The Multidimensional Block-Lattice Public Chain with Smart Contract Support that provides a Network as a Service

Tables of Contents

1. Lattice Network Introduction

LATTICE is a highly scalable and eco-friendly Layer 1 blockchain that achieves high throughput and offers compatibility with the Ethereum VM and all other major blockchain networks. Lattice Network is a next generation public blockchain designed for the NaaS, to create a transparent, community – driven, and decentralized ecosystem of products and services. Lattice Network deploys a multidimensional Block Lattice architecture and uses Quantum virtual machines (QVM) to manage and support integrated Smart Contract functionality.

Lattice Networks consensus: Lattice Network implements an AI-powered Delegated Proof-of-Vote (AIDPOV) consensus mechanism for dramatically improved scalability, high security and interoperability developed by the Lattice Network team. Lattice Network is able to deliver a high number of transactions per second (TPS), massive scalability and an inherently decentralized environment for NaaS related decentralized applications (dApps). The framework of Lattice Network will enable everyone to operate network services and benefit from it.

Network as a Service (NaaS) is sometimes listed as a separate cloud provider along with Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). This factors out networking, firewalls, related security, etc.

NaaS can include flexible and extended Virtual Private Network (VPN), bandwidth on demand, custom routing, multicast protocols, security firewall, intrusion detection and prevention, Wide Area Network (WAN), content addressing and filtering, and antivirus.

1.1 Different Network Node Functions in Lattice Network

- **NAT function:** Network Address Translation Node
- **Routing function:** Route forwarding node based on content keyword/DHT/Router table
- **Storage function:** A node with saved content, which can provide contents based on retrieval request from other nodes within the network
- **Security function:** Performs firewall function and enacts security domain access rule

Lattice Network Architecture

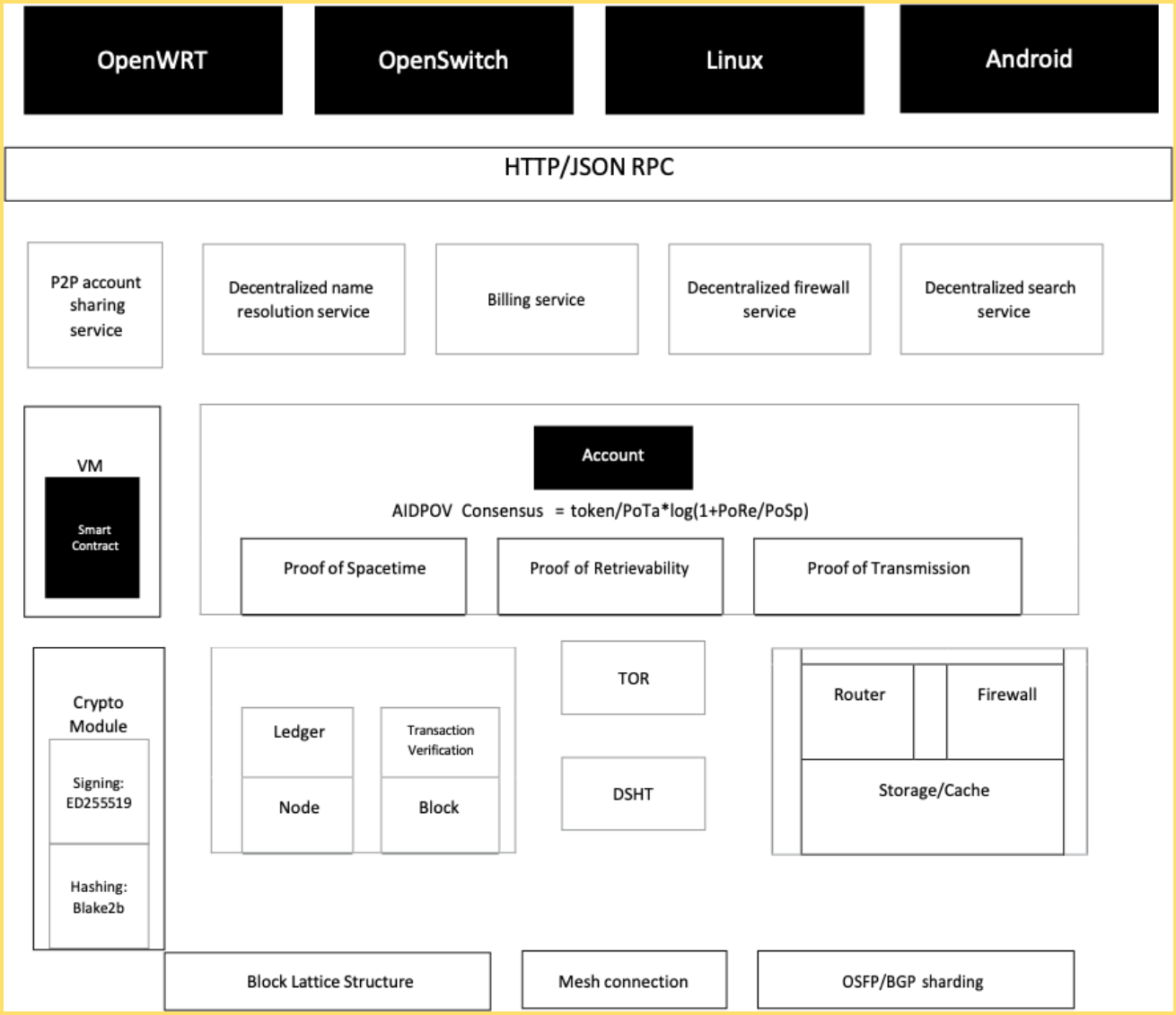

Fig. 1. Lattice Network Architecture

Lattice Network has used the Block-lattice structure introduced by Nano in order to further strengthen OUR AIDPOV

This paper serves to explain the details of Lattice Network, including technical structure, features and advantages.

