Reforging The Cornerstone Of The Blockchain World

2017.12.31

Summery

Satoshi Nakamoto first invented bitcoin in 2008 as a "decentralized electronic cash system," integrating various computing technologies such as P2P communications, cryptography, chain data structures, etc. together and forming a decentralized, non-trust and game-based autonomous system - the blockchain. It followed by Ethereum implementing a Turing-complete smart contract concept, which is with the capability of doing logic processing and value transfer over different entities on the chain. Since then, more and more people are aware of the huge potential of blockchain technology. They started to study blockchain technology and promote the continuous development of blockchain infrastructure.

However, connections between the blockchain and the physical world are still rarely observed. It mainly due to the limitations of the current blockchain platforms: 1) the capacity of the blocks is limited. Recording massive transactions could easily lead to congestion. 2) The consensus algorithm itself lacks flexibility. It’s not easy to introduce other factors in the consensus algorithm to comprehensively measure contribution of chain users. 3) The existing blockchain systems are largely closed in the sense that the smart contracts only accept on-chain events as trigger and lack the interaction with the physical world.

We are aiming to build a brand-new blockchain ecosystem — CanonChain and push the blockchain technology to a higher dimensional space so that it can eventually serve as a value transfer protocol for the future Internet. Through our innovation on variant technologies covering P2P communication, consensus algorithm and smart contracts, physical layer protocol redesign, we are finally to make CanonChain the bridge between the blockchain and the physical world.

CanonChain drives current Internet of Things (IoT) to the era of Fog of Things (FoT). Although various devices in IoT are distributed in different areas, the interaction between them and all the operations are performed through the cloud server. Essentially, IoT is still a centralized architecture with many bottlenecks in large-scale expansion: soaring costs, system congestion, reduced reliability, and server vulnerability. In addition, the devices' data are also owned by service providers, individual users cannot fully access their own data.

Based on decentralization requirement as a foggy network, in CanonChain, we have redesigned some elements of the traditional blockchain:

- Consensus algorithm based on devices' resource contribution such as computation, storage, and communication (Section 3.1).
- New smart-contract design to accommodate both on-chain and off-chain events so to enable value exchange between inside and outside of the chain (Section 3.2).
- Introducing a new transaction rate mechanism to achieve social optimal allocation of blockchain resources (Section 3.3).
- New cryptographic structure and access control allowing individual users to have full access to their own data (Section 3.4).
- Extensive integration of existing third-party software, such as distributed file systems, distributed databases, to combine their strengths with blockchain technologies (Section 3.5).
- Optimized hardware interface allows heterogenous devices to have seamless access to CanonChain (Section 3.6).
- Built-in token exchange system to facilitate value exchange between different DeOS (Decentralized Operating Systems) operating in the foggy network.

With these innovations, the users, devices and services are connected in a decentralized way without going through the central server. The above-mentioned centralization problem won't present in CanonChain. CanonChain is eventually evolving into a hardware and software combined ecosystem. All devices or systems running on CanonChain can be considered as citizens in the community; they buy production from others; they contribute their own productivity to the others and get payment; they pay a certain amount tax; they play a game under regulation.

We will build ecology to make CanonChain sustainable. We can not guarantee that the entire CanonChain project will ultimately fulfill our vision. However, what we can make sure is that once the entire project is started the community-based maintenance will be endless.

CanonChain is committed to creating an ecosystem involving open source communities around the world and third-party developers. With developer reward program, CanonChain encourages

third-party developers working together to promote blockchain innovation across open source applications and industries. The goal is to make CanonChain the medium for exchange of information and value among individuals in the connected world.

To achieve sustainable development of CanonChain and to avoid the scattered structure, we will create CanonChain Foundation to do community governance, code management, financial support, franchise operations and other operations. In meantime, the governance structure will be continuously updated following the development of foundations and communities. The CanonChain Foundation will collaborate with various business and government partners and share resource and opportunities with each other.

Contents

1 Introduction

1.1 CanonChain and the foggy network

Figure 1. CanonChain and the foggy network

Technically speaking, CanonChain is a decentralized supercomputer formed by the idea of blockchains. This supercomputer runs on any network device (such as a computer, smart phone, smart watch, vehicle, gamepad, street light, etc.) with CanonChain protocols and other DeOS (such as Ethereum, EOS, etc.). In the supercomputer, terminal devices provide input and output, data is processed in the foggy network controlled by CanonChain. Section 1.2.2 illustrates the structure of the supercomputer in details.

From the physical world point of view, CanonChain is an open, secure and trusted decentralized operations organization. All connected devices are citizens of the CanonChain: they buy products from other; they contribute their own productivity or means of production for remuneration; they pay a certain amount of tax; they play games under regulation.

Different from current IoT devices, the devices that operate in CanonChain are distributed. Services acquired by a CanonChain device is provided by one or multiple devices on the chain and cannot be identified, that's why we call it a foggy network.

Canon Chain designs a self-evolving system for foggy network based on value incentive from the chain:

• Define the measure of value: CanonChain defines how to measure the contribution of devices in the foggy network and how to provide incentive to the contributing devices. With the accurate measurement of contributions and appropriate incentive to the devices, to guide and achieve the maximum value of the foggy network.

• Build community ecology: Build a community that provides developers with a friendly, positive feedback mechanism to build a thriving decentralized application ecosystem.

• Self-evolution capability: The governance in CanonChain can guide the foggy network evolving towards faster, stronger, and better user experience without excessive human intervention.

CanonChain can not only support traditional Internet and IoT services such as: social, e-commerce, games, home security, medical alarm etc., but also support other new services that will dominate in next era.

• Services that require a large amount of interaction and collaboration between nodes. For example, self-organizing mesh networks, and scene detection in vehicular network.

• Services that require a large amount of computing power. For example, big data scientific computation, artificial intelligence training.

• Services on other blockchains that follow the underline consensus algorithm in CanonChain.

1.2 Technical characteristics

1.2.1 Geometric philosophy

Geometry is one of the most wonderful subjects in mathematics. It has both beautiful graphics and delicate propositions. The earliest word of geometry is from Greek "γεωμετρία", which is to study the spatial structure of the nature. In ancient times, people constantly accumulated and

mastered various concepts of plane, line, square, circle, length in practice and gradually realized their positional and quantitative relationships. Modern geometry has developed many branches such as differential geometry, topology etc. and is used in different fields such as surveying and mapping, architecture, astronomy and computer.

