (7 days), and the contract will perform the deduction operation every 7 days and continue the services of the next cycle. In the service process, MCs can recharge DMC at any time to renew their contract, or withdraw DMC at any time as needed.

When MPs still need to purchase services after their initially purchased services expire, they may choose to recharge DMC to renew their contract. Otherwise, the contract will automatically return the deposit to the account when all the service cycles end. MPs are not allowed to cancel services at their own initiative for the time being. If an MP breaches the contract, a default fine will be imposed on the MP.

After their initially purchased services expire, MCs can still renew their services, but there will no longer be binding on both parties. MCs may cancel the services of the next cycle in advance by withdrawing their pre-deposited DMC at any time.

2.4. Smart contract and Dividend Mechanism

MPs and MPs can get rewards through pending orders and trading, while LPs can obtain income from investments. PoSS and the storage transaction model in DMC ensure fair incentives.

2.4.1. RSI (Real Storage Incentive)

RSI is the reward for MCs, MPs and LPs and it is similar to the reward points in the system, with 8 decimal places and the total supply is 10 billion. MPs or MCs can obtain RSI by issuing news orders, trading, etc. The tradable token pair so far is: DMC/RSI. The amount of DMC swapped by RIS is fixed every day and RSI is periodically repurchased and burned.

2.4.2. Reward of Pending Orders

In the pending order stage, only MPs can receive rewards: RSI, which is calculated according to the duration of pending orders from the moment the MP starts the order until the MC bids the order and delivery is made. The MP also can collect RSI midway through the duration of the order, and the calculation will start from the collection moment until the order is finished. The formula is 1PST = m' · RSI.

For example, one PST can get m·1RSI. As m' is temporarily set to 2, so one PST can get 2 RSI. In other words, 1GB can get 2 RSI in 7 days.

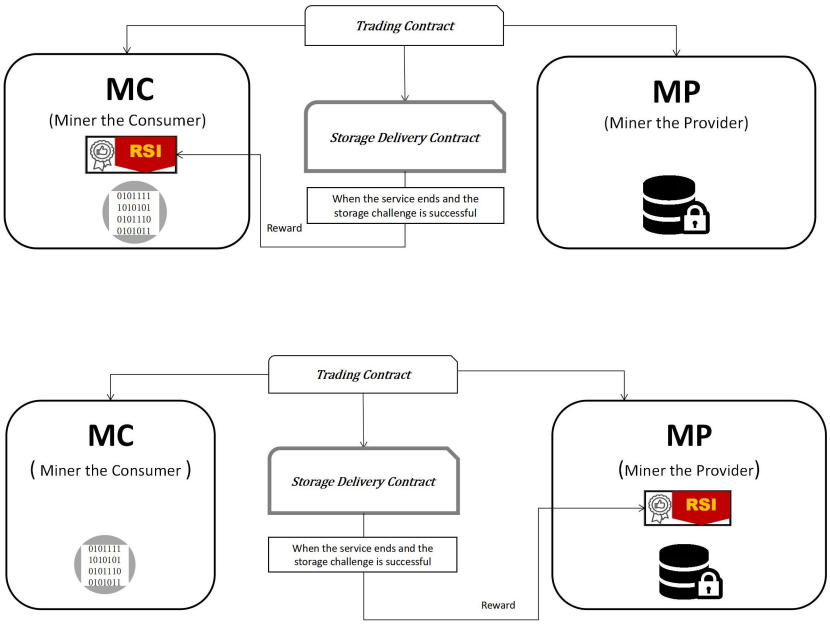
In the pending order stage, if an order of 1GB is pended for 1 day, it can get: 2÷7 = 0.28571429 RSI;

If it is pended for 1 hour, it can get: 2÷7÷24 = 0.01190476 RSI;

If it is pended for 1GB for 1 minute, it can get 2÷7÷24÷60 = 0.00019841 RSI;

And if it is pended for 1 second, it can get: 2÷7÷24÷60÷60=0.00000331 RSI. (There are precision transactions in the output figures of this document. Precise figures should follow the actual calculated results of the system.)

