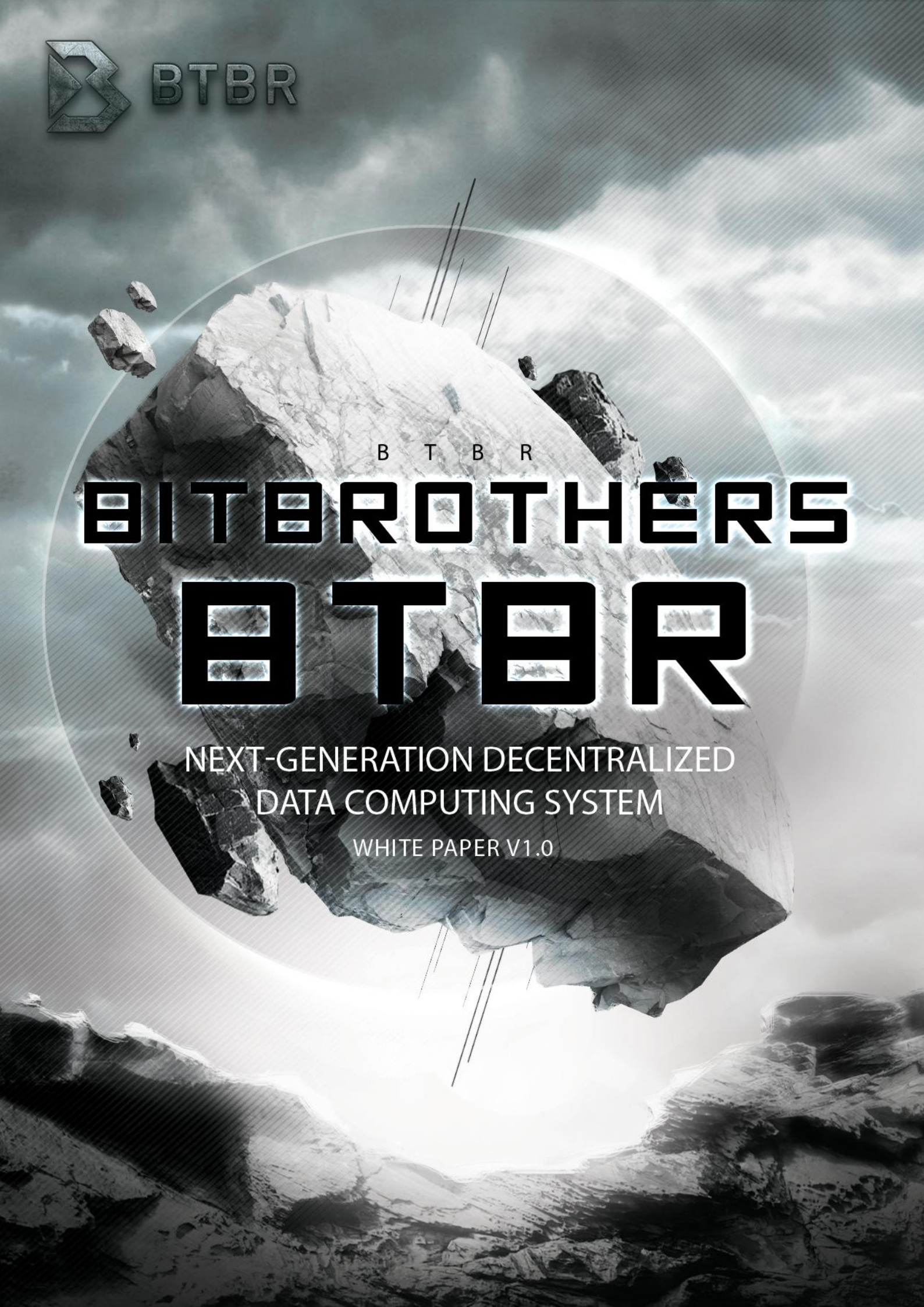
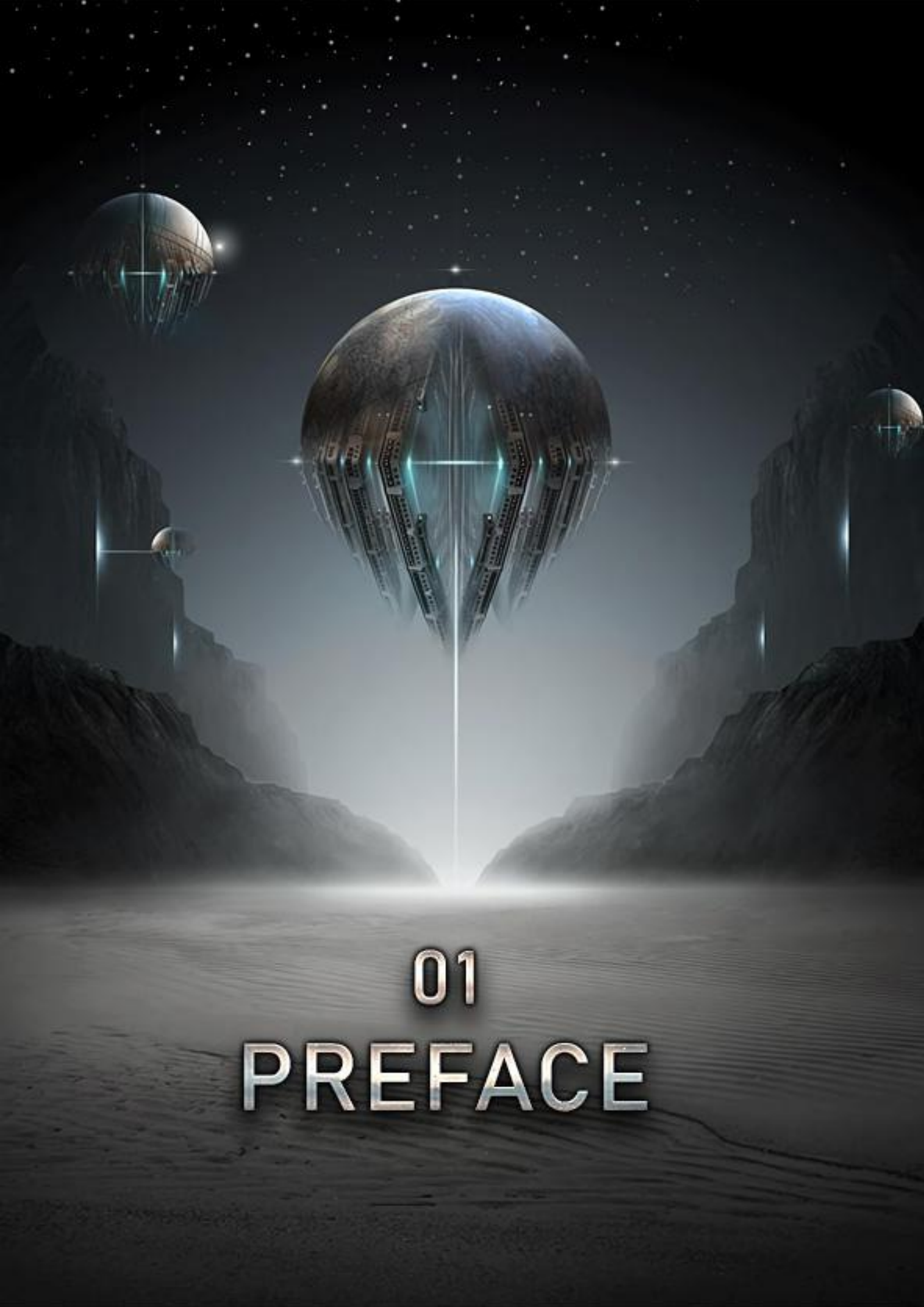

BTBR
B T B R
BITBROTHERS
BTBR
NEXT-GENERATION DECENTRALIZED
DATA COMPUTING SYSTEM
WHITE PAPER V1.0



01 PREFACE

Next-generation decentralized data computing system

Running through the original intention of Satoshi Nakamoto, evolving the concept of community autonomy, breaking the rules of social and economic imbalance, avoiding the possibility of concentration and manipulation of computing power, reshaping the entire mining industry, to truly realize the vision of Bitcoin decentralization.