2. Background

2.1 Block Lattice

A Block Lattice is a block architecture that was first introduced by the Nano cryptocurrency. With Block Lattice, each individual transacting account on the network possesses their own blockchain, which is controlled by the account's private keys. Under the Block Lattice structure, the user's blockchain tracks the account balance, rather than a transaction amount, which allows for less intensive storage and faster transaction speed.

There are 4 types of Blocks in Nano: OPEN, SEND, RECEIVE and CHANGE. They are used for recording transactions. Balances are transferred between blockchains through SEND and RECEIVE blocks.
Nano further introduces the Universal Block to consolidate four types of blocks and encode all account data in every transaction. This design increases the efficiency, improves the security and simplifies codes needed in the network. There are two major improvements:

- Signature checking performance improvement: Before the change, SEND and RECEIVE do not include the account signature, so block signatures have to be verified while running the main ledger insert process and the Input/Output (I/O) operations are blocked from finding associated accounts. This is very time consuming. When the Universal Block contains the account information, an unlimited number of block signatures can be verified without blocking on any I/O operations. Although it increases the block size, the overall performance through TPS and lower transaction processing latency is an acceptable tradeoff.

- Efficient Balance Lookups: The absence of account balance in the current design also occupies long I/O operations. With the implementation of Universal Blocks, we can know the balance simply by looking at one block, instead of searching down the chain for the last SEND Block.

The following illustrates the structure of a Universal Block:

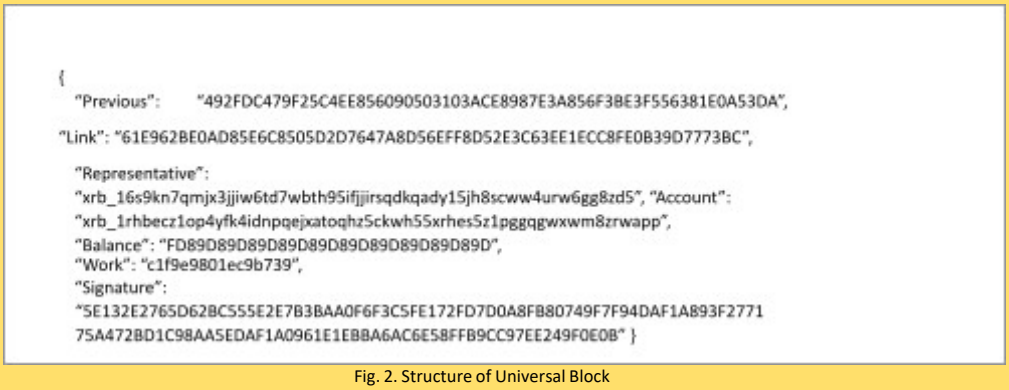

Fig. 2. Structure of Universal Block

Compared with the traditional blockchain architecture used by Bitcoin and Ethereum, Block Lattice delivers almost instantaneous transaction speed and unlimited scalability on low-power consuming hardware, which is highly suitable for network transmission. Block Lattice technology has proven to be stable in its more than two-year operational history. With peak transaction speeds of 7000 TPS and more than 500,000 users on more than 700 consensus nodes, this technology has outperformed most known blockchain technologies.

Lattice Network further advanced the structure by introducing multidimensional Block Lattice to support the Smart Contract and new consensus algorithm especially for network transmission services. Lattice Network team continues the development and aims to build one of highest performing network protocols for the future.

Lattice Network has used the Block-lattice structure introduced by Nano in order to further strengthen OUR AIDPOV

2.2 Network as a Service (NaaS)

Network as a Service (NaaS) is a separate traditional cloud provider along with Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). This family is depicted by following picture