Currently, the incentive period of pending orders is 7 days. If it exceeds 7 days and the order is still not closed, there will be no incentives.



2.4.3. Reward of Delivery

In the delivery stage, both MCs and MPs receive RSI. The total amount of the reward is 2+m' RSI for each PST, of which MCs get 1 RSI and MPs get 1+m' RSI.

2.4.4. LP Investment Dividend

MPs use the invested DMC to mint PST and obtain the dividend, of which 1/ (1 + m') RSI is returned to MP's account and (m') / (1 + m') RSI belongs to staked pool. When the LP or the MP claims the DMC, the claimed amount is calculated according to their corresponding weights, and the calculation formula is:

$$staked_{total} \times \frac{weights_{individual}}{weights_{all}} \times rate_{input\ proportion}$$

2.5. Smart Contract and Claiming Mechanism

When claiming DMC, LPs can claim the reserve without limitation.

When MPs claim their DMC, there are two conditions to be followed:

$$\begin{cases} \dfrac{weights_{miner} - weights_{miner\_sub}}{weights_{total} - weights_{miner\_sub}} \geq miner\_rate \\ r \geq m' \end{cases}$$

If MPs want to claim all the staked DMC, they can only do so after the LP has claimed the invested DMC and the available PST (not including those delivered and those in pending orders) has been completely burned. All the claimed DMC will be locked for three days, but they can be unlocked in the wallet.

When MPs first enter the market, they can get DMC through OTC trading and other ways. Apart from trading, MPs can also obtain the PST minting right by receiving investment.

2.6. Smart Contract and Liquidation

The r represents the current stake rate. To ensure the stability of the economic model, we require that the stake rate cannot be too low. When r is equal to m', the stake rate reaches the risk value; When r is < m'*60%, the stake rate reaches the liquidation value. When the stake rate of the system reaches the risk value, MPs can adjust their stake rate r above m'*60% to the safe range by burning PST, increasing DMC or performing other operations.

When the stake rate of the system reaches the liquidation value, i.e. r < m'*60%, the system will perform liquidation immediately. During the liquidation, the system will give priority to liquidate the PST in the balance and if r is still below the liquidation value, the system will cancel pending orders and get the PSTs back to liquidate. If the liquidation value has not been reached, the amount of DMC may be increased to adjust r greater than the liquidation value to the safe range.

Meanwhile, DMC in the PST Maker Contract of an equal proportion will be deducted during liquidation, currently the rate is 0.3, and the penalized DMC (the amount of staked DMC deducted during liquidation) will be transferred to the repurchase account. MPs may stake more DMC, or get DMC by collecting the investment income, so as to adjust r above the liquidation value. At the same time, when r < m', MPs cannot mint PST.

3. Storage Challenge

3.1. Rule Description

After MCs place an order and conclude a transaction with MPs, the two sides enter the challenge preparation stage. At this time, it is necessary for both sides to submit Merkel roots to reach consensus. Once consensus is reached, MCs will enter a 7-day delivery cycle.