Bitcoin in the opposite direction-weak decentralization

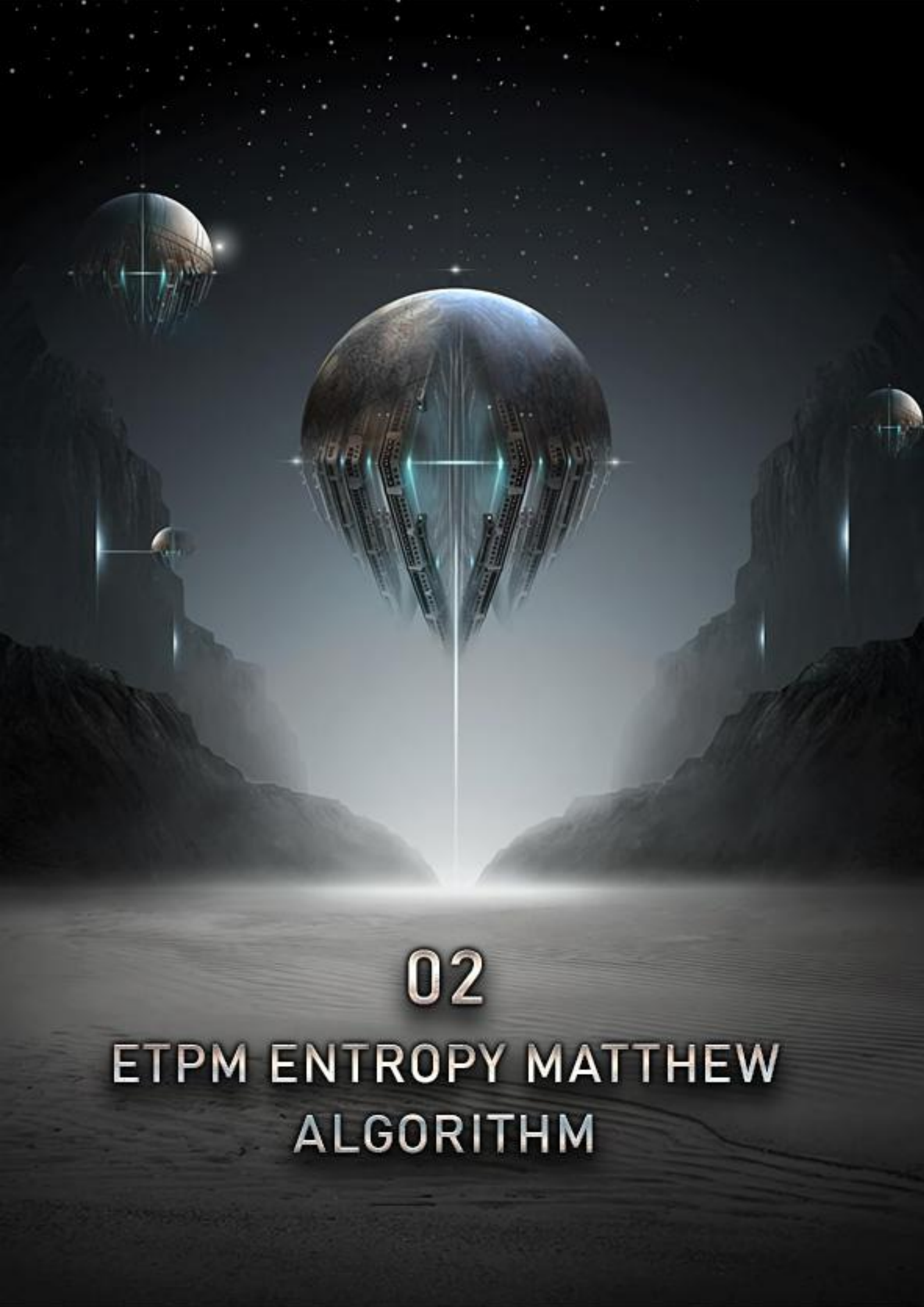
In September 2008, the financial crisis had just begun and it spread rapidly around the world. In this context, on October 31st, a cryptography geek Satoshi Nakamoto published a paper on the mailing list of the website "metzdowd.com (cryptopunk)" entitled "Bitcoin: A Point-to-Point Electronic Cash System", aiming to create a decentralized electronic transaction system. So far, Bitcoin has exposed various flaws, including overall low energy efficiency, poor or limited performance and overly concentrated computing power.

The working principle of the Bitcoin mining algorithm is to allow miners to calculate SHA256 on the slightly modified version of the block header again and again, until a node finally proposes a version whose hash value is less than the target value. However, this mining algorithm is susceptible to two forms of centralization. First of all, the mining ecosystem has been dominated by ASICs (application-specific integrated circuits), computer chips, which are designed for specific tasks in Bitcoin mining, so the efficiency has increased thousands of times. It means that Bitcoin mining is no longer a highly decentralized and equal pursuit and requires millions of dollars in funds to participate effectively. Secondly, most Bitcoin miners do not actually perform block verification locally. Instead, they rely on centralized mining pools to participate in mining. The first few large

mining pools indirectly control about 50% of the processing power in the Bitcoin network. This over-concentrated computing power runs in the opposite derection to Satoshi Nakamoto's original intention.

Vision and structure

Bit brothers will run through the original intention of Satoshi Nakamoto, to evolve the concept of community autonomy and break the rules of social and economic imbalance. Bit brothers integrates the ETPM entropy Matthew algorithm, using a random function to pass the cloud computing power server, so that the computing power node storage is sufficiently dispersed, and the possibility of concentration and manipulation of computing power are avoided, the true decentralized miner is redifined, and the entire mining industry is reshaped, finally the vision of Bitcoin decentralization will be truly realized.


02
ETPM ENTROPY MATTHEW
ALGORITHM

ETPM Entropy Matthew Algorithm

The purpose of the consensus mechanism is to make all nodes agree with each other in an environment where nodes do not trust each other, that is, trust each other. Then it verifies all transactions in the new block and adds the new block to the blockchain. It should be noted that the block will be added to the chain with the longest block height (see Blockchain Forks to understand how multiple chains exist at a certain point in time). Miners (dedicated computers on the network) perform computational work to solve complex mathematical problems to add blocks to the network, so it is called proof of work. As time goes by, mathematical problems become more and more complicated.

Verification requires to add the transactions in the blocks. The process of organizing these transactions in blocks in chronological order and publishing newly mined blocks to the entire network does not take much effort and time. The energy-consuming part is solving the "problem", linking the new block to the last block in the effective blockchain.

When the miner finally finds the correct solution, the node will broadcast it to the entire network at the same time and receive the cryptocurrency reward (reward) provided by the PoW protocol. With the arrival of more miners, the time required to mine new blocks will inevitably become shorter and shorter. It means that new blocks are discovered faster. In order to continuously find 1 block every 10 minutes

(it is the time required by Bitcoin developers to continue to stabilize and reduce the flow of new coins until the maximum number of 210 million is reached), the network periodically changes the difficulty level of mining new blocks .

Bit brothers uses the ETPM entropy Matthew algorithm, which is to break the centralization imbalance rule. For example, the entropy in an isolated system will only increase. Generally speaking, entropy is a measure of how the system can be arranged. Before the genesis block is mined, the Bit Brothers genesis block system files will be published in a decentralized encrypted network world.

At the same time, as the computing power increases, it will continue to promote the growth of computing power. Then, in the computing power world, entropy can balance order and disorder, and entropy will decrease when computing power is concentrated or increased. Conversely, entropy increases when the world of computing power becomes lower.

The Entropy Matthew algorithm of Bit Brothers can make the overall network computing power system reach an optimal balance. The computing power computing system is the safest, and the self-operation of decentralized nodes is more balanced.

When performing random hash operations, the proof-of-work mechanism introduces scanning for a specific value. For example, under SHA-256, a random

hash value starts with one or more zeros. Then as the number of 0s increases, the amount of work required to find this solution will increase exponentially, and checking the result only requires one random hash operation.

We add a random number (Nonce) to the block, which makes the random hash value of the given block appear as many zeros as needed. We try to find this random number until we find it. In this way, we have constructed a proof-of-work mechanism. As long as the amount of work consumed by the CPU can satisfy the proof-of-work mechanism, the information in the block cannot be changed unless a considerable amount of work is completed again. Since the subsequent blocks are linked after the block, if you want to change the information in the block, you need to re-complete the entire workload of all subsequent blocks.