As a new subject in computer science, blockchain has great relevance to geometry. The P2P network in a blockchain involves the knowledge of topology. The cooperation and competition among miners in a blockchain can be studied by using game theory and differential geometry. After a blockchain grows to a certain volume and reaching a certain extent of coverage, the development of each region will have statistical self-similarity and fall into the category of fractal geometry. If other factors other than computational power, such as storage and bandwidth, are introduced into consensus algorithm, the block chain goes to a higher dimension.

The purpose of CanonChain is to create a super computer and network system with disperse storage, computing and bandwidth based on blockchain technology, which is extremely complicated. To simplify the design and abstract the idea of CanonChain, we borrowed the concepts of "ruler" and "compass" in geometry, and developed the *CanonChain Zen Ruler* (CZR) and *CanonChain Zen Compass* (CZC) as tools to study and design CanonChain. Ruler and compass are the most basic research tools in geometry. Rulers is used to measure and compass is used to regulate drawings. In the Canon Chain world, CZR measures the performance of components while CZC regulates the interaction of components. The objective of CZR and CZC is to find the optimal solution of CanonChain in different use cases, and serves as the engine driving CanonChain development and evolution.

1.2.2 CanonChain + foggy network = supercomputer

Figure 2 shows the structure of a supercomputer combined with CanonChain and foggy network:

• CZR and CZC are the core of the entire computer. They define the basic operation rules and supply core security mechanisms. They can be analogized to the BOIS in an ordinary computer.

• The other components in CanonChain, such as consensus algorithms, smart contracts, etc., instruct the operation of the computer and coordinate devices in the foggy network. They can be analogized to the CPU in an ordinary computer.

• The foggy network provides computational power to process data and can be analogized to the GPU in an ordinary computer.

• The devices in the foggy network are connected by a P2P network, which can be analogized to the system bus in an ordinary computer.

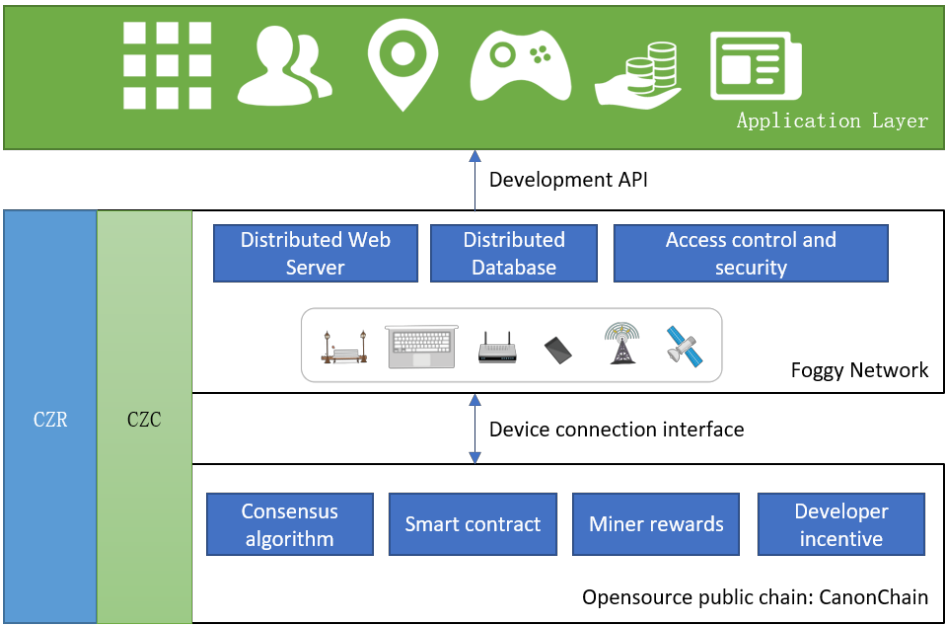


Figure 2. CanonChain supercomputer

We install and integrate third-party software such as distributed Web servers, distributed databases etc. and provide development API. Developers can build application and construct ecosystem on top of these.

Table 1 summarizes the correspondence between various components in CanonChain and an ordinary computer:

Computer	Canon Chain + Foggy Network	Function
BIOS	CZR/CZC	Provide basic rules and core security mechanism
总线	P2P network	Data transfer between nodes
GPU	Foggy network	Data processing
CPU	CanonChain	Logic processing，node coordination and resource management

Table 1. CanonChain/Foggy network components and the corresponding computer parts

1.2.3 CanonChain Zen Ruler (CZR) and Canon Chain Zen Compass (CZC)

CZR is a component comprehensively measuring the value of elements in CanonChain and the foggy network; CZC is a component that standardizes all aspects of behavior in CanonChain and the foggy network. The existing blockchain solutions mainly focus on the technology of improving the blockchain itself, and only look for an optimal solution in the blockchain. However, CanonChain combines the blockchain technology (on-chain) and foggy network characteristics (off-chain) together and do cross-layer optimization, aiming to find the global optimal solution such as shown in Figure 3.

CZR measures CanonChain and the foggy network performance and plays a critical role in building the safety and stability of the CanonChain. The on-chain value measured by CZR include size of the blocks in the chain, time of generating blocks, participation of the miners, etc. The off-chain value measured by CZR includes P2P transmission delay, bandwidth, storage, load and other capabilities of the devices and stability, spread speed, usage frequency, token exchange rate of the applications. CZR post-process the measurements obtained and supply the analysis results to CZC.

After receiving updated data from CZR, CZC set out to re-establish the value system of CanonChain: build a more sophisticated model, tap more dimensions, and adopt a more robust and energy efficient consensus algorithm. After determining the changes in the value, CZC can propose to the community to upgrade CanonChain.

CZC also defines the principle of governance in CanonChain (Section 1.2.4). CanonChain uses CZC to establish a P2P service agreement or binding contract between signed users. CZC defines the obligations between users that cannot established solely on computer programs. At the same time, CZC establishes the jurisdiction and laws based on the generally accepted rules among each other. For each transaction broadcast on the chain, it's signature must include the hash of the CZC to regulate the contract signer.