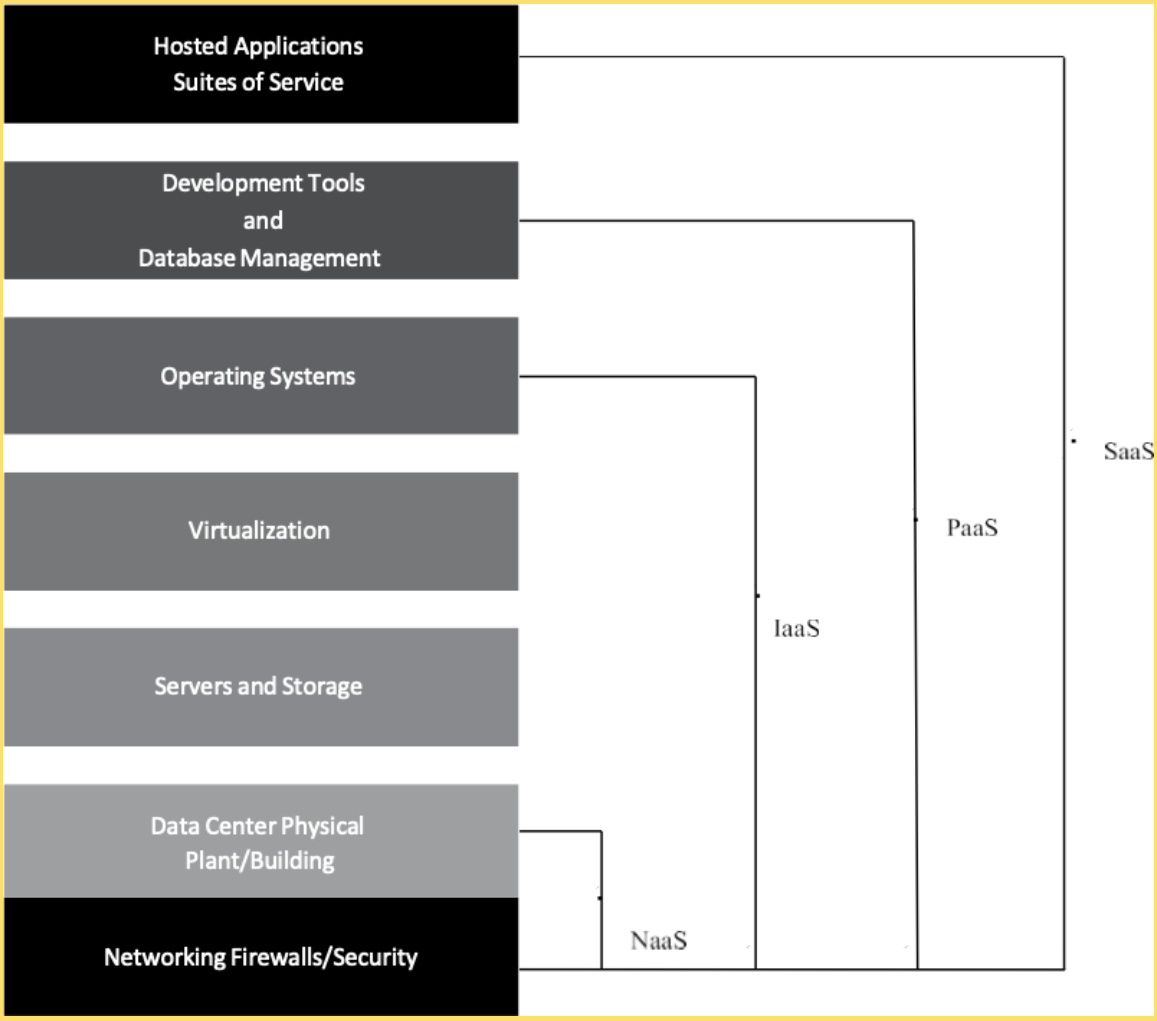

Fig. 3. IT Service Layer

NaaS can include flexible and extended Virtual Private Network (VPN), bandwidth on demand, custom routing, multicast protocols, security firewall, intrusion detection and prevention, Wide Area Network (WAN), content addressing and filtering, and antivirus.

The “AI Protector” of the network detects anomalies (abnormal behaviour of nodes) and recommends optimal values of blockchain parameters.

3. Lattice Network Components

General Account: Each public key/private key pair constructs an account. Each account has its own blockchain recording all engaged transactions. Account owner has authoritative control over transactions related to this account.

External-owned Account: When an account owner issues one Smart Contract, the General Account becomes an External-owned Account and simultaneously generates an Internal-owned Account. The External-owned Account holds the ownership of all assets related to this Smart Contract and is able to issue more than one Smart Contracts.

Internal-owned Account: The issuance of a Smart Contract from the External-owned Account generates the Internal-owned Account. This new account shares the same feature as the General Account, including Send and Receive. Sending the Public Token to the account will activate an Internal-owned Account and trigger the smart contract execution. The transaction will be recorded in a Transaction Block under the Sender Account and the Smart Contract Account.

Transaction Block: General Block recording transactions between general accounts.

Smart Contract Block: On top of Transaction Block, Smart Contract Block stores smart contract instances.

Transaction Ledger: The ledger that records general transactions.

Asset Ledger: Asset Ledger is used to register network assets and to record asset exchanges.

Node: A piece of software running on a computer that conforms to the Lattice Network protocol and participates in the Lattice Network.

Storage Node: New added type of node for the input / output data storage in the Smart Contract Instance.

Quantum Virtual Machine: A virtual machine exclusively for Lattice Network to compile the Smart Contract into an ABI and to provide a secure environment for deployment

4. Lattice Network: Features and Benefits

4.1 Lattice Network's 5 lemma blockchain

The blockchain 5 lemma is the Lattice Networks outlook on the future of distributed ledger technologies, which have to balance between **speed**, **security**, **scalability**, **computational effectiveness**, and
eco-friendliness.

Comparing the 5 lemma to other blockchain operational consensuses, it is required to optimize all 5 pillars for the future of blockchains, unlike others where it's not possible to optimize all 5 at the same time without any tradeoffs.

For example, a distributed ledger such as Bitcoin has arguably strong security through its consensus protocol and decentralization, but gives up speed, scalability, computational effectiveness, and eco-friendliness as a result.