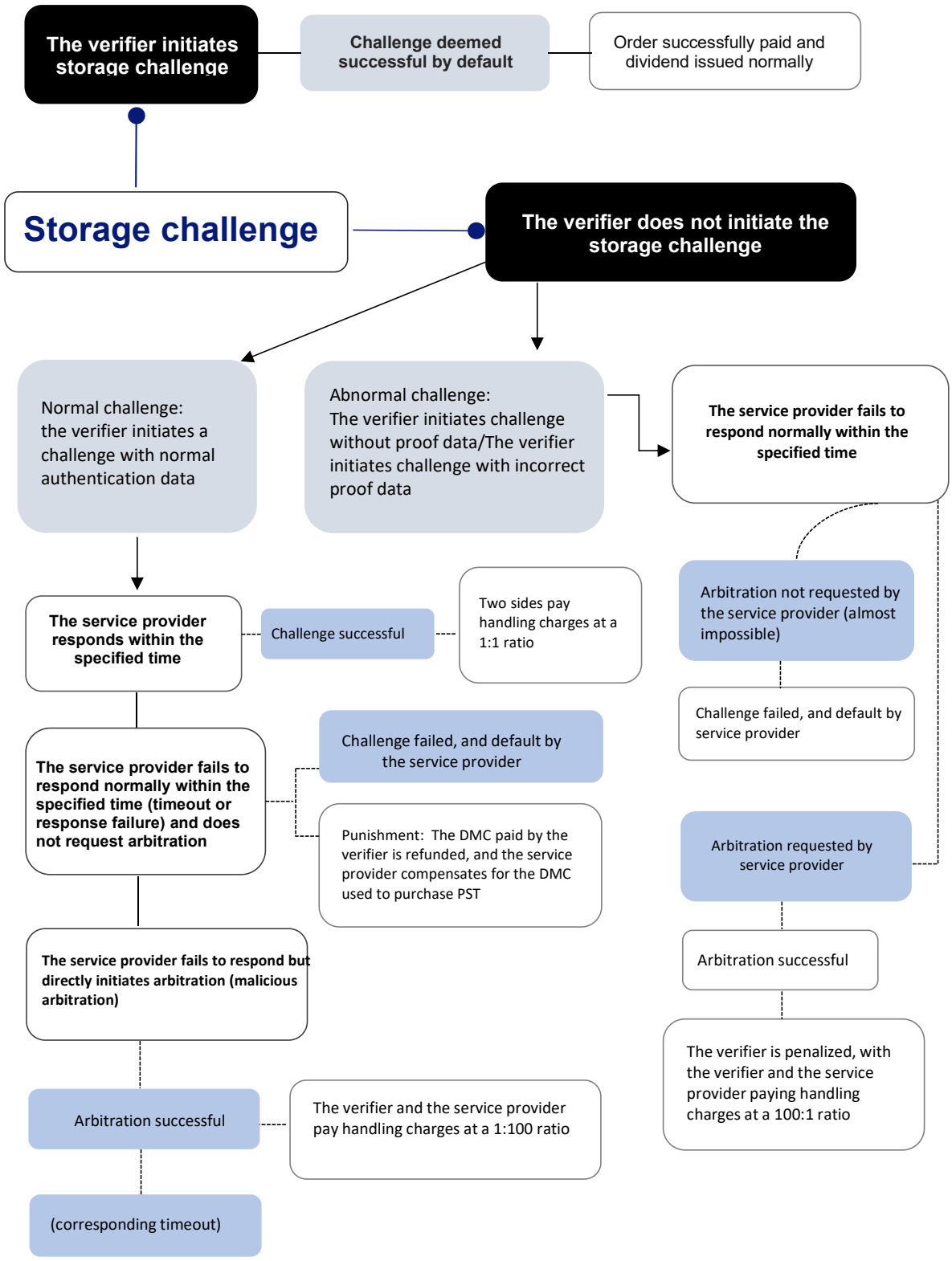
MCs may initiate multiple storage challenges in each delivery cycle (provided that the next challenge can be initiated only after the preceding one has been completed, and concurrent challenges are not supported). If no challenge is initiated within a delivery cycle (within 7 days), it is deemed a default successful storage challenge. During the service delivery period, MCs in principle, may cancel services of the next cycle (in which case the deposit will be deducted), but MPs are not allowed to cancel services.

The procedure can be stated as below:

Delivery stage → both sides submit Merkel roots to reach consensus → the challenge preparation stage →the MC decides whether initiates storage challenge or not

After formally entering a storage challenge, MCs may choose to challenge (for the time being, the service provider, i.e. MP, cannot initiate storage challenges). In the case of normal challenge, the MC initiates the challenge with proof data, and the service provider responds to it.

