SMARTX Technical White Paper

Version 3 210311

Table of Contents

1 Project introduction

1.1 About SmartX

SmartX is a blockchain technology public chain developed by the Smart community team. It is a decentralized BlockDAG project that supports Pow mining and a "transaction-based DAG" structure. It can achieve unlimited performance expansion. Compared with the traditional blockchain system that expands in parallel, it can expand infinitely in parallel.

Based on the Smatx platform, blockchain technology can be applied for decentralized financial Defi, decentralized exchanges, decentralized social networks, private communications, game payment, and other products.

SmartX aims to connect global service providers and users through the blockchain and use decentralized financial Defi and social entertainment as the entry point to build a trusted and secure blockchain ecosystem.

SmartX platform is to build a multi-platform for capital flow, information flow, and value flow. In the trust value system constructed by SmartX, people will transmit their self-value through the blockchain network to form a value interconnection ecology, ultimately improving social production efficiency.

1.2 Project overview

With the development of the blockchain system, transaction speed has become a bottleneck for further developing blockchain industrialization, such as DAPP, Internet of Things, and trusted data exchange. BTC/BCH achieves the expansion goal purely by increasing the block size; this method is not advisable. Besides, other methods like using BTC/ETH sub-networks such as Lightning Network, cross-chain, and other technologies to improve TPS are, in fact, at the expense of security. Besides, they must satisfy more attached conditions (such as both parties must be online simultaneously, provide mortgage, etc.) to finish the transaction. As such, the BlockDAG solution appears. From the earliest IOTA to the recent Conflux project, SmartX also applies this solution.

SmartX creates an innovative and brand-new DAG structural algorithm-fragmentation lazy fusion algorithm (B-DAG algorithm), which can integrate the transaction shards

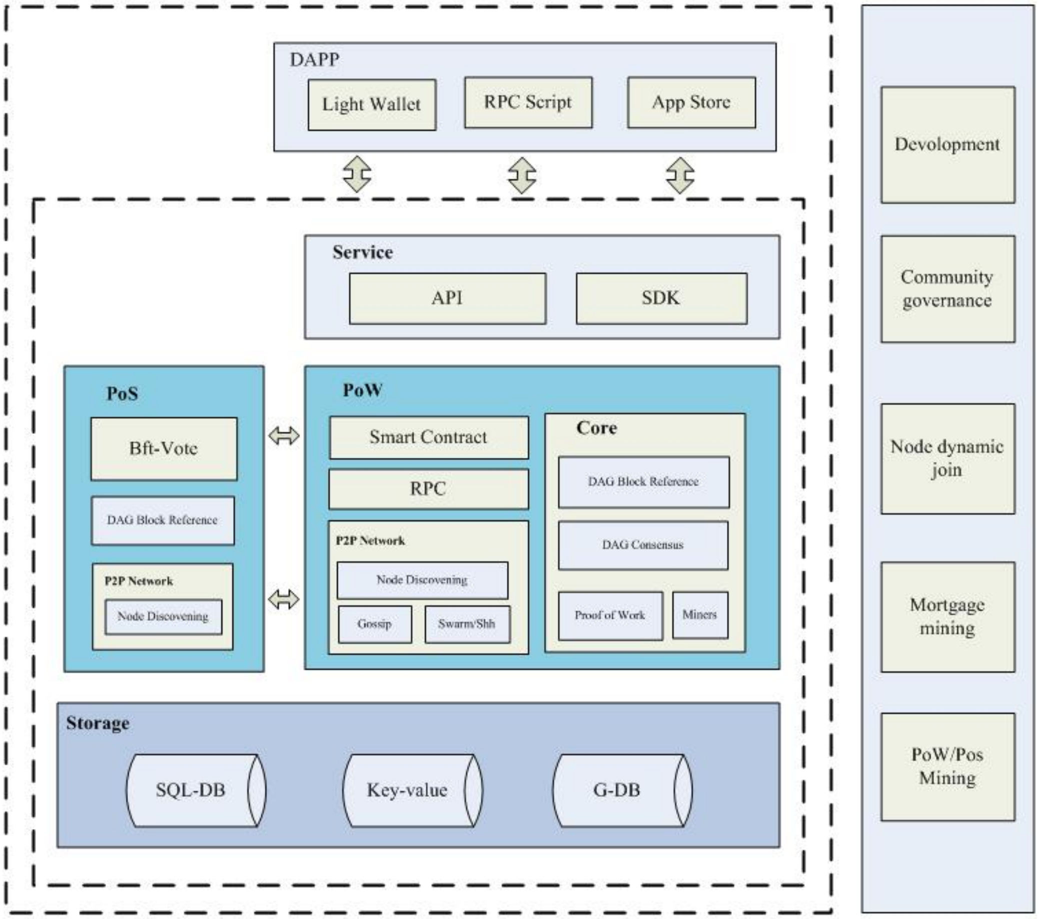
generated by nodes around the world. Currently, Mainstream BTC/BCH, Ethereum, and other PoW tokens blocks must be pending transactions simultaneously and then packaged one by one. Different nodes package transaction blocks mutually exclusive (although Ethereum has GHOST and Uncle Block mechanisms, it is far from enough).
It is equivalent to the highway with only one exit; no matter how many roads or vehicles there are, there will be only one exit. The exit speed determines the flow of traffic. SmartX is equivalent to opening up unlimited highway exits to allow complete expansion of vehicle traffic.
A stable principal unit MC is generated through the Epoch cycle, which is linearly linked according to the B-DAG algorithm, forming a Bitcoin chain structure. However, the full view is a DAG structure, which can be divided into independent Partitions without interfering with each other to generate transactions and then performing transaction block integration, which can expand the performance infinitely.
In this structure, double-spending detection can succeed, as the algorithm is stable and consistent in the global perspective. Once a node can see the global data of a particular Epoch with a stable and consistent sorting algorithm (B-DAG algorithm), the double-spending detection will succeed.