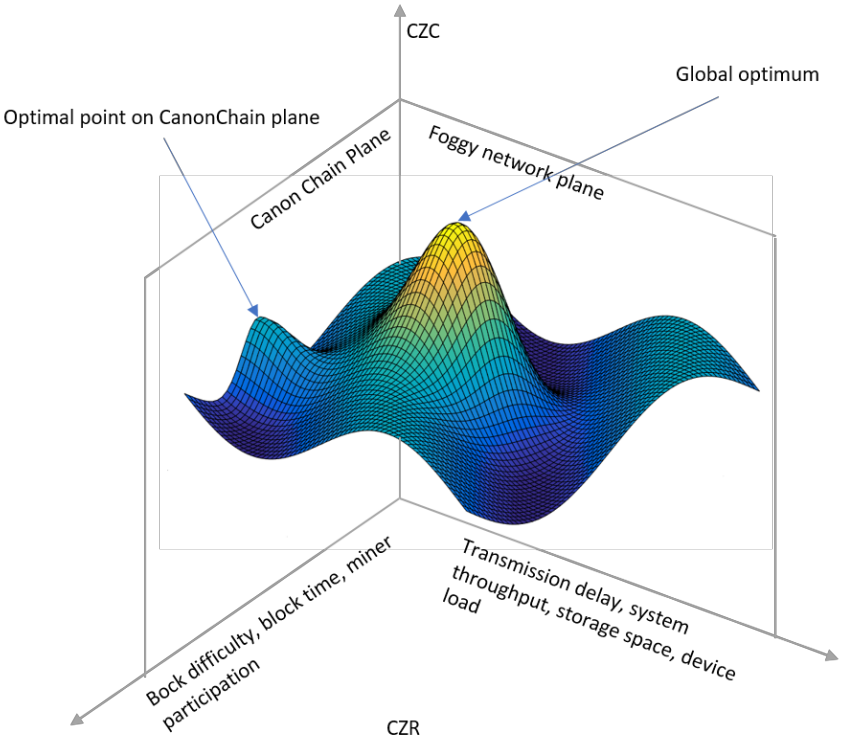


Figure 3. CZR and CZC do global optimization on CanonChain and the Foggy network

1.2.4 Governance

When in the process of upgrading blockchain protocols or when there is security vulnerability, we often see some temporary, informal, and even controversial governance processes in the blockchain community, which could create hard-forking or produce unpredictable results. For example, bitmain is a hard-fork from bitcoin because of the disagreement over block size. Ethereum had a hard-fork because the DAO was hacked. CanonChain realizes a community governance process through CZC to deal with a few problems that cannot be completely solved by software algorithms at present.

In CanonChain, the governance power belongs to block producers elected through community consensus (Section 3.1.3.1). Block producers have limited and supervised privileges to freeze accounts, update defective applications, and make changes to underlying protocols. Block producers must represent the wide range of interests of all accounts in CanonChain community, and communities can elect to replace block producers if they abuse power or refuse to vote on changes in community interests.

1.2.6.1 CanonChain parameters

After CZC establishes operation principles of CanonChain, the parameters of CanonChain are determined by community voting. If the parameters are approved by community, they are written into this version of the CZC. These parameters include:

• Transaction fees:

• Block time and maximum block size

• Methods to measure the contribution of devices

• The principle of generating candidates for block producers

• The number of block producers

• Allocation of block rewards

• Interface of interoperating with other blockchains or DeOS

1.2.6.2 CanonChain upgrade

CZC defines CanonChain update method as following:

1. Developers submit an update proposal and get more than three-fourths of the votes in favor and holds all the votes for seven consecutive days.

2. After step 1, all users must use the hash of the new CZC to confirm the transaction.

3. Developers submit modified source code to the test chain.

4. Developers need to continue holding more than three-fourths of the votes in support for seven consecutive days.

5. After the source code modification passes on test chain, all nodes need to finish upgrade within seven days. The nodes that have not upgraded to new code will be automatically excluded.

2 CanonChain scenarios

According to different delay and computation requirements, CanonChain application scenarios can be divided into several categories. Different category of scenarios put forward different challenges for the blockchain technology, but CanonChain provides corresponding solutions.

2.1 Platform application

We call the blockchain operating system class application model for computer chips such as Windows and Linux systems "platform applications".

Following the standard chain of fog networking protocols, any underlying blockchain operating system can run on standard chain systems such as Bitcoin, Ethereum, EOS, and so forth, if they are willing to follow the consensus of the standard chain or Interoperability agreement, then you can also operate in the standard chain and use, scheduling chain resources.

This type of platform application we call "DOS."

2.2 Interactive applications

We refer to applications with low latency and high reliability requirements as interactive applications, which include vehicular network, mesh network, medical diagnosis, security alarm, etc. Services for such applications need to be acknowledged within a short time period (possibly millisecond) that is much less than the block production time of current blockchain technology.

2.2.1 Vehicular network

In a vehicular network, vehicles exchange information such as location, speed, route, etc. collected by GPS, radar, camera and other sensors. In traditional implementation, a vehicle quickly transmits its own information to a cloud server through WAN connection. The cloud server analyzes the information reported by all the surrounding vehicles, and infer current

traffic conditions and scenes so to have an optimal scheduling of vehicles route and semaphores cycles.

In CanonChain, any peripheral devices can participate in the vehicular network to process information about the surrounding vehicles. For example, with CanonChain enabled, the street light can be a beacon to help vehicle positioning; all roadside restaurant cashier stations can be the computing center for road scene calculations; pedestrian handsets can send alerts to vehicle collision avoidance systems.

2.2.2 Mesh network

Mesh network can break the limitations of cellular network and build a low-cost next-generation wireless network, and it gets more and more attention on its performance such as bandwidth, self-organization and robustness. At present, it has been widely recognized by the industry as one of future directions of wireless network technology.

In mesh network, each node is connected to other nodes in a multi-hop manner. Any node can either be a client or be a relay. If a node itself connected to other nodes, it is a client. If a node help transmitting messages of others, it is the relay.

The physical layer (such as: Bluetooth, Wi-Fi, etc.) and network layer technology in mesh network are quite mature. However, how to give nodes incentive to share their own resources (processor, memory, battery, etc.) as a relay is still a problem.

If a mesh network is built on CanonChain, the consensus algorithm can be used to measure the contribution of the nodes and the block reward can be allocated proportionally to them, so that the mesh network is economically sustainable.