Challenges are divided into the following situations:


The verifier initiates storage challenge
Challenge deemed successful by default
Order successfully paid and dividend issued normally
Storage challenge
The verifier does not initiate the storage challenge
Normal challenge: the verifier initiates a challenge with normal authentication data
Abnormal challenge: The verifier initiates challenge without proof data/The verifier initiates challenge with incorrect proof data
The service provider fails to respond normally within the specified time
The service provider responds within the specified time
Challenge successful
Two sides pay handling charges at a 1:1 ratio
Arbitration not requested by the service provider (almost impossible)
Challenge failed, and default by service provider
The service provider fails to respond normally within the specified time (timeout or response failure) and does not request arbitration
Challenge failed, and default by the service provider
Punishment: The DMC paid by the verifier is refunded, and the service provider compensates for the DMC used to purchase PST
Arbitration requested by service provider
Arbitration successful
The verifier is penalized, with the verifier and the service provider paying handling charges at a 100:1 ratio
The service provider fails to respond but directly initiates arbitration (malicious arbitration)
Arbitration successful
The verifier and the service provider pay handling charges at a 1:100 ratio
(corresponding timeout)

3.2. Penalty Mechanism

Challenge handling charges: 10% of the PST unit price of the transaction order, i.e. 10% of the corresponding DMC price of a single PST. If an MC initiates a storage challenge, the handling charges will be deducted from the deposited DMC by the MC. The MP's part will be deducted from the DMC paid to the MP.

Default fine: For example, when an MP fails to respond within the specified time and there is no arbitration, it will be deemed that the MP defaults. In which case the default fine should be paid according the following rules: The DMC paid by the MC will be refunded to the MC, while an amount of DMC (based on the stake rate in real time) for purchasing the PST of the order will be deducted from the staked pool, 50% will be given to the MC as compensation and 50% will be given to repurchase account.

For example, if the MC has spent 500 DMC to buy 100 PST, then the 500 DMC will be refunded to the MC. Besides, the MP needs to compensate the MC with an amount of DMC that is m' * the total transaction price (m' is current m' in real time). In this example, this amount of DMC is m' * 500. Assuming that it is equal to X, then X DMC will be deducted from the stake pool of the MP, and 50% of it will be paid to the MC, and the remaining 50% will go to the repurchase account.

Default fine handling charges: The part of the default fine that is paid into the repurchase account.

Arbitration handling charges: Refers to the charge 100 times the challenge handling charges that one party needs to pay in circumstances such as malicious arbitration by an MP (i.e. not responding to direct arbitration) or successful arbitration due to abnormal challenges instituted by an MC. The charges will be paid to the repurchase account.

3.3. Technical Scheme

3.3.1. Role Description

Verifier: The party who pays for storage services in the storage service market. When a verifier selects a service provider to purchase storage service, it can use the storage challenge to confirm with the service provider that its data is securely stored within the service cycle.

Service provider: This role is the provider of storage services in the storage service market. After a verifier purchases services from the service provider, it will transfer its

data to the service provider. The service provider needs to respond to the storage challenge proof of the verifier within the service cycle.

Smart contract as notary: The scheme uses a smart contract as the notary of storage challenges.

3.4. Process Description

3.4.1. Stage 1: Storage Preparation

3.4.1.1. The verifier partitions the original data into blocks, calculates the Merkle Tree, and then sends the Merkle Root hash to the smart contract. At the same time, it also sends the original data to the service provider. It should randomly reserve several data blocks (data_block_1, data_block_2, etc.) and the Merkel tree in local for subsequent storage challenges; (note: for storage challenges packaged in Filecoin, the service provider needs to submit the Filecoin packaging certificate and the Merkel root in the form of memos);

3.4.1.2. Based on the original data provided by the verifier, the service provider partitions data and calculates merkle_1 in the same way, then queries the smart contract, verifies whether the hash (merkle_root_1) of the Merkle tree is consistent with the merkle_root_hash submitted by the verifier, and checks whether the number of PST purchased by the verifier can support the data size of the verifier;

3.4.1.3. If the service provider confirms that the merkle_root_1_hash of the Merkle tree is consistent with the merkle_root_hash submitted by the verifier in the smart contract, it will confirm that the Merkel root is valid and submit merkle_root_1 to the smart contract, and the service will start to run. Otherwise, the service provider will send the inconsistency conclusion to the verifier;

3.4.1.4. If the two parties cannot reach consensus on the data, either party may cancel the order. Once an order is cancelled, there will no dividend from a cancelled order. (This step is still to be decided, and no order cancellation is supported for the time being.)