1.3 System structure

A centralized system is like a centralized church, where there must be a single point of failure (SPOF). If a single point of failure appears, the system will collapse. Blockchain systems are like bazaars, where there are multiple centers, many people, and unified market rules. Complete decentralization means that there is no Owner nor centralized stakeholders. Advantages are that single point of evil, and single point of collapse caused by centralization could be avoided. The disadvantage is that it pays tremendous energy consumption and waste of resources.

If a transaction is made on a centralized cloud server such as WeChat payment, it consumes only 1k bytes of disk and network bandwidth resources in less than a second, and the energy consumption is negligible. However, in the ETH system, based on 10,000 nodes, its hardware resource consumption is correspondingly 10,000 times, and its energy consumption maybe tens of millions of times.

In SmartX, we use some of the advantages of churches and bazaars. The system architecture operates in a federated model. The main network consists of multiple PoW master nodes that pledge tokens and multiple PoS verdict voting nodes. PoS nodes are elected to join by reputable organizations in the community. The PoW master node can join and exit freely by staking a certain amount of tokens. The PoW master node is equivalent to a mining pool node and can accept various computing power submissions, transactions, and ledger downloads from mining machines. The periphery is an ordinary node. Users can join ordinary nodes or use light wallets, RPC scripts, SDK APIs, or applications on the APP Store to access the main network and use SmartX services.



1.4 Transaction verification and confirmation

In the beginning, a transaction TX will be sent to the memory pool of the nearest master node NodeA. In a certain T cycle, NodeA will generate a master unit M to actively refer to the transaction TX and combine the master unit M and the transaction unit. A large Mmuster composed of TX is broadcast to all master nodes. Other master nodes will also do similar operations at the same time. In the end, only one M main unit with the largest proof of work volume becomes the winning main block MC. Unlike BTC/ETH, other low-work proof main units M will not fail. They will be directly or indirectly referenced by the winning main unit MC, forming a B-DAG tree structure.

When the MC block of the T cycle is confirmed, all M blocks and TX blocks that are directly or indirectly referenced by the MC block are confirmed. The process of selecting MC is a typical PoW process.

The PoS consensus is a voter network voted by block producers. When the most difficult M unit becomes the winning MC, it must be confirmed by the PoS voter network. The tree graph structure of B-DAG can run independently because of the PoW consensus on the chain. SmartX runs under the PoW consensus, which is no different from DAG projects such as IOTA.

In order to accelerate the convergence of the B-DAG structure (such as bifurcation integration under multiple networks), SmartX introduces a PoW/PoS double-layer consensus, and the PoS layer is equivalent to a network accelerator. Accelerated confirmation of PoW consensus.

The introduction of the PoS voting layer can converge the B-DAG dynamic structure and vote for another consensus on the chain, such as balance snapshots, data trimming, irreversible nodes, and time consensus, node QoS services, etc. In order to reward PoS nodes for their services to the main network, each time a reward block is produced, PoS will also be assigned a fixed reward.

The advantage of adopting the PoW/PoS joint consensus is that it allows the DAG network to deal with actual business problems without losing security and high TPS under the premise of ensuring decentralization.

1.5 Block diagram technology and stretchable dynamic block

Since the SmartX master node sends out blocks simultaneously, each master node (MC) will actively quote all transactions generated by the node. In theory, each master node can independently generate a block (MC-Muster). This block is all MC. The collection of

TX MC-Muster is limited by computer memory and network bandwidth. In theory, MC-Muster is exceptionally scalable. As small as a single transaction, as large as tens of thousands of transactions can be packaged into an MC-Muster (that is, the link block in the SmartX system). Therefore, the B-DAG technical solution can also be used as one of the dynamic expansion solutions of the traditional BTC chain structure blockchain.

High TPS has become the biggest bottleneck in the development of blockchain. Similar to Fomo3D games, or red envelope games on the high-frequency chain, high-frequency TPS is required. SmartX tries to solve such problems through B-DAG technology.

2 SmartX's DAG structure

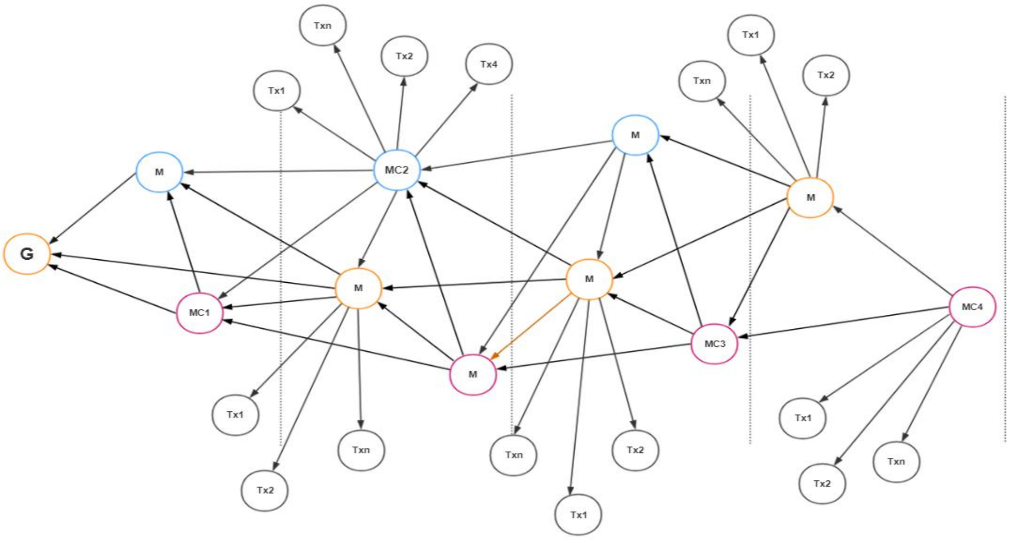
2.1 DAG structure

Unlike the traditional BTC chain blockchain, SmartX block is transaction TX, and transaction TX is blockchain.(Smartx block is transaction TX, and transaction TX is a block)In addition to the TX transaction unit, the SmartX system also has an MC main unit. The MC unit randomly generates transactions that have not been quoted according to a certain Epoch period. MC and TX form a DAG structure. From the perspective of each Epoch, with time, if only the SAT main chain reference (described later) is retained, and the other MC reference lines are removed, it is close to the BTC transaction chain structure. This structure combines some of the advantages of BlockChain and DAG technology. This is called B-DAG technology.

The B-DAG technical model does not require a large number of blocks to be suspended on the entire network; that is, the transaction volume cannot be controlled, and the transaction volume reaches a substantial-high point and triggers an avalanche. The solution to the above problems is to hand over the valve that controls the transaction volume to the primary controller for processing. It is mainly based on its performance indicators to dynamically set the valve and achieve the goal of dynamic expansion.

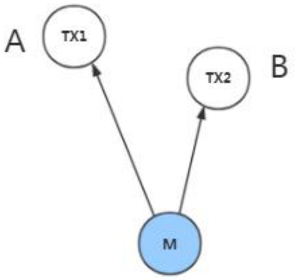
Master node: the mining router that generates and packs blocks on the network. After this, it is similar to the BTC/ETH mining pool router, but a certain amount of SAT coins must be mortgaged to join the network.

A typical B-DAG structure is as follows:



2.2 Simple blockchain structure

The main block M quotes two transactions TX1 and TX2, signed and confirmed by PoS decoders A, B, C, and D.



If we use the Json format to describe the structure of a simple master block reference and signed by the PoS node:

```
{
"hash": "THIS HASH",
"time": "2019-08-06 19:10:18",
    "type": "1",
"diff": "THIS diff",
"owner": "OWNER ADR OR PUBKEY",
"nonce": "5CC911F3B8434911B691B6CF7A361333",
        "flds": [{
        "type": "SAT_FIELD_OUT",
        "hash": "A",
        "time": "2019-08-06 21:21:19"
    }, {
        "type": "SAT_FIELD_OUT",
        "hash": "B",
        "time": "2019-08-06 21:21:19"
    }],
    "Signers": ["A", "B", "C", "D"],
    "Signinfo": "SIGN CONTENT",
    "sign":"THIS ECDSA SIGN"
}
```

2.3Account model

Like traditional trading systems, SmartX uses a balance account model. Every Epoch consensus cycle processing finds that a TXBlock OUTPUT account does not exist, then the entire network creates this account. Moreover, transfer the Amount of INPUT to this account, and INPUT must exist, and the balance is sufficient, Like traditional trading systems, smartx uses the balance account model. If the output account of a txblock does not exist in each epoch consensus cycle, the whole network will create this account. And transfer the amount of input into this account, and the input must exist and the balance is sufficient.

The balance of each account is determined by the difference between the trading unit INPUT and OUTPUT. In an Epoch, from the creation block to this moment, the account balance Balance = All(INPUT) – All(OUTPUT). The INPUT party signs each transaction unit with the ECDSA private key and the INPUT public key is used to verify the legality of the block.

The system guarantees that the state machine processes each Transaction (hereinafter referred to as TX) as idempotent; that is, the final result of the Transaction TX executed once and executed multiple times is the same. When the balance is modified, it needs to be locked concurrently, and every process and function of the system needs to be reentrant. Therefore, the account balance modification is the same as the state machine; no matter how many transactions flow in or how many repeated transactions, the final result is the same.

Judging whether it is a repeated transaction is based on the Nonce value of the random number of the Transaction TX. Except for Nonce, the hash value of the entire packet is called a reentrant success. Otherwise, it is called reentrant failure. The system handles these two situations separately.

At the same time, in the account table, each account will have a Nonce field, which is used to detect each transfer-out Transaction's repeated transactions.

However, if repeated transactions are in different pools and are sorted stably, the block with the higher priority is executed first, and the block with the lower priority does not store the actual block but only the hash.

When updating blocks later, the blocks with high prioritywill be updated first, while the transaction with only hash but no actual blocks should be checked to see whether there isduplicate transactions before.

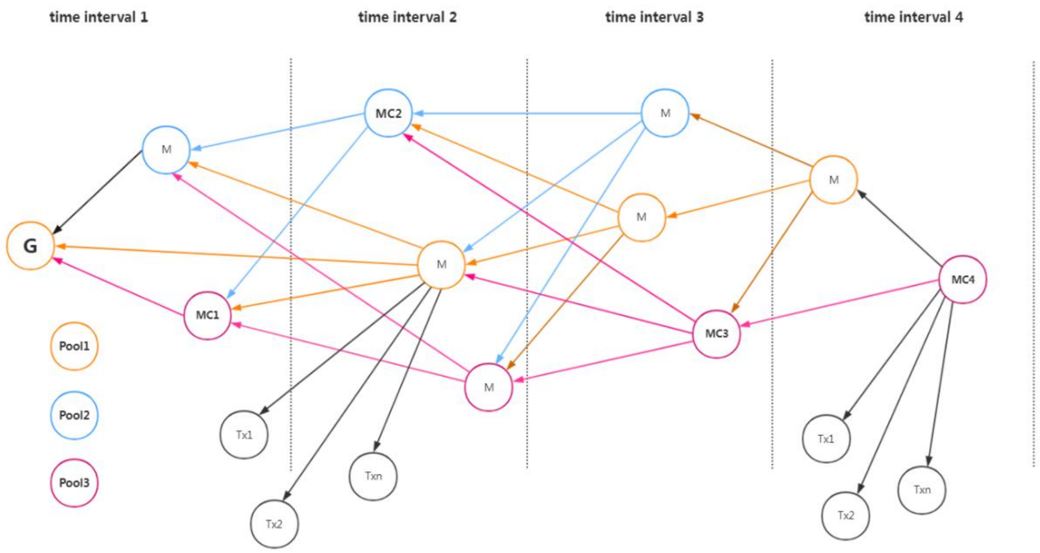
2.4 Reference relationship

The master node of the entire network concurrently generates the master unit MC according to a certain Epoch, and all transaction units TX generated under the Epoch. All MC and TX refer to each other according to the agreed rules. The general rules are as follows:

- Conventional rule 1: The main unit generated by the main node itself can refer to other broadcasted main units, but the main broadcast unit cannot refer to the main target unit, and can only be forwarded
- Conventional rule 2: The main unit can refer to the transaction unit
- Conventional rule 3: No circular reference between main units.

Own master block: A master block generated by a certain Epoch, and the master block actively searches for all unquoted broadcast master blocks and transaction blocks in the memory pool.

Mainbroadcastblock: After the main target unit generates and completes the reference operation, the main unit becomes the master broadcast unit after broadcasting to the connected node.



As shown in the figure, SmartX uses the forward reference rule to distinguish whether to quote the main block broadcasted by other mining pools through the cycle:

•If the main broadcast block and the current main block are in the same period or after the current main block period, do not quote immediately, but wait until the next period to quote the main broadcast block

•If the main broadcast block is before the current main node, the main block should be quoted if it meets quotation rules

• All M blocks (including the winning MC block) must refer to the last Epoch cycle ruling MC block

• The main network will discard the main block that was not directly or indirectly referenced by the MC block in the previous Epoch cycle

3 SmartX consensus

3.1 PoW-PoS double-layer consensus

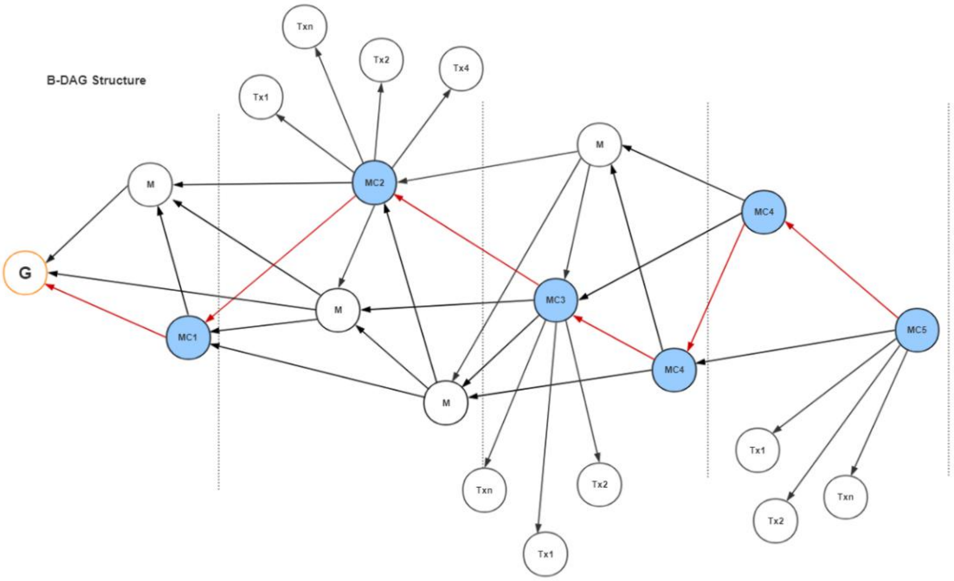
In SmartX, the double-layer consensus of PoW and PoS is adopted. The consensus on the PoW chain is that each time slice decides to win an MC block, that is, the main block with the most challenging proof of work is the winning main block. The PoS layer is a consensus network that votes by block producers. The most difficult MC block is also the winning main block voted by PoS voters. The PoS layer is similar to an accelerator that can accelerate transaction confirmation.

In addition, the PoSlayer can also handle various types of network-wide consensus voting such as irreversible nodes, snapshots of the entire network balance, PoS services, penalty transactions, and time proofreading. When the isolated network splits, the main network has PoW/PoS double protection.

3.2 Maximum weight chain

In SmartX, every Epoch cycle will generate a stack of MC blocks, and the MC blocks refer to each other. Since the MC block with the largest weight can be selected from many MC blocks through the PoS/PoW protocol (normally, it is the main block with the largest PoW workload), it is called the winning MC (main MC). The main MC is based on Epoch. The cycle is connected to a particular line in the entire DAG structure called the MC main chain.

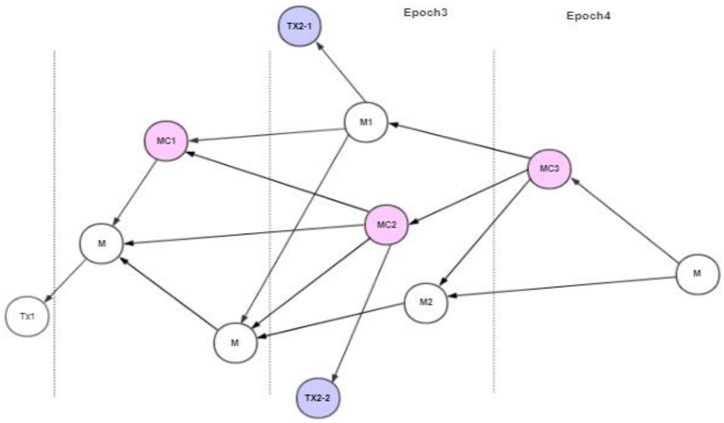
According to the SAT main chain, all transaction TX serial numbers (according to the topological relationship, time, and hash value relationship) can be discharged, and double spends can be excluded based on this serial number.



3.3 Double spending elimination

In the figure, there are two double-spending transactions TX2-1 and TX2-2, in Epoch3. Since the SAT main chain has determined that MC1<-MC2<-MC3, it can be judged through the topological sorting of MC3 that M1 is better than M2, thus TX2 -1 sequence number is better than TX2-2. So far, TX2-2 transaction is excluded, double spendingdetection is successful, and thedetection sorting rules are as follows:

- Blocks can be directly reached by the MC block and can be sorted by the topological order of the block and the main MC
- If there is no path to the MC block, douse the shortest path sequence to reach the nearest path to the genesis block
- If none of the aboverulescan make a judgment,rankby block hash and time



3.4 Merkle hash

The current merkle_hash of each MC of SmartX = hash(MC_hash + Pre-merkle_hash), that is, its hash plus the hash result of all blocks referenced by the MC. Unlike chain blockchains such as BTC, SmartX does not have a one-way traceable merkle hash. However, when looking back at the Epoch-N cycle in each Epoch cycle, the merkle hash of each MC must be consistent, and the SAT hashes accordingly to ensure the final consistency of the data of all nodes at a particular moment.

3.5 Checkpoints and data trimming

As time goes by, the data of the blockchain system will expand to the point that ordinary nodes cannot afford it. In theory, if a snapshot of the data is taken before a specific checkpoint, the account balance table and part of the Merkle hash are retained for calibration Test. The system can still participate in mining and transactions. For ordinary nodes, this will significantly reduce storage pressure.

4 SmartX market and mining

4.1 Deposit contract

SmartX carefully designs economic modelsandPoW&PoS mining rewards to strike a balance between the market value of SmartX and the selling pressure in the PoW/PoS market.

The initial support of the SmartX system will have 24-30 Pos nodes. It is the largest verification node in the SmartX consensus, but not a primary requirement for system operation. No matter how many PoS nodes or PoW node systems are available, they can continue to operate. This is consistent with the principle of BTC/ETH's block producer.

At each block generation, the smart contract automatically screens out the PoS nodes with the maximum number of SAT coins previously pledged as verification nodes.Sothis is a dynamic competition mechanism.Only the staking nodes that successfully squeeze into the shortlisted nodes set by the system in each round can obtain PoS verification rewards. The reward ratio between PoS and PoW is 80:20, and PoS and PoW can be the same node. When a PoW node generates a certain master block and verifies it simultaneously as a PoS node, the node can get both PoS and PoW rewards.

The purpose of SmartX is to better screen out nodes with a more substantial willingness to participate and get a better economic model.

4.2 Smart Contract

DApps developed by smart contracts have greatly enriched the ecology of the public chain. SmartX will support smart contracts temporarily by using Lua virtual machine. In the future, SmartXwill be compatible with the EVM virtual machine. After the release of testnet,the development team will further improve the smart contract.

4.3 Contract deposit node

SmartX supports parallel generation and integration of 25 shard chains. The block address of each shard chain is a Lua smart contract. The contract can start with 10,000 SATs. Users can choose any contract node for scheduled investment. The coins produced by each PoS verification node are divided equally by the total amount of coins pledged by the smart contract. If the tokens produced by PoS are X and the total amount of coins pledged by the contract nodes is Y, then each pledged SAT can be divided into The PoS income is X/Y.

When the user withdraws, the income and principal of SAT will be automatically transferred to the user's wallet after a certain production height is exceeded from the day of the request.

4.4 Contract node interest

When the deposit amount of the contract node exceeds a certain total amount, the interest generated by the depositamount will increase.

4.5 SmartXtokenburn

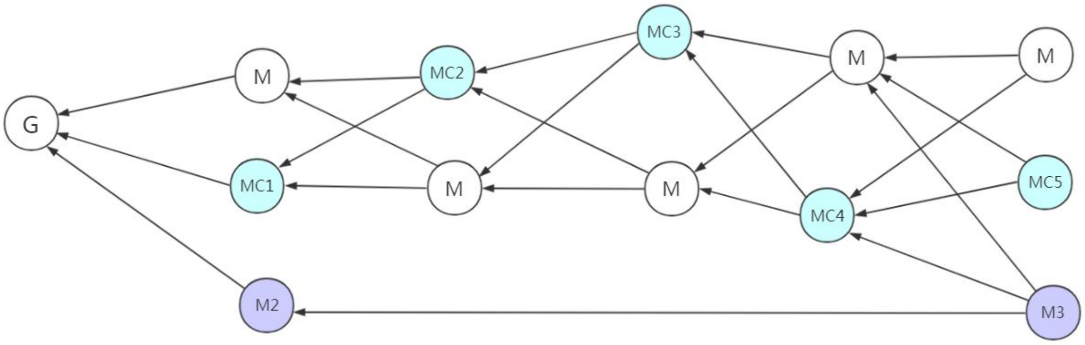
When the deposit amount of the contract node reaches a certain total amount, the team will be triggered to burnecertain amounttokens.

5 Security of SmartX

5.1 Fork chain attacks and solutions

There are two fork chains, the main chain, G<-MC1<-MC2<-MC3<-MC4<-MC5, and the private mining fork chain, G<-M2<-M3, where MC3's difficulty is greater than MC5's cumulative difficulty. In the ordinary DAG network, M3 will generate the winning MC, the main chain of MC5 will be rolled back by the main network, and the M3 chain will become the real main chain. Similar problems may also occur with BTC. This is a 51% attack. Once a node's computing power becomes an enormous computing power, there is a possibility of reversing the transaction.

In SmartX, once such a situation is detected, the 3F+1 node of the SmartX adjudication layer will intervene in processing, preventing the M3 win.



5.2 Timestamp

Like most blockchain systems, the SmartX system has a default presupposition -- the timestamps of most master nodes are almost the same. If the master node time is far from the NTP time, the mining thread will be stopped, and a master node will receive a block from a neighboring node whose time exceeds the NTP time by a certain amount of time, the block will be rejected. If the node time error is within the tolerable range, it is considered that the network abnormality causes the delay.

The time slice is sliced every fixed T period according to the "Greenwich" timestamp. All the main blocks generated according to the current timestamp are mapped to the time slices of Green Time. If they are all consistent, it is considered that the timestamps are the same.

The BTC-style blockchain may have the flaws of timestamp attacks. Since the timestamp is uncontrollable, an attacker can forge the timestamp to cause excessive fluctuations in the timestamp, leading to considerable fluctuations in difficulty adjustment. The direct consequence is that the network cannot produce blocks, or blocks are produced too fast, which increases production in disguise.

The SmartX design avoids such situations; that is, blocks are generated periodicallywithoutadjusting the difficulty. As time goes by, the latest MC difficulty is equal to the sum of the difficulty since first block. The main MC is determined by comparing the total difficulty.

5.3 Sybil Attack

John R. Douceur proposed Sybil Attack (Sybil Attack) in 2002. It is a form of attack that acts on Peer-to-Peer (P2P) networks: the attacker uses a single node to forge multiple. The identity exists in the P2P network to weaken the redundancy of the network, reduce the robustness of the network, and monitor or interfere with the normal activities of the network.

In the SmartX network, to solve the security threats caused by malicious nodes or node failures, each message will require the message sender to sign the message with its private key. The parameters for verifying the signature include message content and block height. As well as the epoch where the current block is located, each public key can only submit a block once in a block generation cycle. If repeated submissions, the earliest received block shall prevail, and other blocks will be ignored. If the verification fails, the block issued by the public key is discarded.

6 SmartX experimental features

Note: The experimental features are exploratory in nature, and the release time is currently uncertain.

6.1 Increase the blacklist mechanism with penalty blocks

Add a type of transaction, called penalty transaction. When the consensus of the whole network meets certain conditions, the PoS decision node will dynamically votetheticket, and the transaction takes effect after the height of N blocks. Penalizing block transactions is mainly used to punish malicious attacks, hackers, and dishonest nodes. The penalties include restricting no transactions within N time, deduction or cancellation of fees, etc.

6.2 SmartX App Store

In addition to the EVM virtual machine and smart contracts, SmartX will introduce an RPC scripting language to enrich SmartX's development ecology. On this basis, a SmartX store, an official SmartX scripting language platform similar to the Apple AppStore, is introduced.

6.3 Dynamic PoW Hash Algorithm

Use script languages such as smartJS to implement the hash function of the workload proof. The PoS decision node initiates an algorithm update transaction and puts the content of the algorithm script in the transaction block. The process is similar to that of adding a dynamic mortgage to the master node. After a certain period, the new algorithm takes effect.

6.4 Encrypted message distribution

SmartX's P2P network can negotiate a passkey between nodes so that data can be transmitted after being encrypted. Decrypted and decentralized social applications are realized through the Ed25519 key system, and the high performance of SmartX provides the most reliable network guarantee for the experience of social applications.

7 Project roadmap

2019/07 Complete SmartX basic framework and basic transaction

2019/09 Improve SmartX's voter voting PoW hybrid consensus mechanism

2020/02 Launch SmartXtestnet

2020/10 release of the Alpha version of Zhitu

2021/04 launches SmartX official website

2021/05 Fully develop the SmartX ecosystem

8 Team Introduction

International developers invested by Singapore and LD Capital Foundation

9 Token distribution

- Total 10 billion
- 43% community airdrop
- 30% node mining
- 17% project development team (locked)
- 5% reserve fund (locked in)
- 5% Early ecological and legal affairs (locked)

10 References

[1] LWMA Difficulty Algorithm, https://github.com/zawy12/difficulty-algorithms/issues/3.

[2] Daniel J. Bernstein, Peter Birkner, Marc Joye, Tanja Lange, and Christiane Peters, Twisted Edwards Curves (2008), https://eprint.iacr.org/2008/013.pdf.

[3]Ethash ·ethereum/wiki Wiki - GitHub, https://github.com/ethereum/wiki/wiki/Ethash

[4]Ethereum. Ethereum. https://github.com/ethereum/wiki/wiki/White-Paper

[5]IOTA. IOTA. http://iotatoken.com/IOTA_Whitepaper.pdf

[6]Byteball. Byteball. https://obyte.org/Byteball.pdf

[7]Cardano.Ouroboros A Provably Secure Proof-of-Stake Blockchain Protocol. https://iohk.io/research/papers/#ouroboros-a-provably-secure-proof-of-stake-blockchain-protocol

[8]Cardano.OuroborosPraos – An adaptively-secure, semi-synchronous proof-of-stake protocol https://iohk.io/research/papers/#ouroboros-praos-an-adaptively-secure-semi-synchronous-proof-of-stake-protocol

[9]J. Chen and S. Micali. Algorand. Technical report, 2017.URL http://arxiv.org/abs/1607.01341

[10] Peercointalk. Peercoin invalid checkpoint.https://www.peercointalk.org/t/invalidcheckpoint/3691, 2015

[11] D. Dolev. The Byzantine Generals Strike Again. J. Algorithms, 3, (1982), pp. 14-30.

[12] D. Dolev and H.R. Strong. Authenticated algorithms for Byzantine agreement. SIAM Journal on Computing 12 (4), 656-666.

[13] P. Feldman and S. Micali. An Optimal Probabilistic Algorithm for Synchronous Byzantine Agreement. (Preliminary version in STOC 88.) SIAM J. on Computing, 1997

[14]Philipp Winter, Roya Ensafi, KarstenLoesing, and Nick Feamster, Identifying and characterizing Sybils in the Tor network (February 25, 2016), https://arxiv.org/abs/1602.07787.

[15]The Sybil Attack.JR Douceur，https://www.freehaven.net/anonbib/cache/sybil.pdf

[16] Go Ethereum - Postal Services over Swarm
https://github.com/ethersphere/go-ethereum/blob/ddfc0a2a02ce574f4c252068ce81f0f5ada1c1ff/swarm/pss/README.md