2.3 Big data applications

We refer to applications that require high computational power and high throughput as big data applications, including artificial intelligence, drug development, image rendering. The amount of data exchanged between service nodes in such applications far exceeds the block size of the current general blockchain technology.

2.3.1 Artificial intelligence training

Machine learning algorithms, especially neural networks, have become more and more popular in recent years. Neural networks have been applied to fields such as computer vision, speech recognition, natural language processing, stock market prediction, etc.

Neural networks are driven to emulate the work of a biological brain. Computers are given same kind of behavior as humans, sometimes even beyond human capabilities. Neural network could be as many as hundreds or even thousands of layers, known as deep learning network. Projects on deep learning networks require tremendous computational power for their deployment, training, and tuning. Modern PCs can train networks with tens of thousands of samples in a reasonable amount of time, supporting hundreds of dimensions of input data. Deep learning networks, however, require even larger sample of data, all currently implemented in more powerful GPUs.

CanonChain provides an economically efficient solution for implementing deep learning networks. By assigning the layers in the deep learning network appropriately into different nodes, only necessary parameters between layers are exchanged among nodes to speed up the learning process and improve the learning quality. In CanonChain, the data in the deep learning network is reasonably allocated according to the computing power of the device, and their corresponding rewards are provided.

3 CanonChain technology

3.1 PoP consensus algorithm

3.1.1 Node contribution

Device nodes can measure their own resources contribution to the chain, such as CPU, memory, bandwidth and storage space, and application attributes and record it in the form of a contract to the chain. Application attributes could indicate the urgency and importance of the transaction.

Based on these data, consensus algorithm will rank the contribution of nodes and pick out the candidates of block producer. Even if two nodes have the same resource contribution, different application attributes can cause two nodes to have different ranks on contribute.

Applications (T)	CPU (C)	Mem （M）	Bandwidth (B)	Storage(D)	Importance (I)
Vehicular network	low	low	low	low	high
Mesh network	low	low	middle	middle	middle
CDN	low	middle	high	high	low
Database query	middle	middle	middle	high	middle
Scientific computation	high	high	middle	high	low

Table 2 Comparison on contribution for different type of applications

In measuring the importance of applications, we consider the marginal effect. For example, in a CDN transaction, if a node has processed enough CDN services for a certain period, then the value of the same amount of resource will decrease with the marginal benefit, thus reducing the contribution of the node.

Let P denote device contribution as a function of CPU utilization (C), memory utilization (M), bandwidth (B), storage (D) and application importance (I). Take the block time in Canon Chain as unit time. For a given application s, suppose it starts at time t_0^s and the current time is t, then the contribution at current time is $P_s(t^s, t_0^s) = \gamma^{t-t_0} P(C, M, B, D, I)$, where γ is the discount factor considering the marginal value reduction. If a node serves more than one application at the same time, the contribution of the entire node at time t is the summation of contribution of all applications $P(t) = \sum_{\{s \in S\}} P_s(t, t_0^s)$.

The formula above is over simplified. In actual case, CZR considers more factors. Determining node contribution is a sophisticated dynamic process. We use one example in mesh network to illustrate this. The contribution of nodes could also depend on network topology. As shown in

Figure 4, there are two mesh networks and both have good connectivity. The nodes inside each mesh network are active and contribute constantly. However, there is only one node between these two mesh networks. If any node in mesh network 1 wants to talk to nodes in mesh network 2, the message has to go through the middle node. In this example, the physical resources contributed by the middle node may not as much as those inside two mesh networks, but CZR will rank it higher in terms of contribution because it is the necessary device to maintain the connectivity between the mesh networks. It should be noted that the nodes in mesh networks has mobility, so the importance of the nodes will change accordingly, which requires intelligent and dynamic measurement from CZR.

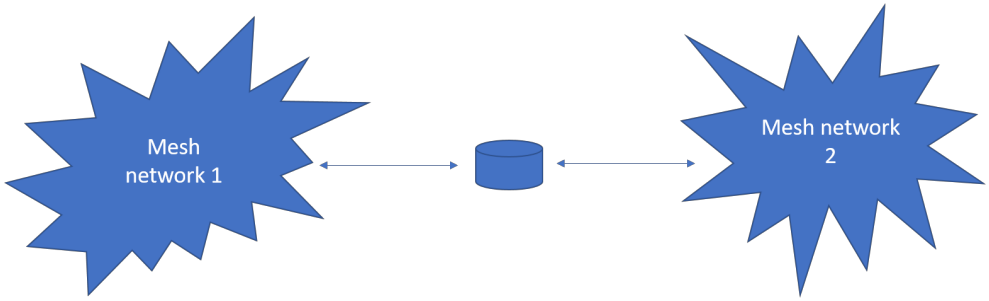


Figure 4. Node contribution relates to network topology

3.1.2 Design goals

Consensus algorithms evolve along with the development of the blockchain. Table 3 compares the advantages and disadvantages of the popular consensus algorithms and their features. Consensus algorithm of CanonChain is designed to be fast, irreversible, and fair to the devices in foggy network.

The original consensus algorithm was designed mainly for irreversibility. When one or more accountants made fake transactions on account book, the remaining accountants were still able to reach a consensus on the books. PoW (Proof of Work) algorithm used by Bitcoin and the first generation of Ethereum is an irreversible consensus algorithm. PoW uses competitive hashing to do accounting, with a large amount of energy being consumed in competition. Mining costs gets higher, and the speed is gradually limited. As a result, as the number of nodes involved in mining increases, the cost of maintaining a healthy ecosystem under the PoW protocol will continue to rise.

In order to deal with the issues in PoW, reduce accounting energy consumption and improve accounting speed, blockchain researchers have proposed two alternative solutions: Proof of Stake (PoS) and Proof of Importance (PoI). The PoS consensus algorithm uses the amount of assets to replace the amount of computation power in determine the probability of being an accountant. This algorithm solves the shortcomings of PoW, but enlarges the impact of capital on the allocation of probability of accounting, which makes it easier for big capital to form an oligopoly and loses the fairness of accounting. PoI consensus algorithms introduce the concept of account importance, using the account Importance score to assign the probability of accounting. However, the importance of accounts often lacks community consensus.