Lattice Network addresses the challenge by achieving an asynchronous foundation. Lattice Network's AIDPOV consensus that runs on Quantum computers, allows transactions to be processed asynchronously, increasing the speed and the throughput of transactions, the computational effectiveness, achieving new heights in security, cutting-out mining, allowing scaling to a next level for technologies, which is incomparable to synchronous BFT ledgers such as Ethereum and Bitcoin.

Lattice Network achieves decentralization and security through a chosen permission factor and a next generation consensus protocol, in which anyone can join and leave the network at any time and all nodes are equal.

4.2 Multidimensional Block Lattice Structure

In Block Lattice structure, every account has a unique blockchain to record its own transactional information. With Smart Contract functionality, Lattice Network supports multiple token issuance within one account. Each account supports multiple tokens and each new token added will be mapped to a new chain within the same account, so that each account can have multiple chains. Each token has its own “OPEN Block” in every single account. Since one token/one chain is one dimension, the structure with multiple tokens creates a multidimensional Block Lattice.

Each blockchain for an identical token is independent from others. The underlying structure of each token blockchain carries the Block Lattice structure and thus stays concise and agile.

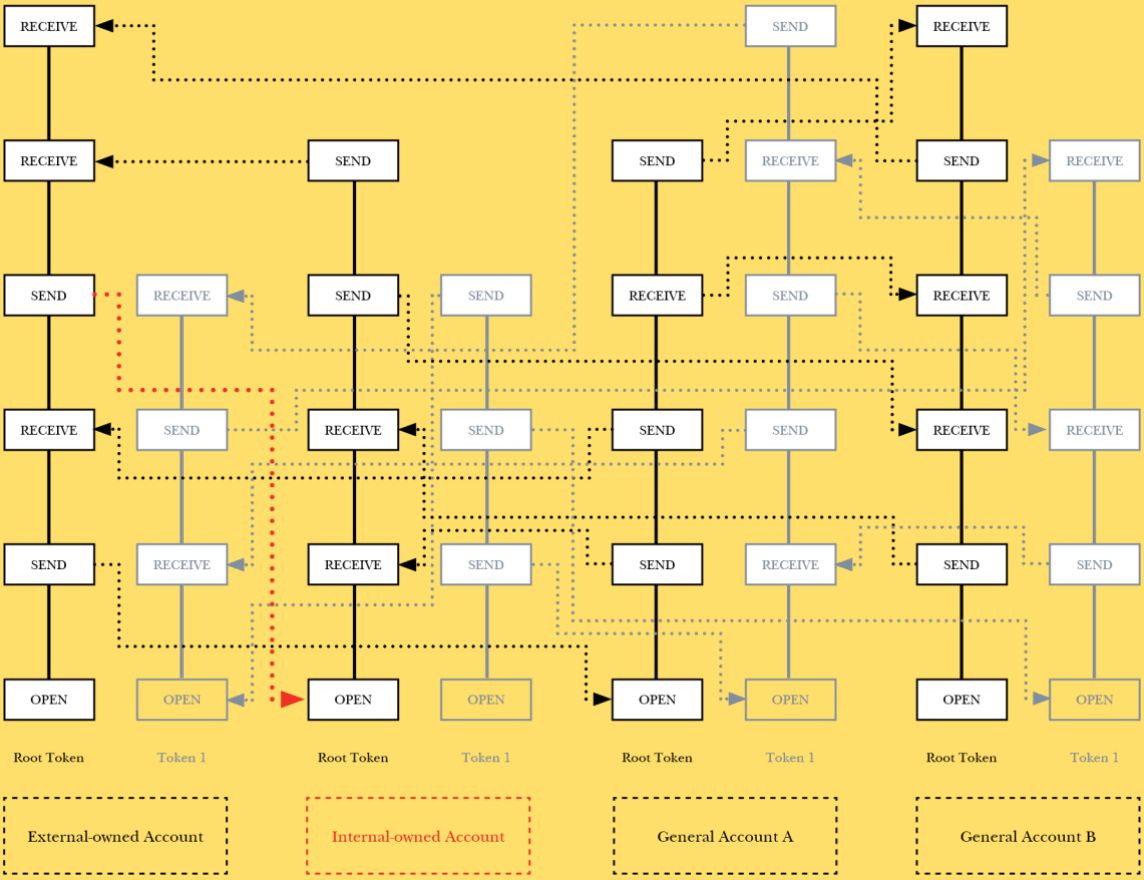

Fig. 4. Multidimensional Block Lattice Example

Multidimensional Block-Lattice structure brings Lattice Network the following benefits:

Low Transaction Validation Latency

The use of independent account-chains enables the user accounts to be updated asynchronously, without the need to involve the entire network. The dual-transaction approach leaves the process of transaction verification to the affected accounts, such as the sender and the receiver. This option eliminates the need for miners, meaning that transactions are instant and with zero fees. The network, therefore, becomes more scalable and agile.

Scalability

Transactions on LATTICE are handled independently of the main ledger. Every transaction is also an independent blockthat fits into a User Datagram Protocol (UDP) transactional packet and recorded as a unique block. UDP's are transactional packets that help keep computational costs low, allowing you to send transactions to accounts thatare offline. Using a system of references and hash pointers eliminates issues relating to block size and allows the network to scale without all nodes having to hold a copy of every transaction ever made. Rather, nodes store themost recent and current blocks of each account-chain. Consequently, the network can achieve a drastically higherscale than other blockchain networks. This is where block lattice and mainstream blockchain differentiate. A transaction on the blockchain cannot be isolated and recorded on the main chain. A specific number of transactions are verified before being added to the main chain. This means increased transactions lead to a steady decline in speed, slowing down the entire network. Lattice Network uses "account chains" to create a lighter network, reducing the problems of scalability that blockchain-based solutions often encounter.