3.4.2. Stage 2: Storage challenge

3.4.2.1. The verifier initiates a random storage challenge: randomly select a data block (say data_block_1) from the data blocks of the preparation stage, and send the data_block_1_id and the timestamp to the service provider;

3.4.2.2. The service provider receives the storage challenge from the verifier and responds within the specified time in accordance with the challenge requirements. The response result calculation formula is: hash (data_block_1 + timestamp);

3.4.2.3. The verifier receives the response of the service provider and verifies the service provider's response result: successful verification will be deemed as the service provider's response being valid; verification failure will be deemed as the service provider's response being invalid, in which case the verifier may initiate a challenge notarization;

3.4.2.4. If the service provider fails to send a response to the verifier or refuses to respond within the specified time, the verifier may initiate a challenge notarization;

3.4.2.5. If the verifier does not initiate a storage challenge, the contract will be settled automatically after the service expires.

3.4.3. Stage 3: Challenge Notarization

3.4.3.1. Challenge with Data

3.4.3.1.1. The verifier initiates challenge notarization: randomly extract a data block (say data_block_1), add timestamp to it, calculate the corresponding hash (hash (data_block_1 + timestamp)), and then submit the hash value, timestamp and data_block_1_id to the smart contract. The calculation formula is: hash (hash (data_block_1 + timestamp));

3.4.3.1.2. After the verifier initiates a challenge notarization to the contract, the service provider needs to respond within the specified time according to the challenge requirements. The calculation formula is: hash (data_block_1 + timestamp); and the specified time should be based on the block time;

3.4.3.1.3. The contract receives the response from the service provider, and verifies the verifier's challenge signature. Successful verification will be deemed as the service provider's response being valid; and verification failure will be deemed as the service provider's response being invalid;

3.4.3.1.4. Both sides of the challenge notarization need to pay an amount of challenge handling charges. The handling charge ratio between the two sides is 1:1;

3.4.3.1.5. If the service provider fails to send a response to the contract or refuses to respond within the specified time, the service provider's response will be deemed as invalid; and

3.4.3.1.6. If the service provider deems the verifier's challenge as invalid, it may initiate an arbitration request to the contract against the challenge.

3.4.3.2. Challenge with Designated Data ID

3.4.3.2.1. The verifier initiates challenge notarization: randomly designate a database ID (say data_block_1_id), and submit it to the smart contract;

3.4.3.2.2. The service provider needs to respond within the specified time according to the method of arbitration response, which is described below.

3.4.4. Stage 4: Arbitration

3.4.4.1. It the service provider is not satisfied with the challenge result and thinks that the verifier has done evil, it may request arbitration from the contract and submit the data_block_1 of the designated data block and the corresponding merkle_cut to the contract;

3.4.4.2. The contract verifies whether the original Merkel root is consistent with the submitted Merkel cut; if the two are inconsistent, the service provider is deemed as having breached the contract;

3.4.4.3. The contract verifies the Merkel cut provided; if the two are inconsistent, the service provider is deemed as having breached the contract;

3.4.4.4. The contract calculates the hash of the designated original data block provided, and confirms whether it is consistent with the hash of the corresponding leaf node in the Merkel cut; if the two are inconsistent, the service provider is deemed as having breached the contract;