The DPoS (Delegated Proof of Stake) algorithm, is an improvement of PoS. The accountants are generated by the election from the community. The reduction in number of bookkeepers allows the consensus algorithm to run faster. The bookkeeper gets community recognition, making the entire accounting process more democratic and fairer.

Consensus Algorithm	How does it work?	Pros and Cons	Application
PoW	Competitive hash to determine accounting advantages	Pros: BFT， irreversible Cons: waste electricity, high cost and low speed	Bitcoin，Ethereum
PoS	Use the amount of asset to allocate the probability of obtaining accounting rights	Pros: low energy consumption, high speed, irreversible Cons： oligarchy, loss fairness	Ethereum Casper
DPoS	Selects a small group of accounts as a proxy for PoS	Pros: faster， democratic Cons: doesn't include importance of the accounts	Bitshare, EoS.io
PoI	Use account importance to allocate the probability of obtaining accounting rights	Pros: low cost, fast, fair Cons: lack consensus from community, account importance not necessarily matches device contribution	

Table 3 Popular consensus algorithm and their comparison

But for foggy network, the above several consensus algorithms have their own flaws. In PoI, the importance of the account is mainly judged by the activity of the account on the chain and the number of transactions, which cannot meet the requirement of fully measuring the contribution of devices off the chain. The accountant lacks community consensus. In DPoS, community consensus cannot completely reflect the contribution of devices due to huge volume of devices widely spreading in foggy network.

We propose the Proof of Participation (PoP) algorithm based on device participation. PoP combines the PoI and DPoS advantages to ensure fairness of devices and community consensus. Notice that current PoP algorithm is not designed to meet the ultimate goal of fog networking, and it is a consensus algorithm still based on current blockchain technology. CanonChain is a joint optimization of blockchain technology and fog networking needs. As shown in the previous example, device importance and account importance are very different in the foggy network. It is not the optimal solution to use only account importance to decide accounting rights. Therefore, CanonChain consensus algorithm will keep evolving as the project moving forward and will be adopted after consensus in community.

3.1.3 PoP algorithm design

3.1.3.1 Elect block producer

In PoP consensus mechanism, the system will first select a wide range of representative accounts as candidate accounts. When choosing candidate accounts, the system considers a number of factors: the geographical distribution of the accounts, type of applications the account is running; and contribution of devices associated with the account, etc. Candidate accounts are broadly representative and this approach is very close to the election of congresses. Each representative has the same voting power, and they are leaders in their respective regions and in their respective industries.

The community votes on the system-generated candidate accounts. According to the votes obtained, the system picks up a total of N block producer from the candidates, where N is determined by community voting and written into CZC (Section 1.2.4.1). The more votes a candidate account obtained, the greater the chance it is selected as a block producer. Therefore,

the final selection of the block producers has a wide range of representative, but also with community consensus. Voting through the community can eliminate accounts that, though make contributions, are less active in community construction or even vandalize the ecology of CanonChain.

3.1.3.2 Block generation

CanonChain generates blocks in every fixed T seconds, where T is voted by the community and written into CZC (Section 1.2.4.1). Let N-block production time be a cycle. Within N intervals of a cycle, the block producers generate blocks in round robin. If one of the block producers don't generate a block within the specified time interval, then this block will be skipped and the interval to the next block becomes $2T$ seconds.

In the next cycle, the order in which the block producers generate blocks is shuffled. This is to maintain good interoperability among the block producers and to avoid CanonChain branches develop into a fixed pattern with a small probability so they could never merge.

In the following cases, the community will vote again to elect new block producers:

• CZR is aware of major change in the status of devices and CZC submits new candidates to the community.

• Some block producers did not fulfill their duties and did not generate blocks for a long time.

3.1.3.3 Transaction confirmation

With 100% block producer participation, CanonChain does not fork and transactions can be confirmed in seconds on average. However, if there is a software bug, or the network is not smooth enough, or some blockchain producer maliciously cause the chain forked, a transaction needs confirmation from at least $(2/3 * N + 1)$ + 1 block producers to be irreversible.

3.2 Smart contract

A smart contract is more than just a computer program automatically executes, it is also a participant. It responds to received messages, it receives and stores values, and it can send messages and values as well. Smart contract is like a trustworthy person who can hold assets temporary and always operate in pre-specified rules. Shown in Figure 5, from left to right is an

ordinary smart contract model. A piece of code (a smart contract) is deployed on a shared, replicated ledger. The code maintains its own state, controls its own assets and responds to external messages or assets transfer.

On top of the ordinary smart contract design, on one hand, CanonChain introduces device event which can trigger contract execution or state change. On the other hand, a device can write callback functions to a contract. When the contract enters a certain state, the device is notified to take appropriate action.

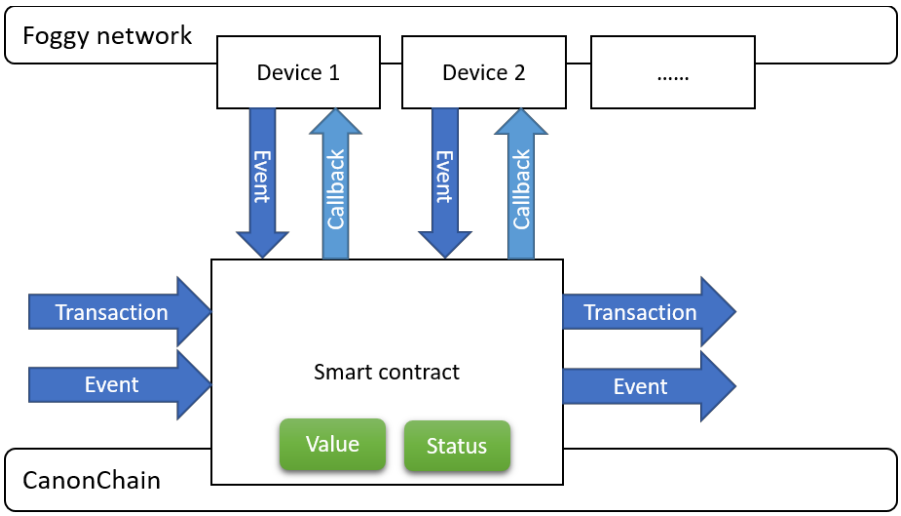


Figure 5 CanonChain smart contract introduces off-chain events

Smart contracts are value exchange channels between inside and outside of CanonChain. They are used for intermittent interactions between devices themselves or between devices and the chain. Take the following two contracts as examples:

Contract 1: When multiple devices cooperate to process a job, they sign a contract guaranteeing to contribute certain resources and put deposits into the contract. If a device wants to quit, it can send an advanced notice via an event, and the contract refund its deposit. In contrast, if the contract sees that one of the devices' contribution doesn't not meet the goal, it runs the callback in the device and confiscate the deposit.