Low Energy Consumption

The Lattice Network is built upon a AIDPOV architecture: AI-powered Delegated Proof-of-Vote (AIDPOV). This consensus can achieve low energy consumption because it does not require mining activity. All energy is contributed to make effective computing. Both consensus mechanisms will be elaborated later in this paper.

Inherent Anti-Centralization

Mechanism guaranteed anti-centralization refers to the fact that each account has its own ledger, namely, the account-chain structure, and validation is conducted by delegates via an asynchronous mode. This is unlike the Proof-of-Work (PoW) consensus used by Bitcoin, where ledger generation and confirmation is completed by miner nodes; and unlike the Proof-of-Stake (PoS) where transaction validation is based on the number of coins a validator stakes.

In addition, the structure of the anti-centralized Block Lattice requires that the transaction sender and receiver to conduct a small computational effort input - local PoW process. This process has decreased the possibility of transaction centralization, similar to how a decentralized exchange decreased the possibility of super account formation.

Another important mechanism guaranteed anti-centralization factor is the Lattice Network AIDPOV consensus of theLattice Network, which will be introduced later.

4.3 Lattice Network Smart Contract

Lattice Network enables Smart Contracts and Distributed Applications (dApps) to be built on Block Lattice, which brings the advanced structure beyond just supporting a digital currency. By introducing Smart Contract, Lattice Network defines the Smart Contract as an account which owns its account-chain and designs the Smart ContractBlock.

Lattice Network supports two types of Smart Contract: the Token Smart Contract - for new token issuance in theecosystem, and the Asset Smart Contract - for digital asset registration without new token generated.

4.3.1 Explanation of the Smart Contract Block

Lattice Network Smart Contract contains two parts: 1) Contract handle for Smart Contract addressing, and 2)Contract instance for saving ABI and contract signature.

The structure of block is illustrated as following:

- Smart Contract Handle: “SC_INFO_HASH” is null for an ordinary transaction block. In the transaction concerning the Smart Contract, this field must not be null. The combination of “link” and “SC_INFO_HASH” can accurately classify the Smart Contract related transaction from ordinary transactions.

Fig. 5. Transaction Block

- Smart Contract Instance: SC_INFO_HASH is the hash value from the Smart Contract Block illustrated in Fig. 5, which records the original data of Smart Contract ABI.

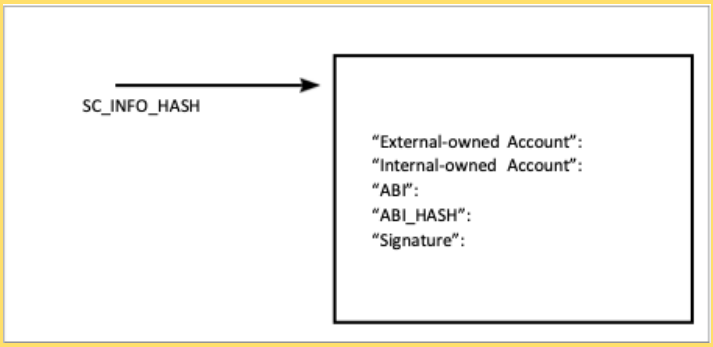

Fig. 6. Smart Contract Block (SC_INFO Block)

4.3.2 Properties of the SMART CONTRACT Block

The SMART CONTRACT Block used in Lattice Network has the following properties:

- The owner of the Smart Contract activates the account by sending Root Token to the Smart Contract account.
- By executing the file in the contract, the owner signs the block with his private key.
- The SMART CONTRACT Block is reserved for an ABI. The Lattice Network Virtual Machine compiles the SmartContract into an ABI and further provides a secured deployment for the Smart Contract.
- The virtual machine further retrieves the ABI by loading the SMART CONTRACT Block and calling the function of the Smart Contract.
- The consensus protocol of the Smart Contract block is completely the same as in the transaction block.
- Extend Smart Contract Block to support the Smart Contract related data storage so that the Asset Smart Contract can be saved.

4.3.3 Token Smart Contract Protocol

- The External-owned Account sends SC_INFO to other nodes.
- External-owned Account sends a SEND Block containing a Public Token transaction to activate the Internal-owned Account. Nodes in the network will verify the Smart Contract based on the SC_INFO_HASH in SEND.
- After confirming the SEND Block, External-owned Account issues the OPEN Block in the Internal-owned Account chain and the Genesis Block of token by utilizing “Init()” in the Smart Contract and consequently broadcasts to the entire network.
- When each node receives the broadcast, it updates the local account information and calculates the balance of the Root Token for the Internal-owned Account and the balance of the new token. The process of issuing a Token Smart Contract has been completed.