At the same time, the workload proof mechanism also solves the problem of who is the majority in collective voting. If the majority of decisions are based on IP addresses, one vote for one IP address, then if someone has the power to allocate a large number of IP addresses, the mechanism will be destroyed. The essence of the workload proof mechanism is one vote for one CPU. The "majority" decision is expressed as the longest chain, because the longest chain contains the largest amount of work. If most of the CPU is controlled by honest nodes, then the honest chain will be extended at the fastest speed and surpass other competing chains. If you want to modify an existing block, the attacker must re-complete the workload

of the block plus the workload of all blocks after the block, and eventually catch up with and surpass the workload of honest nodes. We will prove later that if a slower attacker tries to catch up with subsequent blocks, the probability of success will decrease exponentially. Another problem is that the computing speed of hardware is increasing at a high speed, and the degree of node participation in the network will fluctuate. To solve this problem, the proof-of-work difficulty (the proof-of-work difficulty) will be determined using a moving average target method, that is, let the difficulty point to make the speed of generating blocks per hour be a predetermined average. If the rate of block generation is too fast, the difficulty will increase.

Hubs and partitions

Here we will describe a new model of decentralization and scalability. The BTBR network runs numerous blockchains through the ETPM entropy Matthew algorithm mechanism. Although the goal of the existing proposal is to create a "single blockchain" that contains all transaction orders around the world, BTBR allows multiple blockchains to run in parallel while maintaining interoperability.

On this basis, the BTBR hub is responsible for managing many independent blockchains called "partitions" (sometimes called "shards", referring to the well-known database expansion technology "shards"). The shards on the hub will continuously submit the latest blocks, which allows the hub to synchronize the status of each partition. Similarly, each partition will be consistent with the state of the hub (but the partitions will not be synchronized with each other, unless it is implemented indirectly through the hub). By issuing a Merkel certificate to prove that the message is accepted and sent, the message is passed from one partition to another. This mechanism is called "inter-blockchain communication", or simply "IBC" mechanism.

Any partition can become a hub on its own to create a non-cyclic graph. But for clarity, we only describe this simple configuration with only one hub and many non-hub partitions.

Hub

The BTBR hub is a blockchain that carries a variety of distributed ledger assets, in which tokens can be held by individuals or partitions themselves. These tokens can be transferred from one partition to another through a special IBC packet, the "coin packet". The hub is responsible for keeping the total amount of various tokens in each zone unchanged. IBC token data packet transactions must be executed by the sender, hub and block recipient.

Because the BTBR hub plays the role of the central token ledger in the entire system, its security is extremely important. Although each partition may be an ETPM entropy Matthew algorithm blockchain - only required to pass 4, (or fewer validators to ensure security without the need for Byzantine fault-tolerant consensus), the BTBR hub must pass the global Decentralized verification group to ensure security, and this verification group must be able to withstand the most serious attacks, such as regional network split or attacks initiated by the state.

Partition

The BTBR partition is an independent blockchain that can exchange IBC messages with the BTBR hub. From the perspective of the hub, a zone is a multi-signature account with multiple assets and dynamic membership,

which can be used to send and receive tokens through IBC data packets. Just like a cryptocurrency account, a partition cannot transfer tokens in excess of its holdings, but it can receive tokens from others who own tokens. Partitions may be designated as the "source" of one or more tokens, giving them the power to increase the supply of tokens.

The BTBR of the BTBR hub may be used as a bargaining chip for the district verifier to connect to the hub. Although under the ETPM entropy Matthew algorithm bifurcation responsibility system, the double-spending attack in the partition will cause the number of BTBR to decrease. However, if there are more than ⅔ votes in the partition that have the Byzantine problem, the partition can be submitted for invalid status. The BTBR hub does not verify or execute transactions submitted to other partitions, so it is the user's responsibility to send tokens to reliable partitions. In the future, the management system of the BTBR hub may solve the problem of partition failure through improvement proposals. For example, when an attack is detected, the transfer of tokens initiated by some partitions (or all partitions) will be suspended to realize emergency disconnection (ie, temporarily suspend the transfer of tokens).

Cross-chain communication-IBC

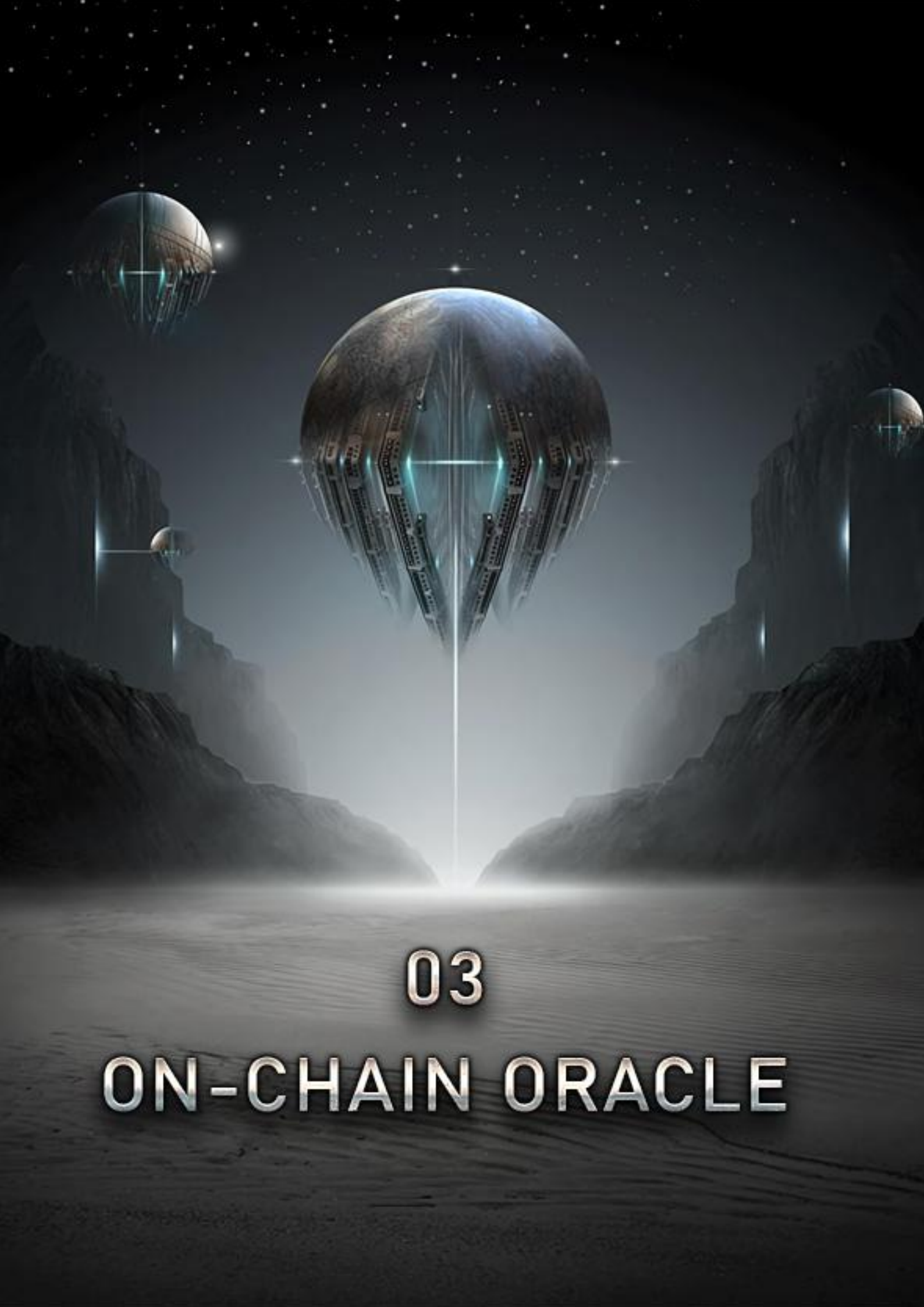
Now we will introduce the communication method between the hub and the partition. If there are now three blockchains, namely "Partition 1", "Partition 2" and "Hub", and we want "Partition 1" to generate a data packet and send it to "Partition 2" through the "Hub". In order to transfer a data packet from one blockchain to another, a certificate needs to be issued on the receiver's blockchain to confirm that the sender has initiated a data packet to the specified destination. The proof to be verified by the receiver must be consistent with the block header of the sender. This mechanism is similar to that used by side chains, which requires two interacting chains to "know" the situation of the other party by bidirectionally transmitting the existence proof data element (transaction).