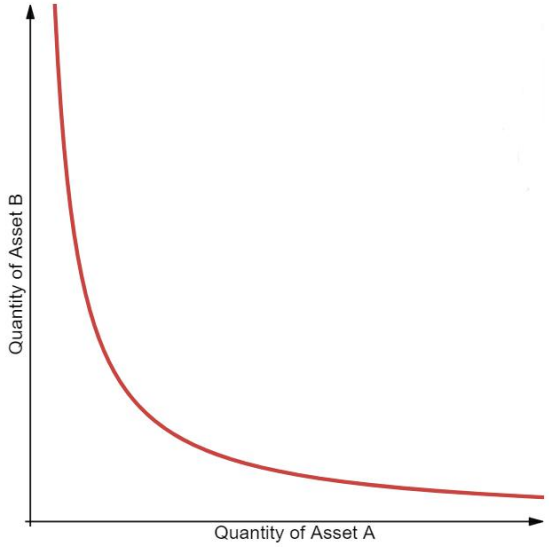
3.4.4.5. The contract verifies the Merkel root, the Merkel cut, and the data hash and the corresponding child node. If the results are consistent, it will prove that the service provider’s response is valid;

3.4.4.6. Specify that the sufficient handling charges should be paid by both sides before initiating a data ID challenge or arbitration. The verifier and the service provider need to pay handling charges to the contract at a ratio of 100:1;

3.4.4.7. If, based on the designated data ID challenge, the service provider is found to have breached the contract, a certain proportion of the order’s payment will be deducted from the fund pool and paid to the verifier.

4. DMCswap Trading Market

DMCswap adopts an automatic Market Maker system. This alternative method of adjusting asset price according to the asset supply and demand relationship utilizes a long-term mathematical equation. Its working principle is to raise or lower the asset price based on the proportion of tokens in each pool. It should be noted that whenever someone adds a new token A to DMCswap, the person must add a certain quantity of the selected Token A and an equal quantity of another token B to start the liquidity pool. The formula for calculating the price of each token is x * y = k, wherein the quantity of Token A is x, the quantity of Token B is y, and K is a constant value.



Currently, only RSI/DMC can be exchanged in DMCswap, DMC foundation will add the liquidity to the pool. For the time being, and only algorithmic matchmaking transactions are supported.

4.1. Handling Charges

4.1.1. DMCswap transaction handling charges: 0.3% after an algorithmic matchmaking transaction is reached.

4.1.2. There are no handling charges for removing and adding liquidity.

4.1.3. All handling charges are deducted from the currency after swapping. For example, when Currency A is swapped for Currency B, the handling charges will be deducted in Currency B.

5.Glassory

Arbitration handling charges: Refers to the charge 100 times the challenge handling charges that one party needs to pay in circumstances such as malicious arbitration by an MP (i.e. not responding to direct arbitration) or successful arbitration due to abnormal challenges instituted by an MC. The charges will be paid to the repurchase account.

Challenge handling charges: The MC and the MP need to pay 1:1 challenge handling charges during the storage challenge and the challenge handling charges are paid into the repurchase account.

Default fine: If an MP fails to respond within the specified time and there is no arbitration, it will be deemed as default by the MP, in which case the MP needs to pay a default fine.

Default fine handling charges: The part of the default fine that is paid to the repurchase account.

Limited Partner (LP): The DMC investor invests a certain amount of DMC on the MP. The MP mints PSTs by staking the invested DMC. When LP claims the staked DMC, the smart contract will calculate the profit based on the proportion of investment.

m: Benchmark stake rate, the minimum stake reserve ratio that must be met for minting PST.

m': The stake rate that MP may define by themselves, which is equivalent to the ability to compensate. When setting m', the MP only needs to make it equal to or greater than the benchmark stake rate. Each modification should be ± 10%, and the modification can be made every 7 days.

Miner the consumer (MC): MC is the consumer of storage capacity by purchasing PST on the platform and is also the verifier who initiates storage challenge during the transaction of storage service.

Miner the provider (MP): MP is the provider of storage capacity by selling PST on the platform to earn DMC and is also the service provider who accepts storage challenge during the transaction of storage service.

P: the average transaction price of PST in the past seven days, the price is in DMC.

r: Current stake rate. The calculation formula is:

$r = \frac{DMC}{PST \times p}$ (Assuming that 1 PST = p DMC)

Reserve: The amount of DMC for staking to mint PSTs.