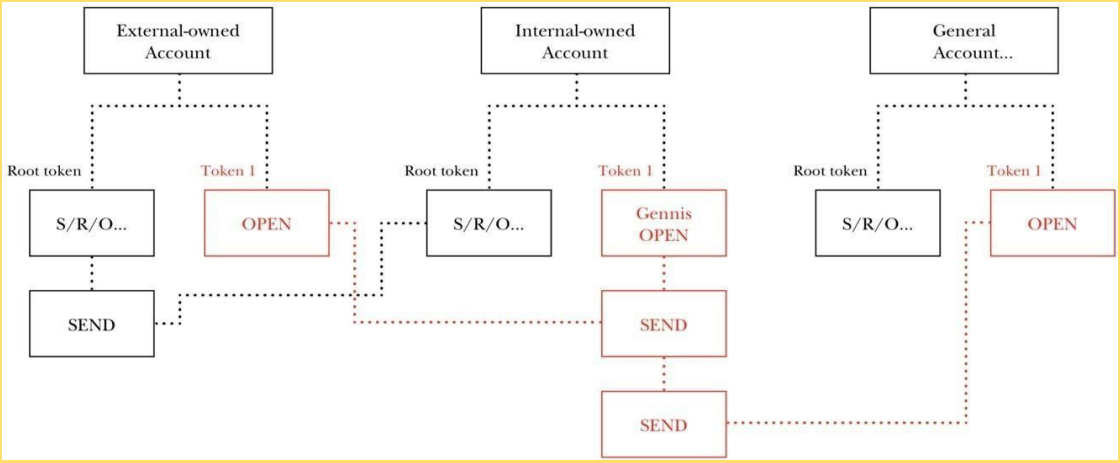

Fig.7. Release Procedure of a Token Smart Contract

4.3.4 Asset Smart Contract Protocol

After the Asset Smart Contract is released, the real asset will have a digital identity and a Lattice Network accountthat affirms the ownership relationship between the real asset owner and the real asset on the blockchain. In addition, the owner can provide services with the registered asset.

Asset Smart Contract follows below principles:

- Asset Smart Contract allows the asset owner to register the real asset on the blockchain without new token generated.
- Asset Owner Account produces the Asset Smart Contract Account when issuing the Asset Smart Contract.
- When Asset Smart Contracts are deployed, the Asset ledger will be generated on the Lattice Network for assetinformation record.
- User triggers the execution of the Smart Contract by sending the transaction to the Asset Smart Contract Account. The transaction can be completed in Public Token or other Tokens.
- Asset Smart Contract Account will provide corresponding services after receiving the payment.
- The block storage requirement brought by the Asset Smart Contract is stored on the storage node on the Lattice Network.
- The transaction history from the asset ledger is saved in the storage node of Lattice Network and further confirmedthrough the AIDPOV Consensus.

4.3.5 Features of Lattice Network Quantum Virtual Machine

Lattice Network Quantum VM architecture is illustrated as the following:

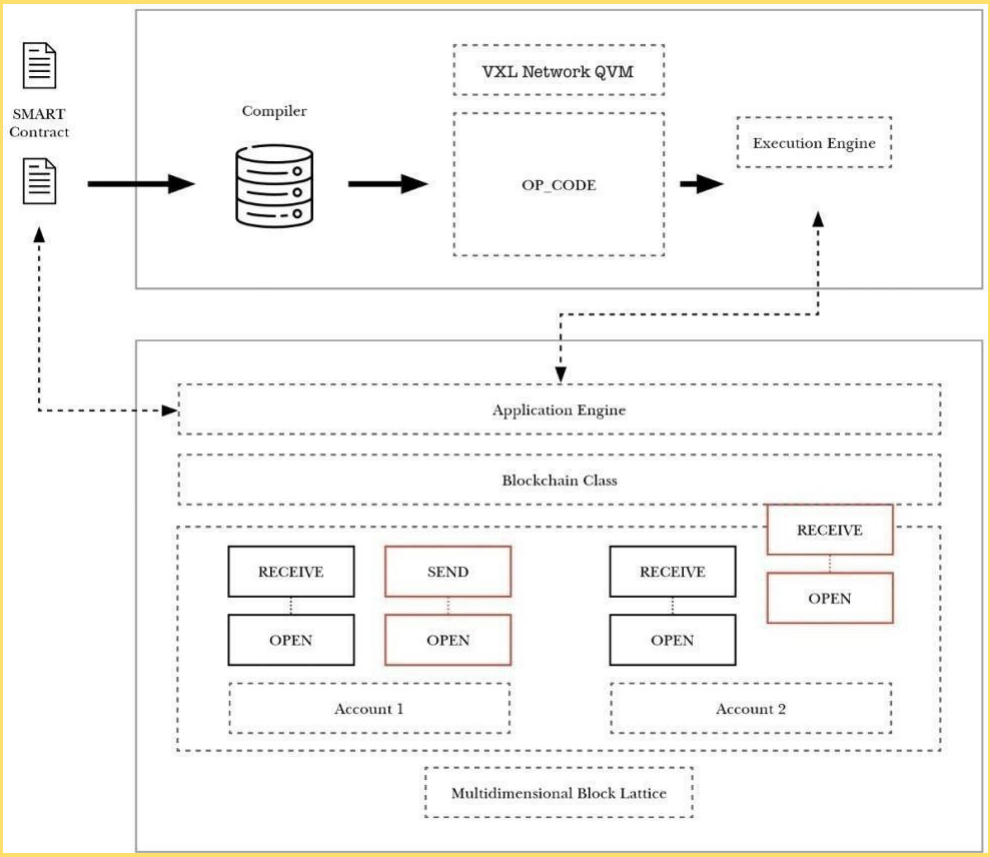

Fig. 8. Lattice Network Virtual Machine

- Lattice Network Quantum VM is Turing complete and supports native language such as Golang/C#/Java/C++.