The IBC protocol can naturally be defined as the use of two transactions:

One is the IBCBlockCommitTx transaction, which allows the blockchain to prove its latest block hash value to any observer. The other is the IBCPacketTx transaction, which can prove that a certain data packet is indeed sent by the sender's application to the hash value of the latest block through the Merkle-proof mechanism.

By separating the IBC mechanism into two separate transactions, the IBCBlockCommitTx transaction and the IBCPacketTx transaction. We can allow the local fee market mechanism of the receiver chain to decide which data packet to recognize, while at the same time ensuring the complete freedom of the sender, allowing it to determine the number of data packets that can be transmitted.

In the above case, in order to update the block hash of "Partition 1" on "Hub" (or the block hash of "Hub" on "Partition 2"), the "Partition 1" block of the IBCBlockCommitTx transaction must be updated. The desired value is published to the "hub" (or the hash value of the "hub" block of the transaction is published to the "partition 2").


03
ON-CHAIN ORACLE

On-chain oracle

It has on-chain components composed of three main contracts: reputation contract, order matching contract and aggregation contract.

The reputation contract tracks the performance metrics of the oracle service provider. The order matching smart contract takes the recommended service level contract, records the SLA parameters, and collects bids from the oracle supplier. Then, it uses reputation contracts to screen bids and end the oracle SLA. The aggregation contract collects the replies from the oracle supplier and calculates the final summary result of the BTBR query. It also feeds back the measurement data of the oracle supplier to the reputation contract. The BTBR contract is designed in a modular manner, allowing users to configure or replace as needed.

The workflow on the chain has three steps:

1. Oracle screening

2. Data report

3. Summary of results

Oracle screening

The oracle service purchaser specifies the requirements that constitute the service level contract (SLA) program. The SLA plan includes details such as query parameters and the number of oracles required by the buyer. In addition, the buyer specifies the reputation and aggregation

contracts used in the rest of this contract.

Using the reputation recorded on the chain and more complete data collected from past contract records, buyers can manually sort, filter and select oracle services through the off-chain list service. Our intention is to allow BTBR to also have such a listing service to collect all BTBR-related records and verify the binary files of the listed oracle contracts. We introduce the listing service and reputation system in further detail in Section 5. The data used to generate the list will be extracted from the blockchain, allowing the construction of an alternative oracle list service. The buyer will submit an SLA proposal to the oracle and conclude the contract before finalizing the on-chain SLA.

In all cases manual matching is impossible. For example, the contract may need to request oracle services from time to time to dynamically respond to received information. Automation solves this problem and improves usability. For these reasons, BTBR proposes automated oracle matching by using order matching contracts.

Once the buyer specifies the SLA plan, instead of directly contacting the oracle, they will submit the SLA to the order matching contract. A log file (log) will be started by submitting a proposal to the order matching

contract. The oracle provider can monitor and filter this log file based on its capabilities and service goals. Then the BTBR node will choose whether to bid for the solution, and the contract only accepts bids from nodes that meet the SLA requirements. When the oracle service provider bids on the contract, they will be responsible for it, especially by means of additional penalties. According to the terms of the SLA, the penalties will be turned over if the provider conducts improper behavior.

You can bid during the entire bidding period. Once the SLA has received enough qualified bids and the bidding period has ended, select a certain number of oracles from the bid pool. During the bidding process, the penalty will be returned to the unselected oracles. A final SLA record will be created. When the SLA recording is completed, a log file will be triggered to notify the selected oracles. Then, the oracle will perform the tasks specified in the SLA.

Data report

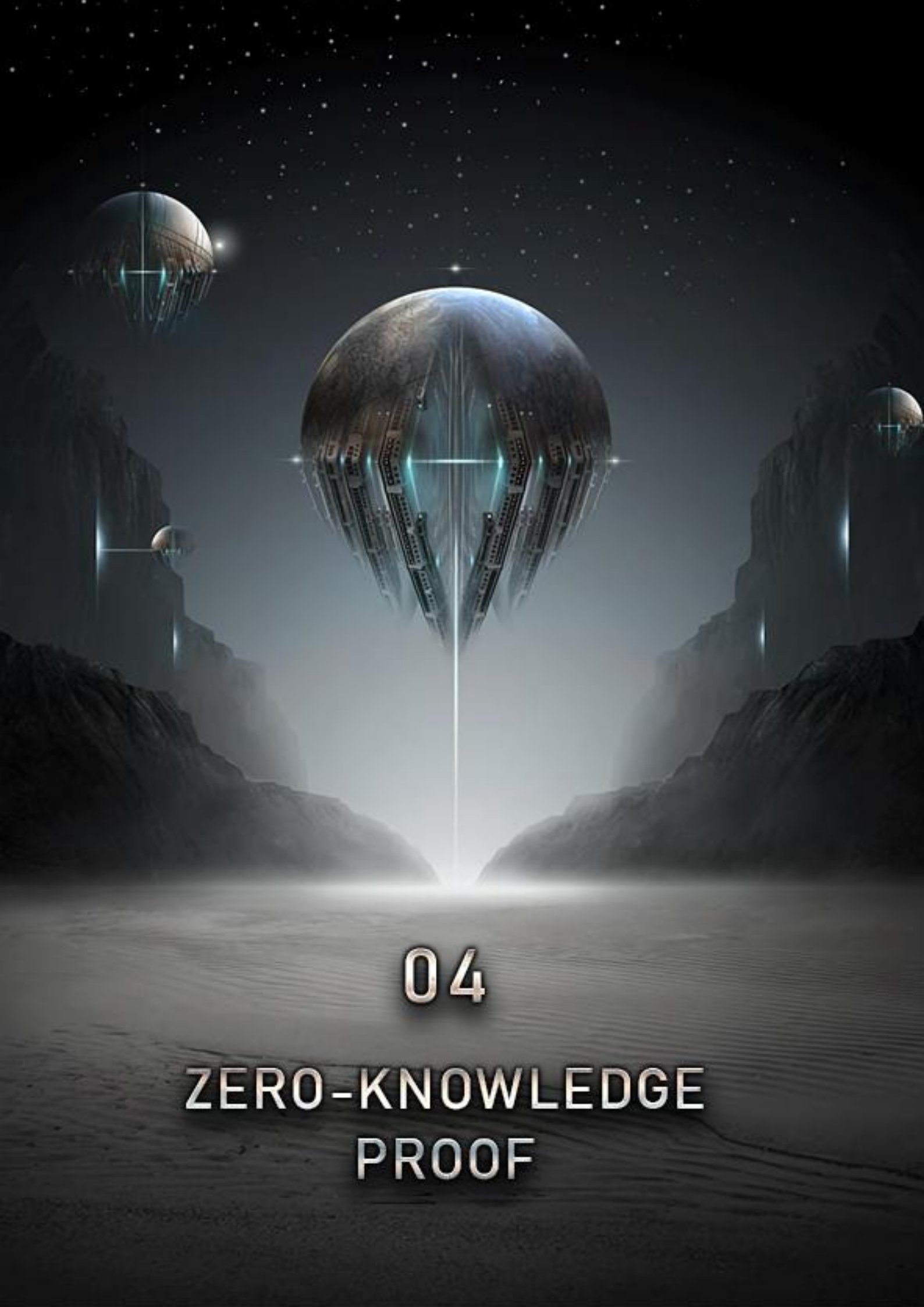
Once a new oracle record is created, the oracle under the chain will execute the contract and report to the link.

Summary of results

Once the oracles announce their results to the oracles contract, these

results will be sent to the aggregation contract. The aggregate contract collects the aggregate results and calculates a weighted value, then reports the validity of each oracle's response to the reputation contract. Finally, the weighted value is returned to the contract function specified in USER-SC.

Detecting outliers or erroneous values is a problem specific to each type of data feed and application. For example, detecting and rejecting abnormal answers before averaging is necessary for numerical data, but not for Boolean values. Therefore, there is no specific aggregation contract, there is only one settable contract address specified by the buyer. BTBR will include a set of standard aggregate contracts, but customized contracts can also be specified as long as they conform to the standard calculation interface.


04
ZERO-KNOWLEDGE
PROOF

Zero-knowledge proof

The prover can convince the verifier of a certain assertion without providing any useful information to the verifier. This technology is zero-knowledge proof, and asymmetric encryption is often used for identity authentication. As long as the verifier uses the public key to solve the random number provided by itself, it can prove the identity of the authenticated party without providing its own private key.