Contract 2: When a CanonChain device runs a commercial promotion program, it signs contract with the promoter. Whenever a new user is developed on the device, a certain amount of asset is transferred from promoter's account to the device account.

3.3 Mechanism design on transactions

In order to maintain ecological health of a blockchain, a transaction sender needs to pay a certain amount of fee to the block producer who did confirm the transaction. The transaction fee is a compensation for the block producer's contribution to the chain. The account that initiates the transaction needs to specify the fee that it is willing to pay for the transaction, while a block producer each specify the minimum transaction fee willing to accept. Only transactions with fee higher than the minimum fee will be dealt with by a block producer. The block producers give priority to process transaction with higher fees. If a transaction fee is too low, the transaction may take a long time to be confirmed. Transaction fees are designed to encourage more efficient contract code, reduce unnecessary calculations, and prevent system attacks. After all, attackers pay a price for the resources they consume.

From economic point of view, transaction initiators purchase transaction confirmation services from block producers through auction. Therefore, we can guide the design of transaction rates by mechanism design in economics. The theory of mechanism design is to study designing a set of rules to achieve certain objectives under incomplete information and decentralized decision-making.

Figure 6 illustrates two different cases that a block producer deals with. A block can hold up to N transactions and the price of the trades received by the block producer from high to low is $p_1, p_2, \dots, p_n, \dots, p_N$. As shown in (a), the block producer sets the rate lower than the last transaction i.e. $p_m \leq p_N$, so that all transactions are packed into the block and the total transaction fee is $\sum_{\{i=1\}}^{N} p_i$. In (b), the block producer sets only a lower rate than the first n transactions, so that only first n transactions are packed into the block, and the total transaction fee is $\sum_{\{i=1\}}^{n} p_i$.

In such a setup, the fee that the transaction initiator pays to the block producer is the same as his bid price: if his bid is p, the transaction fee that must be paid to the block producer is p. This

approach is called "The first-price auction" in auction theory. Although the first price auction is simple and easy to understand, it's not incentive compatible i.e. the bids doesn't reflect bidders' true valuation of the item. Therefore, the final allocation of the items is not optimal from the social point of view.

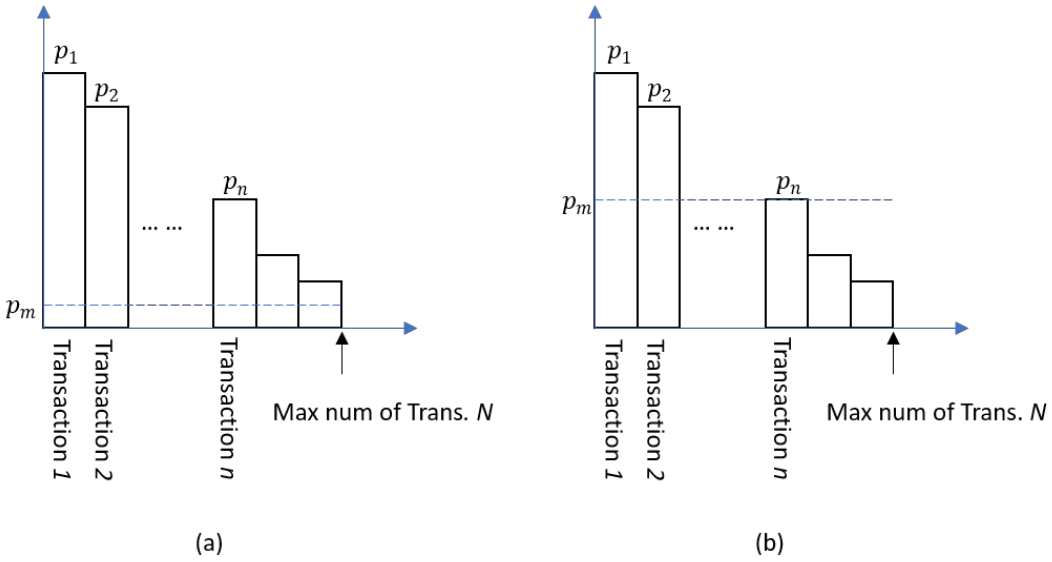


Figure 6 Transaction rate mechanism design

In mechanism design theory, incentive compatibility can be achieved by "The second-price auction". The transaction fee paid by the initiator to the block producer is not his bid p, but rather the highest bid that lose the auction. In Figure 6, in the second price auction, the final fee of all confirmed transactions is the price p_m proposed by the block producer. By introducing the second price auction in transaction rate design in CanonChain, the transaction initiator will bid according to their true valuation of the transaction. Therefore, the distribution of transaction in CanonChain is socially optimal.

3.4 Encryption and access control

On traditional blockchains, all transactions are recorded publicly. Any user can identify a transaction and parse it. This design ensures the openness and transparency of the transactions and greatly helps the financial industry. However, in CanonChain and foggy network applications, not all the data should be public. For privacy, a user's data on the chain should only belongs to himself. If anybody need to access the data, he needs to get permission from

the user. This is one of the motivations for us to change IoT from cloud based computing into a foggy network.

We designed new user access control for this purpose. In CanonChain devices, a user's data is encrypted and stored with his own key and cannot be access by others without authorization. If a user gives other users or a service provider authorization to access his data, the activity is stored publicly in Canon Chain and is irreversible.

Data owner grants access to his own data on CanonChain in main steps. Firstly, data owner sends a transaction containing the ID of the data stream (hash of the data digest) and register the data in CanonChain. Secondly, if the data owner wants to share the data authorization to others, he sends another transaction which includes the ID of the data stream and the public key address of who is shared.

When a device gets data request, it will first obtain authorization from the requestor's address in CanonChain. If the data owner does not register the authorization on CanonChain, the request is rejected; otherwise, the request is accepted.

3.5 Third-party tools

CanonChain is designed to handle heterogeneous applications in foggy network., especially for high-throughput and low-latency application. To achieve this, CanonChain also rely on 3rd party software and development tools.

3.5.1 Distributed File System IPFS

IPFS (Inter Planetary File System) is a permanent, decentralized method of saving and sharing files. It's a distributed protocol for content addressability, version tracking, and point-to-point hypermedia.

Content addressability: Use a unique hash value is generated from file contents to identify the file, instead of the location where the file is stored. Therefore, only one copy exists for files with same contents, which saves storage space.

Versioning tracking: Traceable file modification history

Peer to peer hypermedia: Use P2P protocol to store a wide variety of data types

IPFS can treated as a BitTorrent farm and accessed through the same Git repository. It combines the advantages of products such as distributed hash tables, BitTorrent, Git, self-certified file systems, etc.

There will be distributed files system on top of CanonChain based by improving IPFS open source program, adding security, decentralization and anti-loss features.

3.5.2 Distributed database NoSQL

To enable applications on CanonChain to access data using traditional distributed database approaches, CanonChain support the interface for distributed database such as NoSQL. Canon Chain have both the advantages of blockchain and distributed database technologies, so it can support business level scalability with high-speed parallel database read and write, while recording the user behavior in the chain to maintain the irreversibility of the database data.

Requirement	Blockchain	Distributed Database
High throughput	-	√
Low latency	-	√
Quick inquiry	-	√
Diverse authority	-	√
Distributed control	√	-
Irreversible	√	-
Asset transfer	√	-

Table 4 Comparison of blockchain database and distributed database

3.5.3 Lightning network

Lightning network is initially designed for blockchain technology to adapt to massive micro-payment scenarios. By building a micro channel for both parties, a large number of payments can be instantaneously completed. When the transactions in the micro channel need to be settled, the result is submitted to the blockchain for confirmation. In theory, lightning network

solve the scalability issue of a blockchain network and can support tens of thousands of transactions per second. In Bitcoin and Ethereum, there are already concept proof implementation of lightning network.

Not only in the financial sector, in the field of fog computation there are also numerous micro-trading scenarios. As mentioned before, in vehicular network application, several adjacent cars need to quickly exchange information with each other to determine the surrounding environment with a very short delay to enable functional safety. In the meantime, DeOSes running on CanonChain also require lightning networks to support such fragmented transactions due to large amount of cross-chain transactions.

Therefore, CaononChain includes lightning network as one of fundamental components in blockchain infrastructure and provides ample flexibility on design. A third-party developer can use lightning network service on CanonChain to support high-frequency transaction scenarios.

3.5.4 Developer Tools

To support developers in the community, the CanonChain development team will supply a wealth of development tools, including smart contract development IDE, block browser, all kinds of popular IDE plug-ins, debuggers, simulators, smart contract validation tools, mobile SDK, etc. At the same time, there will be plenty of lectures and panel discussions in the CanonChain to promote the developer tools.

3.6 Open source hardware

In CanonChain and foggy network, because of its complete openness, data security is particularly important. The data security on the chain can be guaranteed by the consensus algorithm, while the data security off the chain must be ensured by hardware redesign and encryption algorithm.

A hardware device that can be used safely in CanonChain must include: 1) secure P2P communication; 2) secure electronic wallet storage space; 3) an intrusion detection system; and 4) tampering proof. The current hardware products on market obviously cannot meet this requirement. For example, in a smartphone that CanonChain team is developing, a security

sandbox is appended to ordinary smart phone hardware. CanonChain user's wallet is stored in the sandbox.

CanonChain team has many years of communication protocols and chip design, intelligent gateways, smart routers and other hardware development experience. In the ecosystem building of CanonChain, we will take advantage of our strengths to rapidly develop intelligent hardware that supports CanonChain protocol and contributes to the community. After a period of testing, we open-source the hardware as a reference design to allow third-party developers to customize their CanonChain hardware.

4 Team

4.1 Core team members

Sichao Yang

CanonChain co-founder and core team member

Dr. Yang got his Ph.D in Department of Electrical and Computer Engineering and M.S. in Department of Mathematics both from University of Illinois at Urbana-Champaign. His Ph.D. research is on resource allocation and optimization in distributed networks.

Dr. Yang was a senior staff engineer in Qualcomm Inc. (NASDAQ: QCOM), the world's leading wireless communications chip and service provider. He was one of the key contributors to the design and development of the 4th generation mobile communications LTE chip. He also served as a technical lead in Qualcomm's research on vehicular network and autonomous driving, and holds many invention patents.

Dr. Yang has publications on top journals and his work has been cited by world renowned scholar worldwide including MIT, Stanford University, and University of Cambridge, etc. (Google scholar shows the current total number of citations is over 500); he was a reviewer for *Mathematics of Operations Research, IEEE Transaction on Networking, Games and Economics Behavior and scientific* and other magazines.

He was an adjunct professor at Rutgers University.

Hailong Jin

CanonChain co-founder and core team member

Founder of the first third-party payment platform in China;

Member of Payment Industry Expert Committee in Chinese E-Commerce Association;

Found of *Universal Search* keyword precise marketing software, which has been installed on more than 30 million PCs and with more than 6 million concurrent users.

Founder of sodao.com, the nation's largest beauty network with annual online sales over 500 million yuan.

Founder of WiFiSONG.com, the leader in design of intelligent hardware including Wi-Fi router, iBeacon and other edge computing devices.

Joe Thong

CanonChain co-founder

Malaysian, MBA

Co-founder of Popify, Twinova, and Raytheon Capital

Business partner with Samsung, SAP and other Fortune 500 company

Zhihao Shan

CanonChain co-founder

Former legal unit leader in network security bureau in Hangzhou, China

China's first Internet lawmaker

Researchers and developer on cryptography technology

4.2 Advisory team

Michael Saunders

Chief Mathematician

Research Professor

Dept of Management Science and Engineering (MS&E)

Stanford University

1985: William Orchard-Hays Prize in Computational Mathematical Programming, Mathematical Programming Society, first recipient

2004: ISI Highly Cited Researcher in Computer Science

2007: ISI Highly Cited Researcher in Mathematics

2007: Honorary Fellow of the Royal Society of New Zealand

2012: SIAM Linear Algebra Prize (with S.-C. Choi and C. C. Paige)

2012: Stanford University Invention Hall of Fame (with P. E. Gill, W. Murray, B. A. Murtagh, and M. H. Wright)

2013: SIAM Fellow

Yihao Liu

Chief Financial Advisor

Noah Holdings (NYSE: NOAH) Strategic Fund Partners, CEO of Noah Fortune Hong Kong

Former chief representative of the New York Stock Exchange in Beijing O. Assistant Alibaba (NYSE: BABA) listed on the NYSE during his tenure

Former chief operating officer of Merrill Lynch (NYSE: MER, TYO: 8675)

Yuefei Pan

Founder of zinc financial

Former Cheetah Mobile global content director and Sohu technology editor.

5 CanonChain roadmap

January 2018 ERC20 tokens issued, CZR listed on exchange

2018Q2 PoP consensus mechanism complete. CanonChain code open source and the public chain goes online

2018Q2 Release world's first blockchain gateway to support PoP Consensus algorithm

2018Q2 Release CanonChain API

2018Q3 CanonChain community open to global developers and fog computation alliance founded

2018Q4 Launch CanonChain Smartphone and start pre-sale

2019 Release foggy network protocol v1.0 on CanonChain

2020 Release first chip for foggy network protocol and make it open source

2021 ~ Low-cost micro-power adapters with foggy network specifications to appear, the world is fully connected for fog computation

6 CanonChain token: CZR

6.1 CZR

In CanonChain, all contributions will be CZR rewarded. Meanwhile, all consumption of resources need CZR as a payment

CZR is allocated by consensus algorithm based on device contribution

CZR is a measure of value transfer. It's also the fundamental asset for other DeOSes and Dapps running on CanonChain

6.2 CZR allocation plan

The total CZR is 1,618,033,988€. Among them:

For early investors and private placement 40%

Founding team, development team 10%

Community Incentives, Global Promotion, Cooperative Incentives Reserved 20%

POP incentive reserve (i.e., "pool" in traditional blockchain projects) 30%

6.3 CZR release mode

CZR is first issued by Ethereum in the form of ERC20 Token, and change to the token on the main chain after it gets online

Users can get CZR by private investment, purchase from exchange and personal transfer

CZR conversion rate during private investment: 1ETH = 6000CZR

6.4 CZR contract in the team

CZR allocated to the founding and development team is vested gradually in three years after initial release. The rules are as follows:

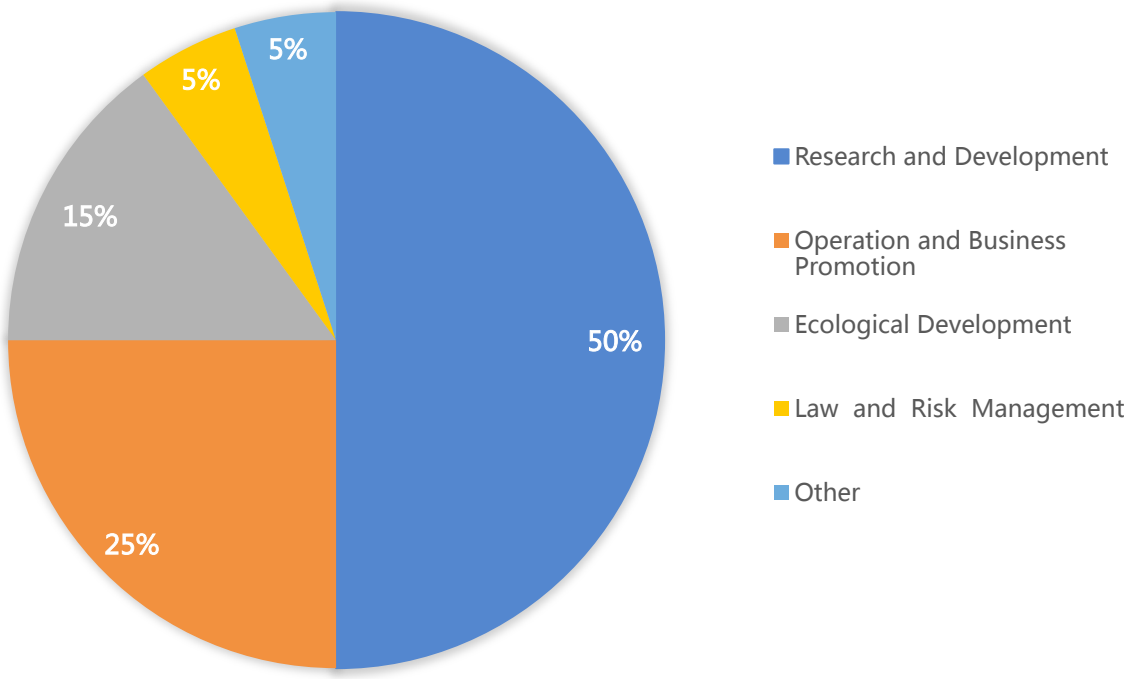
Vest immediately after release: 25%

Vest after one year: 25%

Vest after two years: 25%

Vest after three years: 25%

6.5 Funds from private investment



7 Risk disclaimer

1. This whitepaper is for the sole purpose of conveying information to specific audiences who are actively seeking information about the project and it does not constitute any future investment guidance, nor any form of contract or commitment.
2. Once a participant involves in the TOKEN distribution plan, it means that he / she understands and accepts the risk of the project. He/ she is willing to take personal responsibility for all the corresponding consequences.
3. The project team has made it clear that no guarantee for any return, nor assume any direct or indirect losses caused by the project.
4. The TOKEN involved in this project is an encrypted digital code used in the trading session and does not represent the project equity, right of income or control rights.
5. Due to the uncertainties inherent in the digital currency (including but not limited to the macro-environment for digital currency regulation in various countries, the industry competition and the technical vulnerabilities in the digital currency itself), we cannot assure that the project will be successful and the project will have a certain risk of failure. The

TOKEN in this project also has the risk of zeroing. Although the team will work hard to solve possible problems that may be encountered during the project, there will still be policy uncertainties in the future. Before supporting the project, you must ensure that all aspects of the blockchain are understood and make rational participation with adequate understanding of the risks.