- The compiler translates Smart Contract into the “OP_CODE” which is further converted into an ABI. The result of the Smart Contract is stored in the block lattice.

- Lattice Network Quantum VM bridge is configured for high performance which allows those applications to run faster and smoother with Lattice Network’s increased transaction per second (TPS) capabilities.

- Lattice Network Quantum VM is the fastest VM chain to compete with Ethereum 2.0.

4.4 The Lattice Network AIDPOV Consensus Protocol

Lattice Network designed an AIDPOV consensus protocol for global agreement. A multi-dimensional Block lattice structurepowers our proof-of-vote (PoV) consensus algorithm.

Unlike Proof-of-Work and Proof-of-Stake, which require miners to solve complex computational problems to verify transactions and secure the blockchain, Proof-of-Vote does not use mining at all. This means that a blockchain secured by Proof-of-Vote is fully eco-friendly. Using a Proof-of-Vote consensus algorithm allows every account to freely choose a representative at any time to vote on their behalf, even when the delegating account itself is offline. Additionally they are fault tolerant and contain very strong mathematical proofs to ensure their security and stability. These representative accounts are configured on nodes that remain online and vote on the validity of transactions they see on the network. The validity of a transaction is determined by rules set out in the blockchain, which when the software is open-source (as LATTICEs software is), specifies how transactions are to be validated. These rules prevent the network against malicious or dishonest nodes.

4.4.1 The Background of the Lattice Network AIDPOV Consensus

1) Economic Thoughts behind the Lattice Network AIDPOV Consensus

- Historically, both Proof-of-Work (PoW) and Proof-of-Stake (PoS) protocols have drawbacks. PoW is plagued by concentration in hash-rate, while PoS is the game for the wealthy (nodes with more tokens are more likely to be selected for voting). Although both Proof-of-Work and Proof-of-Stake acknowledge the issue of centralization by increasing the cost of “being evil,” it is still possible that the network is compromised or manipulated by miner-alliances or large-stake token holders and eventually discourage the fairness of the ecosystem.

- In PoW and PoS protocols, the capacity to independently verify transactions and add new blocks to the ledger (or blockchain) is proportional to the size of an individual’s capital stake. In contrast, we believe that key players in the Cryptocurrency market should be the “middle-class” and “poor,” who are able to enhance liquidity and dissemination. However, if transaction fees are charged for such a process, it will not only be evidence of the Matthew Effect but also destroy the fairness of the cryptocurrency ecosystem.

- Similarly, Delegated Proof-of-Stake and Byzantine fault tolerance, both modified versions of Proof-of-Stake, sacrifice decentralization by selecting a group of representatives to vote. This can lead to problems such as voting manipulation. As a result, the consensus is hindered by the concentration of powerful parties.

2) The goal of Lattice Network AIDPOV Consensus Protocol

- PoV is measured by effective workload. The amount of rewards depends on the ratio between number of tokens on hand and effective workload. By distinguishing between nodes that transmit information and nodes that verify information, we can reduce the Matthew Effect. Lattice Network encourages more “Middle Class” nodes to participate in managing the ledger and getting rewards. The “rich node” will be rewarded bytransmitting transactions for resource contribution and making secondary distribution to “Middle Class.” Resources become more fairly distributed and the overall network becomes more secure.

3) Illustration of attacking scenarios in Lattice Network AIDPOV consensus

- In an example scenario, a number of nodes on LATTICE's Network are holding a proportionately large amount oftokens, however they provide very little workload. These nodes are more likely to be selected as the bookkeepers, aka transmitting tasks and managing the ledger. Over time, they inevitably spend tokens in these ongoing transmission tasks. Yet, their bookkeeping power is gradually impaired when workload and "wealth" are downsizing.
- Another example are nodes who accumulate their share of tokens by purchasing a large amount from the open market, yet take on a very limited workload. Eventually, they will run into a "Nothing in Stake" problem, which they can prevent by raising the price of tokens to increase the malicious cost. Our nodes can additionally prevent this attacking scenario by disconnecting with the malicious node to minimize its workload so it is not able to bookkeep the ledger anymore.

- Let's now look at nodes on LATTICE's Network who instead contribute an intensive transmitting workload, but are holding a proportionately small number of tokens. These nodes accumulate "wealth" by working hard and consequently become superior in bookkeeping power. This is fair and derived from the market design and mechanisms. It cannot be predicted or prevented.

- Let's look at the nodes in the instance above who work intensively to get tokens but later speculate tokens for profit. When they downsize their token holdings, they lose bookkeeping power. We avoid the drawback of PoW and PoS in concentration of hash-rate because speculating nodes are not able to simultaneously accumulate both the scale of wealth and bookkeeping power.

- Lattice Networks' innovative AIDPOV Consensus mechanism is designed to achieve mean reversion of token or hash-rate distribution, which prevents extreme centralization of tokens or hash power. Additionally, a dynamicrole conversion exists between token owners and those who provide network bandwidth: nodes with large bandwidth contributions are rewarded with tokens, while nodes with moderate bandwidth will be converted to "ledger nodes" (nodes that maintain the blockchain ledger) and still be rewarded with token sharing. Neither the Ledger nor transmitting nodes have any incentive to take the risk of arbitrage and act maliciously.