Homomorphic hiding

The function E(x) that satisfies the following three conditions is called additive homomorphism.

1. For most x, it is usually difficult to find x at a given E(x).

2. Different inputs will get different outputs - so if $x \neq y$, then $E(x) \neq E(y)$.

3. If someone knows E(x) and E(y), he can generate x and y in the arithmetic expression. For example, they can use E(x) and E(y). to calculate E(x+y).

In the same way, we can define multiplicative homomorphism or even full homomorphism. Both of our commonly used asymmetric encryption methods, RSA and ECC, support additive homomorphism, calculations and proofs require more formula operations, and there is time to open

another article to explain. Like RSA and ECC, note that the E(x) calculation here is performed in a finite field, which is called Fp below. After completing homomorphic hiding, we can achieve a certain degree of zero-knowledge proof. A has the two secret numbers of x and y and needs to prove to B that the sum of these two numbers is 7. It only needs to perform the following three steps:

1. A calculates E(x), E(y) and sends it to B
2. Because the function E(x) satisfies the additive homomorphism, B can calculate E(x+y) by E(x), E(y)
3. B calculates E(7) independently and verifies that E(x+y)=E(7)

Polynomial blind verification

Using the characteristic of additive homomorphism, we can simply extend the zero-knowledge proof to polynomials. Suppose A knows a polynomial P of the highest degree d, and B wants to know E(P(s)) corresponding to a certain s. We hope that during the verification process, A only knows P, does not know s, B only knows s, and does not know P. This can be achieved in the following way:

1. For each index of s, B calculates E(1), E(s),..., E(sd), and sends it to A.
2. A knows all the coefficients of the polynomial, can use the homomorphic property to calculate P(s), and send it back to B.

KCA and complete polynomial blind verification

We first define a concept: α pair refers to a pair of values (a, b) satisfying b= α*a. It should be noted that the multiplication here is actually the multiplication on the elliptic curve (ECC). The operation on the elliptic curve conforms to two characteristics: one is that when the value of α is large, it is difficult to infer α from a and b, and the other is addition and Multiplication satisfies the characteristics of commutative groups. That is to say, the commutative law of addition and multiplication is also valid on the elliptic curve. We use the characteristics of α pairs to construct a process called KCA (Knowledge of Coefficient Test and Assumption).

1. B randomly selects an α to generate α pair (a, b), saves α by itself, and sends (a, b) sends to A;

2. A selects γ to generate (a′, b′)=(γ· a,γ· b), and returns (a′, b′) to B. Using the commutative law, it can be proved that (a′,b′) is also an α pair, b′=γ· b=γα· a=α(γ· a)=α· a′;

3. B checks (a′,b′). If it is confirmed that it is α pair, it can be asserted that A knows γ.

This proof can be extended to multiple α-pair scenarios, called d-KCA.

1. B sends a series of α pairs to A;

2. A uses (a′,b′)=(c1· a1+c2· a2,c1· b1+c2· b2) to generate a new α pair;

3. B’ s verification is passed. It can be asserted that A knows the c array.

Multiplication homomorphism

The last step of the Pinocchio protocol, B needs to check E(L(s)*R(s)-O(s))=E(H(s)*T(s)). But in fact, we only mentioned Until E(x) satisfies the additive homomorphism, B cannot calculate E(H(s)*T(s)) through E(H(s)). The solution needs to return to our mathematical tools, we need to use the characteristics of elliptic curve pairing. This is a long story, and this article only gives a conclusion. Through elliptic curve pairing, we can get a weakened version of multiplicative homomorphism.

Define E1(x):=x· g, E2(x):=x· h, E(x):=x· g.

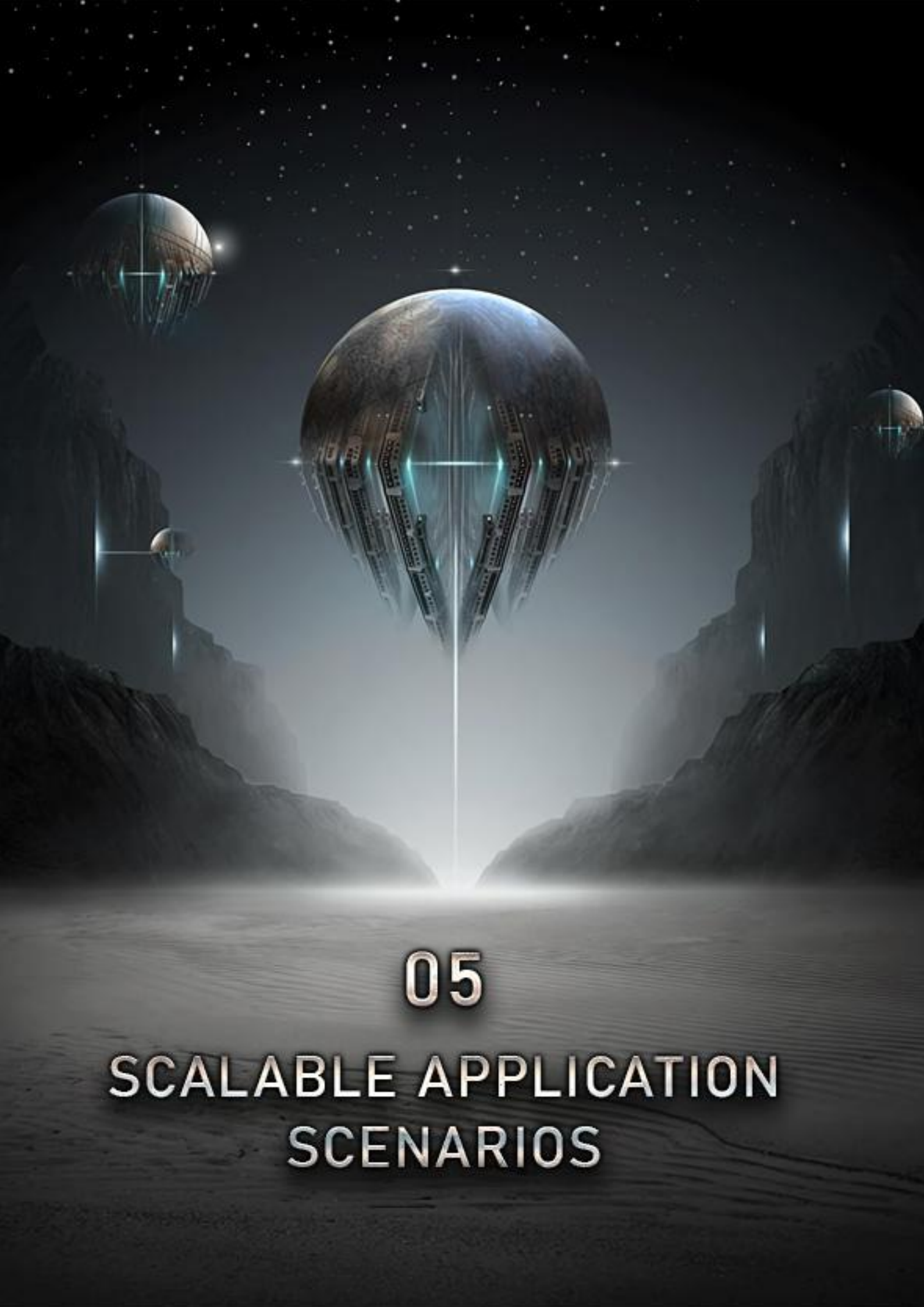
Because the three functions are all elliptic curves, naturally they all conform to additive homomorphism. At the same time, the elliptic curve pairing feature can ensure that we can calculate E(xy) through E1(x) and E2(y).

Reducing interaction

The last and most critical issue is that the Pinocchio protocol requires a lot of message interaction between A and B. In the blockchain, what we want to achieve is "public authentication." The ideal situation is that as long as A puts the evidence on the chain as a string, anyone can verify the

conclusion. Unfortunately, this kind of zero-interaction proof in the strict sense has been proved to be unable to satisfy all proof scenarios.

We took the second place and adopted a method called CRS (COMMON REFERENCE STRING).



05

SCALABLE APPLICATION SCENARIOS

Scalable application scenarios

Distributed exchange

Bitcoin uses massive replication to increase the security of distributed ledgers. In a similar way, we can run exchanges on the blockchain to reduce the possibility of internal and external attacks. We call it a decentralized exchange.

Today, the decentralized exchanges considered by the cryptocurrency community are based on "cross-chain atomic transactions" transactions (AXC transactions). Through AXC transactions, two users on two different chains can initiate two transfer transactions. The transaction is on two ledgers and either submitted for execution together, or neither is executed (that is, the atomicity of the transaction). For example, two users can use AXC transactions to achieve transactions between Bitcoin and Ethereum (or any two tokens on different ledgers), even if there is no mutual relationship between Bitcoin and Ethereum blockchains. connection. Both exchange users under the AXC trading mode do not need to trust each other, nor do they need to rely on transaction matching services. The disadvantage is that both parties to the transaction must be online at the same time to conduct the transaction.

Another type of decentralized exchange is a distributed exchange with

independent blockchains that performs massive replication. Users of this type of exchange can submit limit orders and shut down their computers, and transactions can be executed while users are offline. Blockchain will complete matching and transactions on behalf of traders.

A centralized exchange can construct an order book for limit-price transactions with a large trading volume to attract more traders. In the field of exchanges, liquidity will lead to more liquidity, so in the exchange business, its network effects are becoming more and more obvious (or at least the "winner takes all" effect). Currently, the cryptocurrency exchange Poloniex ranks first with a trading volume of USD 20 million per 24 hours, while Bitfinex ranks second with a trading volume of USD 5 million per 24 hours. Under this powerful network effect, the trading volume of AXC-based decentralized exchanges is unlikely to exceed that of centralized exchanges. If a decentralized exchange wants to compete with a centralized exchange, it needs to support the operation of an in-depth trading order account composed of limit orders. Only blockchain-based decentralized exchanges can achieve this.

The fast transaction execution provided by the ETPM entropy Matthew algorithm is another big advantage. BTBR's internal network can prioritize and quickly determine finality without sacrificing consistency to achieve rapid completion of transactions - at the same time for transaction order

transactions, as well as IBC (inter-blockchain communication) token transactions with other networks .

In summary, according to the existing cryptocurrency exchanges, a major application of BTBR is a decentralized exchange (called BTBR DEX). Its transaction throughput energy and commission delay are comparable to those of centralized exchanges. Traders can submit limit orders while all parties are offline. Moreover, based on the ETPM entropy Matthew algorithm, BTBR hub and IBC, traders can quickly complete the transfer of funds in exchanges and other networks.

As a link to other cryptocurrencies

The privileged partition can be used as a source of tokens linked to other cryptocurrencies. This link is similar to the relationship between the BTBR hub and the partition. Both must update each other's latest blockchain in time to verify that the tokens have been transferred from one party to the other. The "bridging partition" linked to the BTBR network must be synchronized with the center and other cryptocurrencies. This kind of indirect "bridging partition" can keep the hub logic concise, and it is unnecessary to understand other on-chain consensus strategies, such as the Bitcoin workload proof mining mechanism.

Send tokens to BTBR hub

Each validator linked to the bridge partition will run a special ABCI bridge application on the blockchain based on the ETPM entropy Matthew algorithm formula, but will also run a "full node" of the original blockchain .

When the original blockchain digs a new block, the bridge partition verifier will sign and share the starting point blockchain prompts, and their respective partial perspectives can reach agreement. When a bridge partition receives payment from the original blockchain (such as a sufficient number of confirmations on the chain of PoW mechanisms such

as Ethereum or Bitcoin), a balance with the corresponding account is created on the bridge partition.

For Ethereum, the bridge partition can share the same validator with the BTBR hub. In terms of Ethereum (the original blockchain), a bridge agreement will allow ether owners to pass the bridge agreement that sends Ether to the bridge partition of Ethereum. Once the Ether is received in the bridge connection, the Ether cannot be withdrawn unless the corresponding IBC data packet is received from the bridge partition. The bridging contract follows the verification group of the bridging partition, which may be the same as the validator group of the BTBR hub.

As far as Bitcoin is concerned, the concept is similar, except that instead of a bridging contract, each UTXO will be restricted by a threshold multi-signature P2SH database. Due to the limitations of the P2SH system, the signer cannot be the same as the validator group of the BTBR hub.

Withdrawing tokens from the BTBR hub.

The Ether on the bridge partition ("Bridge Ether") can be transferred in and out between hubs. After the transfer is completed to a specific Ethereum withdrawal address, the transferred "Bridge Ether" is completely deleted. An IBC message can prove the transaction on the

bridge partition. This message will be announced to the Ethereum bridge agreement so that the Ether can be withdrawn.

As far as Bitcoin is concerned, the rigorous transaction script system makes it difficult to implement the mirror conversion mechanism of IBC coins. Each UTXO has its own specific script, so when the Bitcoin fulfillment signer changes, each UTXO must migrate to a new UTXO. One solution is to compress and decompress UTXO-set as needed to keep the total number of UTXO down.

Full responsibility system of linked area

The risk of this type of linked contract is that malicious validator groups may appear. If the Byzantine voting power exceeds ⅓, it will cause a fork. That is, while withdrawing Ether from the Ethereum bridge contract, it can also keep the pegged Ether in the bridge partition unchanged. Moreover, if the voting power of Byzantium exceeds ⅔, someone may directly use the bridge logic that separates from the original bridge partition to start sending Ether to the account in the bridge agreement and steal Ether.

If this bridging method is completely designed as a responsibility system, it is possible to solve this problem. For example, all IBC packages of the hub and the starting point may need to be approved by the bridge zone first, that is, the bridge joint agreement in the hub or the starting point

can effectively verify all state transitions of the bridge zone. The hub and the starting point should allow the verifiers in the bridging zone to provide collateral, and the transfer of tokens in the overseas community contract needs to be delayed (and the collateral untie time should be long enough), so that a single auditor has time to initiate any Question. We will open the design description and implementation method of this system as a proposal for future BTBR improvement, pending the approval of the management system of the BTBR hub.

Ethereum expansion

As we all know, the scaling problem is a problem that has plagued Ethereum. At present, the Ethereum node will process every transaction on the node and store all the state.

The ETPM entropy Matthew algorithm submits blocks faster than the Ethereum proof of work, so the Ethereum virtual machine partition promoted by the ETPM entropy Matthew algorithm consensus and running with bridged ether can enhance the performance of the Ethereum blockchain. In addition, although the BTBR hub and the IBC package mechanism cannot implement the execution of the contract logic per second, it can be used to coordinate the circulation of tokens between the Ethereum contracts in different partitions, and lay the

foundation for expansion for the token-centric Ethereum through the sharding method.

Multi-purpose integration

The BTBR partition can run any application logic. The application is set when the partition is created and can be continuously updated by the administrator. This flexibility allows the BTBR partition to be used as a peg to other cryptocurrencies, such as Ethereum or Bitcoin. And it can also be linked to these blockchain derivatives, using the same code base, and distinguishing between verification procedures and initial distribution. This allows multiple existing cryptocurrency frameworks to run, such as Ethereum, Zerocash, Bitcoin, CryptoNote, etc. Combining it with the ETPM entropy Matthew algorithm Core, it becomes a consensus engine with better performance in the general network, providing more interaction opportunities between platforms. In addition, as a multi-asset blockchain, each transaction may contain multiple input and output items. Each input item can be any token, making BTBR directly a decentralized exchange. Of course, it is assumed that trading orders are matched through other platforms. The alternative is to make the partition as a distributed fault-tolerant exchange (including trading accounts), which is a strict improvement on the centralized cryptocurrency exchange-existing exchanges have often been attacked in the past.

Partitions can also be used as a blockchain version of enterprise and government systems. The specific services originally run by one or more organizations are now run as ABCI applications on a certain partition, so as not to give up control of the underlying services, and to maintain the security and interactivity of the public BTBR network. Therefore, BTBR may provide an excellent operating environment for those who want to use blockchain technology but are unwilling to completely give up control to distributed third parties.

Mitigating network partition problems

Some people think that there is a major problem with consensus algorithms that support consistency, such as the ETPM Entropy Matthew Algorithm, that is, the network partition will cause no partition to have more than ⅔ voting rights (for example, more than ⅓ voting rights offline), which will be interrupted consensus. The BTBR architecture can alleviate this problem. It can use the global center. At the same time, each district implements regional autonomy, and then the voting rights of each district are distributed according to the normal geographical location. For example, the general example may be for individual cities or regions, allowing them to run their own partitions while sharing a common hub (such as the BTBR hub), and can continue during the outage caused by the temporary network partition Maintain regional autonomy activities.

Note that in this way, in the process of designing a robust federated fault-tolerant system, you can think about the characteristics of real geography, politics, and network topology.

Federated name resolution system

NameCoin is one of the first blockchains to try to solve the name resolution problem through Bitcoin technology.

However, this scheme has some shortcomings.

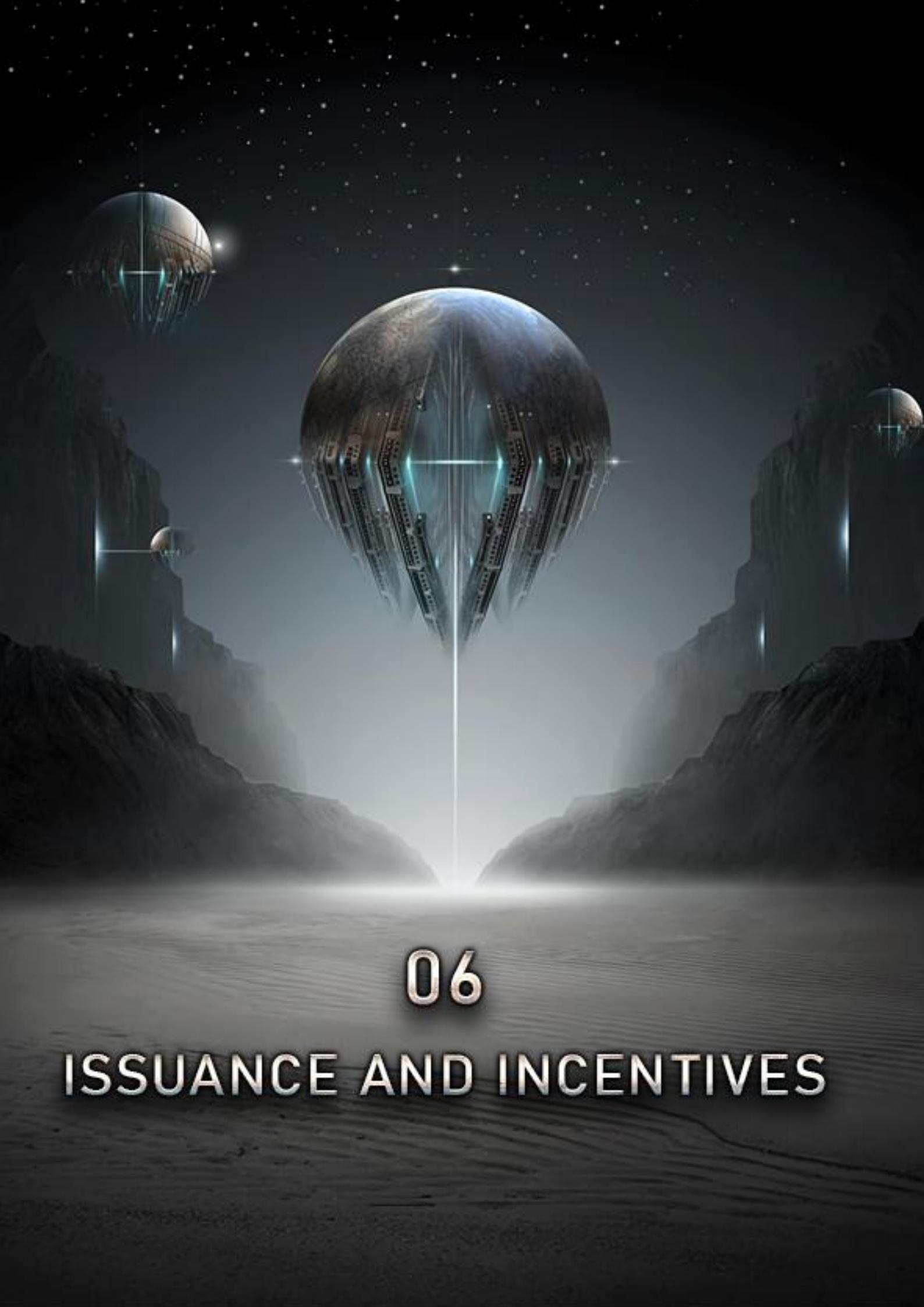
For example, we can use Namecoin to verify that @satoshi was registered with a specific public key at some point in the past. However, we don't know whether the public key has been updated, unless all the names before the last update are downloaded. It is caused by the limitations of the Merkelization model in the Bitcoin UTXO transaction model. In this type of model, only transactions (not variable application states) will be Merkelized into the block hash. It allows us to later use the update to prove the existence of the name, not its non-existence. Therefore, we must rely on full nodes to clarify the nearest value of this name, or spend a lot of resources to download the entire blockchain.

Even if a Merkelized search tree is used on NameCoin, the independence of its proof of work will still cause problems in the verification of light clients. The light client must download a complete backup of all block

headers in the blockchain (or at least all block headers updated since its last name). This means that the bandwidth needs to be expanded linearly over time. In addition, in the proof-of-work system, the name change on the blockchain needs to wait for additional proof-of-work verification and confirmation, which may take an hour on Bitcoin.

With the ETPM entropy Matthew algorithm, we only need the block hash signed by the quorum validator (through voting rights), and the Merkel proof of the current value associated with the name. It makes simple, fast, and secure name value verification of light client possible.

In BTBR, we can use this concept and extend it. Each name registration on BTBR can have a related highest-level domain name (TLD), such as ".com" or ".org", etc., and each name registration zone has its own management and registration rules.



06

ISSUANCE AND INCENTIVES

Issuance and incentives

BTBR token

The total circulation of BTBR is 210 million;

The number of genesis blocks: 102010;

Block time: 10 minutes

Block reward: 500 pieces.

Top mine 101 node

21 head mine nodes will be opened for the first time, and one node will be added for each subsequent increase of 100 reservations. The top mining nodes receive the creation block reward of the whole network, and each top mining node can get 1010 BTBR.

Limit on the number of validators

Unlike Bitcoin or other proof-of-work blockchains, as the complexity of communication increases, the ETPM entropy Matthew algorithm blockchain will slow down as the number of validators increases. Fortunately, we can support enough validators to implement a reliable global distributed blockchain and have a very fast transaction confirmation time. And with the increase of bandwidth, storage and

parallel computing capacity, we will be able to support more validators in the future.

On Creation Day, the maximum number of validators will be set to 100, and this number will increase at a rate of 13% for 10 years, and eventually reach 300.

Become a validator after Creation Day

BTBR holders can become validators by signing and submitting BondTx transactions. The amount of pledged BTBR cannot be zero. Anyone becomes a validator at any time, unless the current number of validator groups exceeds the maximum. In this case, the transaction is valid only when the number of BTBR held is greater than the minimum number of valid BTBR held by the existing validators. The effective BTBR includes the entrusted BTBR. When a new validator replaces an existing validator in this way, the existing validator will be offline. All its BTBR and entrusted BTBR enter the unbound state.

Penalties for validators

For any verifier who intentionally or unintentionally deviates from the recognition agreement, certain penalties must be imposed. Some evidence is immediately admissible, such as dual signatures at the same height and round, or violation of "pre-voting lock" (ETPM Entropy

Matthew Algorithm consensus protocol rules). Such evidence will cause the validator to lose its good reputation, and its bound BTBR and its proportional share in the reserve pool-collectively referred to as "equity"-will be greatly reduced.

Sometimes, the validator will be unavailable due to local network interruption, power failure or other reasons. If in the ValidatorTimeoutWindow block at any point in the past, the validator's submission vote is not included in the blockchain for more than ValidatorTimeoutMaxAbsent times, the validator will be offline and the ValidatorTimeoutPenalty (default 1%) equity will be reduced.

Some "malicious" actions did not produce obvious evidence on the blockchain. In these cases, if there is a majority consensus, the validators can coordinate out-of-band to force these malicious validators to time out. If the BTBR hub is suspended because more than ⅓ voting rights are offline, or if more than ⅓ voting rights are reviewed to enter the blockchain, the hub must be restored with the help of a hard fork reorganization protocol. (See "Fork and Censorship Attacks" for details)

Transaction fee

BTBR hub validators can accept any kind of tokens or combinations as fees for processing transactions. Each validator can set the exchange rate

and choose the transaction they want. As long as it does not exceed BlockGasLimit, every ValidatorPayoutPeriod (default is 1 hour) will be allocated according to the ratio of BTBR bound by stakeholders.

Among the transaction fees charged, ReserveTax (default 2%) will be deposited in the reserve pool to increase the amount of reserves and increase the security and value of the BTBR hub. These funds can also be allocated in accordance with the decisions of the governance system.

BTBR holders who delegate voting rights to other validators will pay a certain commission to the delegating party, and this fee can be set by each validator.

Incentives for hackers

The security of the BTBR hub depends on the security of the underlying validator and the delegation choice of the principal. In order to encourage discovery and early reporting of discovered vulnerabilities, the BTBR hub encourages hackers to publish successful vulnerabilities through the ReportHackTx transaction, saying, "This validator has been compromised, please send the rewards to this address". In this case, validators and delegators will be suspended and the BTBR of each person's HackPunishmentRatio (default 5%) will be reduced. The BTBR of HackRewardRatio (default 5%) will be sent to the hacker's reward address

as a reward. The verifier must use its backup key to restore the remaining BTBR.

In order to prevent this feature from being abused to transfer unauthorized BTBR, the ratio of BTBR (authorized and unauthorized) will remain unchanged before and after ReportHackTx. The hacker's bounty will include some unauthorized BTBR (if any).

Governance norms

The BTBR hub is managed by a distributed organization and requires a clear governance mechanism to coordinate various changes to the blockchain, such as system parameter variables, software upgrades and constitutional revisions.

All validators are responsible for voting on all proposals. Failure to vote on the proposal in time will result in the validator being automatically deactivated for a period of time. This period is called AbsenteeismPenaltyPeriod (default 1 week).

The delegator automatically inherits the voting rights of the verifier it delegates. This vote can be manually overwritten. The unbound BTBR has no voting rights.

Each proposal requires a deposit of MinimumProposalDeposit tokens, which may be a combination of one or more tokens (including BTBR). For each proposal, voters can vote to remove the deposit. If more than half of the voters choose to withdraw the deposit (for example, because the proposal is spam), then the deposit will be deposited in the reserve pool, except for the burned BTBRs.

For each proposal, voters can choose the following options:

- Agree
- Strongly agree
- Against
- Strongly opposed
- Abstention

Deciding to adopt (or not adopt) a proposal requires a strict majority to vote "agree" or "strongly agree" (or "disagree" and "strongly disagree"). However, if more than 1/3 of the people vote "strongly disagree" or "strongly support", then the majority of the decision can be overruled. If the majority of votes are rejected, then each of them will lose VetoPenaltyFeeBlocks (the default is one day's block value, excluding

taxes) as a penalty. The party that rejects most of the decisions will also lose an additional BTBR of VetoPenaltyBTBRs (default 0.1%) as a penalty.

Horizontal expansion

Interledger agreement

The Interledger Protocol (The Interledger Protocol, ILP) is not a strict extension scheme. It provides a specified interactive operation across different ledger systems through a loosely coupled bilateral relationship network. Like the Lightning Network, the purpose of ILP is to achieve payment, but it pays special attention to cross-ledger payments and expands the processing mechanism of atomic transactions so that transaction processing not only supports hash locks, but also includes a quorum of notaries (called For the atomic transport agreement). The latter's mechanism for implementing atomicity in transactions between ledgers is similar to the light client SPV mechanism of the ETPM entropy Matthew algorithm, so it is necessary to compare the difference between ILP and BTBR/IBC, see below for details.

1. ILP does not support the change of connector notaries, nor does it allow flexible weights between notaries. On the other hand, IBC is specifically designed for the blockchain, validators can have different

weights, and with the development of the blockchain, members can change at any time.

2. Like the Lightning Network, the recipient in the ILP must be online to send a confirmation to the initiator. In the transmission of IBC tokens, the set of validators on the blockchain where the recipient is located is responsible for providing confirmation instead of receiving the user himself.

3. The biggest difference is that the ILP connector does not need to be responsible for maintaining authority over the payment status. However, in BTBR, the RP verifier is responsible for the authority of the IBC token transmission status and the number of tokens held in each area. Allowing the safe asymmetric exchange of tokens between areas is an essential innovation. The ILP connector in BTBR can be regarded as a durable and most secure blockchain ledger: BTBR RP.

4. Cross-book payment within ILP requires the support of an exchange's instruction set. Because there is no asymmetric transfer of tokens from one ledger to another, only market equivalents can be transferred.

Side chain

Sidechains is a mechanism that expands the performance of the Bitcoin network by replacing the blockchain with a "two-way peg" to the Bitcoin

blockchain. (Two-way pegs are equivalent to bridging, which is called "bridging" in BTBR to distinguish it from market pegs). The side chain allows Bitcoin to easily move between the Bitcoin blockchain and the side chain, and allows experimentation of new features on the side chain. In BTBR RP, the side chain and Bitcoin are each other's light clients, and SPV proof is used when moving between the Bitcoin blockchain and the side chain. Of course, because Bitcoin uses PoW, Bitcoin-centric side chains suffer from many problems and risks caused by PoW as a consensus mechanism. Moreover, this is a solution to maximize the benefits of Bitcoin, and does not support various tokens and inter-area network topologies like BTBR. However, the core mechanism of two-way pegging is in principle the same as that used by BTBR.

Ethereum's scalability efforts

Ethereum is currently studying many different strategies to partition the state of the Ethereum blockchain to address scalability requirements. The goal of these efforts is to maintain the current Ethereum virtual machine to provide an abstraction layer on top of the shared state space. Currently, a number of research work is underway.

BTBR vs Ethereum 2.0 Mauve

BTBR and Ethereum 2.0 Mauve have different design concepts.

- BTBR is for tokens. Mauve is about expanding computing power.
- BTBR is not limited to EVM, so even different VMs can interact.
- BTBR lets the creator of the area decide the validator.
- Anyone can create a new area in BTBR (unless the manager decides otherwise).
- The failed isolation between RP and area, so the global token invariant can be maintained.

Universal expansion

Lightning Network

The Lightning Network is designed as a token transmission network that runs on top of the Bitcoin blockchain (and other public blockchains) by transferring most of the transactions from outside the consensus ledger to the so-called "payment channel". This is achieved through encrypted currency scripts on the chain. These scripts enable both parties to enter the stateful contract held by both parties, update the state by sharing digital signatures, and finally publish evidence on the blockchain after the contract ends. This mechanism was first welcomed by cross-chain atomic swap transactions. By opening payment channels with multiple parties, participants in the Lightning Network can become a centralized point to provide routing for other people's payments, resulting in complete

connectivity of the payment channel network, at the cost of its funds tied to the payment channel.

Although the Lightning Network can easily span multiple independent blockchains and realize value transfer with the help of the trading market, it cannot realize asymmetric token transactions from one blockchain to another. The main advantage of the BTBR network described here is to enable direct exchange of tokens. In other words, we hope that payment channels and the Lightning Network will be widely adopted along with our token transfer mechanism to save costs and protect privacy.

Segregated verifier

Segregated Witness is a Bitcoin improvement proposal BIP, which aims to increase the throughput of each block by 2 or 3 times while enabling new nodes to synchronize blocks faster. The highlight of this solution is how it allows soft fork upgrades under the limitations of Bitcoin's current protocol (for example, clients with older versions of software will continue to run after the upgrade). As a new protocol, the ETPM entropy Matthew algorithm has no design restrictions, so it has different expansion priorities. ETPM Entropy Matthew Algorithm BFT's round-robin algorithm is mainly based on cryptographic signatures rather than mining. This algorithm allows horizontal expansion through multiple parallel

blockchains, while more conventional and frequent block submissions also allow vertical expansion.

Route planning

Planning route:

2004

Manuscript written by Bitcoin priest, discussed by punk forum enthusiasts.

2008

Satoshi Nakamoto used the content of the manuscript to create Btc.

2020

Release Bit Brothers, open GitRP, and release the original entropy Matthew algorithm to balance the universe. Started pre-sale of cloud computing power in the seed stage.

2021 Q1

Gradually start the ecologicalization of mining pools, and open up higher computing power to establish the development of the universe bit space node cloud computing platform.

2021 Q3

Establish the world's largest ecological cloud computing power and ecological commercial operation of mining pools.

2022 Q1

Diversified cross-chain and multiple computing power implanted in encrypted assets form financial ecological development.

2022 Q3

Create a compatible financial infrastructure and continue the decentralized autonomous system.