4) Deduction of the Lattice Network AIDPOV Consensus from PoW/PoS Consensus

Proof-of-Work (PoW) satisfies the following mathematical equation below. Proof: Follows the below steps:

$$SHA3(previous\ block\ hash\ \ nonce\ time\ stamp, Merkel\ tree\ root\ ) < target$$

The solution ultimately indicates that Bitcoin nodes have a uniform distribution…

$$P(X > x) = \frac{1}{x}\ (x > 0)$$

However, due to variables like the feasible conversion between Bitcoin and legal currency and the widespread application of ASIC chips, hash-rate is artificially centralized. The actual PoW equation in each node adjusts for N, the coefficient of hash-rate concentration in a given node which is significantly correlated to the miner's economic strength.

$$\frac{SHA3(previous\ block\ hash, nonce, time\ stamp, Merkel\ tree\ root\ )}{N} < target$$

We believe N falls under Pareto Distribution:

$$P(N > n) = \frac{x}{min(x)}^{(-k)}$$

- ➔ *X is any number > min (x)*
- ➔ *min (x) is the minimum positive value of x*
- ➔ *k is a positive parameter*

If we define N as the amount of tokens held by a given node, we derive the following PoS equation:

$$SHA3(previous\ block\ hash\ \ nonce\ time\ stamp, Merkel\ tree\ root\ ) < target \times N$$

However, the equation above does not comply with Bitcoin's original intention that one CPU = one vote. The gradual change to the value of the right side of the equation during mining transforms the solution from the uniform distribution to Pareto Distraction gradually. In order to maintain the solution in uniform distribution, we have to introduce a new coefficient to the left hand side of the equation to neutralize the impact.

The process of solving a hash function is similar to a continuous process of producing entropy. The entropy synchronizes the accumulation with the concentration of hash-rate.

$$H(E) \quad -Sigma\ [\ p(e) \times \log_2(p(e))\ ]$$

We add entropy as the new coefficient to the right. In reality, the more the token held in PoS, the more active the node for transmitting data. Simultaneously, the number of bytes transmitted follows Pareto Distribution and can be presented by the following Lattice Network AIDPOV formula:

$$\frac{SHA3(previous\ block\ hash, nonce, time\ stamp, Merkel\ tree\ root\ ) \times E}{N} < target$$

We modify the above by placing the Lattice Network AIDPOV Coefficient to the right.

$$SHA3(previous\ block\ hash\ \ nonce, time\ stamp, Merkel\ tree\ root\ ) < \frac{target \times N}{E}$$

This is the Lattice Network AIDPOV Consensus deduction process.

Considering the network scenario, we can replace the Lattice Network AIDPOV entropy with capacity of channeltransmission. We introduce a new vote coefficient that measures the marginal value of PoV.

$$\boxed{Vote} = \frac{Token}{PoTa \times \log_2 \frac{1+PoRe}{PoSp}}$$

where:

- ➔ *PoTa: the total traffic on the Lattice Network including upload and download of nodes.*
- ➔ *PoRe:the upload traffic to other nodes on the Lattice Network.*
- ➔ *PoSp:the storage for data produced by transaction on the Lattice Network*

And the following condition has to be met:

$$SHA3(previous\ block\ hash, nonce, time\ stamp, Merkel\ tree\ root) < nonce \times stake$$

5) Implementation of the Lattice Network AIDPOV Consensus and related algorithm

a. Global election of validation node

a1. Overlay a hash addressing mesh network on top of the conventional physical network. Mesh network will perform a global next hop which is the random next hop in TOR network. It updates every 10 minutes and ensures the next hop is 7 bytes.

b. Account balance for vote

b1. Under the account balance , each node creates its own account balance and a ledger of the global network balance.

b2. Each account has a private key for the local ledger based on the elliptic curve cryptography. The private key is immutable.

c. Sharding model in network consensus

c1. Shard from OSPF/BGP/VLAN. Each network slice reaches individual consensus and different shards reach secondary consensus through edge network gateway.

c2. The local ledger of each node validates through the Lattice Network AIDPOV Consensus within the individual network slice. Normally, step a1 is adequate to complete the process. The following attackscan also be effectively prevented by the Lattice Network AIDPOV Consensus in global network:

- Double spending fork attack
- 51%+ Attack
- Sybil Attack
- Network Storm that causes election failure of voting nodes
- Quantum Super Computing attack

References

1. A Fee-less Distributed Cryptocurrency Network. Retrieved from https://Lattice.network

2. Block Lattice - Github https://github.com/nanocurrency/raiblocks/wiki/Block-lattice

3. Network as a service - https://en.wikipedia.org/wiki/Network_as_a_service

4. Chantel Cary. Ovum Decision Matrix: Selecting a Real-time Convergent Billing and Charging Solution, 2016

5. A.M Antonopoulos. Mastering Bitcoin, 2014.

6. K. Christidis and M. Devetsikiotis. Blockchains and Smart Contracts for the Internet of Things. IEEE Access, 4:2292 - 2303, 2016.

7. A. Kiayias, I. Konstantinou, A. Russell, B. David, and R. Oliynykov.

8. A Provably Secure Proof-of-Stake Blockchain Protocol, 2016.

9. S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf