Dora

Guo Xionghui
steve@dora.network

Tyler Kot
tyler@dora.network

Li Xing
star@dora.network

Version 1.1.2
December 9, 2018

Abstract

With the emergence of the new generation of smart contract platform, such as Ethereum, the application of Blockchain technology is greatly promoted, but these platforms can't meet the wider application requirements due to the lack of performance. Dora solves this problem from 3 aspects: vertical expansion, horizontal expansion and consensus algorithm.Vertical expansion works on the parallelism between contracts and contract itself; horizontal expansion improves performance using the technology of Hub-Zone chain. The consensus algorithm tries to find the balance between decentralization and performance based on DPoS, VRF, and BFT. In order to facilitate the migration of existing applications, Dora's virtual machine is fully compatible with EVM. At the same time, Dora can promote the development of the upper-layer applications by free transaction fees. In general, Dora[1] is a highly parallelized, high performance public chain without transaction fees.

1 The Current Landscape and Prospect of Blockchain

1.1 Review of Blockchain Technology

In September 2008, Satoshi Nakamoto published the whitepaper[1] of BitCoin. In January 2009, the main chain of Bitcoin was formally launched, and it was running safely, creating a new era of encrypt digital asset and pulling the new technology into the public view. In July 2015, the Ethereum[2] was launched. EVM of Ethereum could execute Turing-complete smart contract, it marks the second generation of Blockchain technology. Among these massive applications of Ethereum, encrypt digital token is undoubtedly the most important and popular one. The ERC-20 standard is introduced in the Ethereum community, and the ERC-721 standard is proposed, which builds up the foundation for the issuance and circulation of cryptocurrency, token and non-fungible token. In 2017, the scale of cryptocurrency markets increased by more than fifty times, and the issuance and appreciation of thousands of new cryptocurrencies and Tokens were the main factors. At present, the innovative technology of Blockchain is mainly embodied in the consensus mechanism, block structure and network structure.

The first is the consensus mechanism. PoW consensus algorithm is adopted in Bitcoin at the very beginning. PoW requires a large number of miners to perform hash operations to compete for block accounting, which will lead to a large amount of electricity consumption. In order to overcome this shortcoming of PoW, the new PoS consensus algorithm is proposed, which determines the accounting right of the block based on both the number of the users holding the token and the time of the user's possession of the token, which greatly reduces the power consumed due to the competition for the right to account and improves the efficiency. Both PoW and PoS need miners as well to fight for the right to account. Delegated Proof of Stake (DPoS) refers to the company's operating mechanism, and selects a limited number of agent accounting nodes by the decentralization of voting. These selected agency nodes take turns to generate the blocks according to the rules to avoid the competition to account, thus further improves the efficiency. Regarding these three

[1]Invested by the Loopring foundation,an incubator project from Hongkong Loopnest accelerator

consensus algorithms, theoretically, as long as no more than 50% of the nodes are erroneous, the whole network is safe. All of them belong to the algorithm family of indirect consensus, which first must compete for the right to account, then generate blocks, and finally solve the fork problem through a deterministic method (such as the maximum difficulty). In the most of Consortium blockchains, BFT algorithm is adopted, which belongs to the direct consensus algorithm. This algorithm can form a consensus immediately among the participants after finishing a round of operation, and does not have to compete for the right of account in advance and has no forks either, however, it can guarantee the whole network is safe only if no more than 1/3 accounting nodes in the network are erroneous.

The next is the organization structure of the block. A tree structure is adopted in the traditional Blockchain. In such case, a block has one and only one parent block, a global ordered linear ledger of all blocks is formed through the paternity relationship between blocks. In the latest research, multiple parent blocks are allowed, such as the IOTA[3] , it mandatorily requires that the new block must point to two parent blocks, and then organize into a DAG, which stipulates a linear ledger with the global order of multiple transactions as a nonlinear ledger that records only partial order relations. Thus the confirmation speed of transaction is improved.

The last is the network structure. The large network is sharded into small networks, and the structure of the Hub-Zone chain is adopted. For example, Ardor[4], Cosmos[5], Asch[6] and PChain[7]. In Cosmos, main chain is called as Hub, and the other sub chains are Zone. Interaction between Hub and Zone is done via IBC(Inter Blockchain Communication) protocol. When a currency is transferred from one Zone to another Zone through Hub, Hub is responsible for maintaining the invariance of its total amount, but Hub is not responsible for verifying the transactions on a single Zone, the Tendermint[8] consensus algorithms is used on both Hub and Zone. PChain is also similar to Cosmos and tries to form a sub chain according to each dApp, but the consensus algorithm is modified, PoS is adopted for both main chain and sub chain. Because there is no record and inspection mechanism for the transaction of the parent chain to the sub chain, the security of the subchain is fully maintained by the miners of the subchain, and it will be a potential security problem in the case of less subchain miners. Considering about this risk, a new miner role is designed in Asch and Ardor to generate and transfer the block records from the subchain to the parent chain. In the design of Asch, it requires the creator of the subchain to upload subchain blocks, while the design of the Ardor does not do the mandatory requirements of the role. By comparison the Ardor is more considerate. After a period of time, snapshots and clipping of the sub chain block data can be done by parent chain to solve its data expansion problem.

1.2 Two Problems to be Solved in the Current Blockchain: Performance and Storage

With the increase of demand, a very urgent problem faced by Blockchain is scalability. At present, the low throughput capacity of Blockchain (about 7 transactions per second in Bitcoin, about 15 transactions per second in Ethereum) is not enough to meet the needs of global financial transactions. By comparison, Visa can handle 56000 TPS, while Alipay has achieved peak 200000+TPS transactions in November 2017. Currently, the scalability problem based on Blockchain network has brought great limitations to its applications. For example, since the smart contract function is supported by the Ethereum, many projects run directly on Ethereum as dApp, but with more and more projects running on the chain, the load becomes increasingly heavier, and just one project like "Cryptokitties" made the transfer function difficult to succeed in the entire Ethereum. How to improve Blockchain throughput without affecting security and decentralization is still to be explored.

As time goes on, another problem faced by Blockchain is data explosion. As shown in figure 1 (data source, http://bc.daniel.net.nz), By the end of Feb. 2018, the current total block size is more than 150G in Bitcoin, and the new node needs to take 14 days to synchronize with all the blocks; in Ethereum, the size of the block is more than 650G, and a new node needs to take 8 days to synchronize with all the blocks, and is still increasing by 145M per day.

1.3 The Prospect of Dora

Dora introduces a brand new Blockchain architecture designed to meet the needs of global business activities with Blockchain technology. Inspired by the pipeline model and branch prediction systems that are currently applied in many

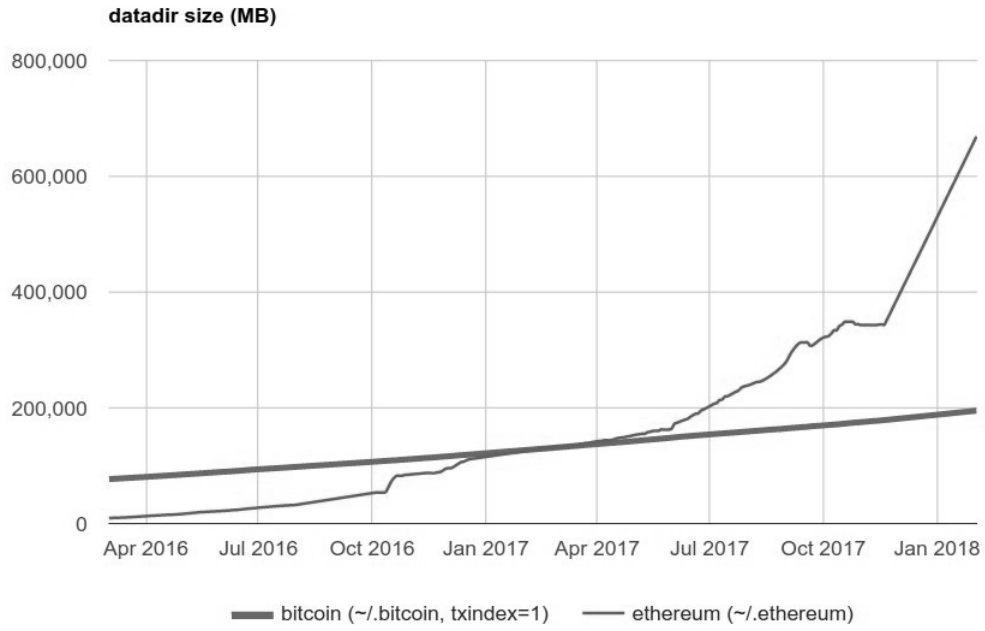

Figure 1: Blockchain Data Explosion

applications in CPU, our core members are trying to apply these technologies and ideas into Blockchain area, creating a solution dedicated for the scalability problem of Blockchain. The solution can substantially increase the applicable scope of the Blockchain while optimizing its security and decentralization characteristics.

Dora will push the blockchain technology into the next generation, and create a network that can add on new nodes quickly by node-snapshot. Ultimately, the goal of Dora is to become a public chain that everyone can use and everyone can use easily.

2 Dora's Vertical Expansion: Parallelization

There are two types of expansion solution in traditional Blockchain: state channel and Multi Chain sharding. Lightning network [9] deploys state channel technology to alleviate the scalability problem of Blockchain. The basic idea is that for the frequent transactions between a fixed group of parties, once all parties complete the transactions, one of them will just publish the final result without having to generate multiple transaction records on the chain (essentially, it decreases the storage of the intermediate results). However, the lightning network is only suitable for frequent transactions among a fixed group of the parties, and if the transaction goals of the user are random and the transaction occurs occasionally, then it will lead to poor efficiency. Multi chain sharding is a horizontal expansion technology, it increases the number of chains or segments, so that transactions are dispersed to achieve final expansion, but this is usually accompanied by security problems on subchains or subsections.

Serialization mechanism is taken as the assumption by all of these solutions when in transaction, because the serialization of the transaction enables each miner node to perform and verify independently. The same output result obviously can be obtained after the same serial operation starting from the same genesis block.

The TPS of the system is always constrained by the performance of a single node definitely because the serialization of transactions. However, we need to think about it carefully, is serial execution the only option for transactions? Can we do parallel processing at the transaction level in order to design a vertical expansion technology?

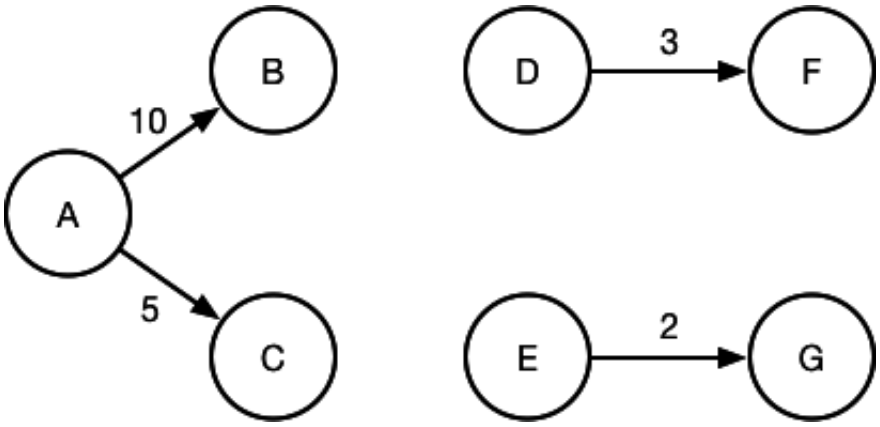

Figure 2: Common Transfer Transation

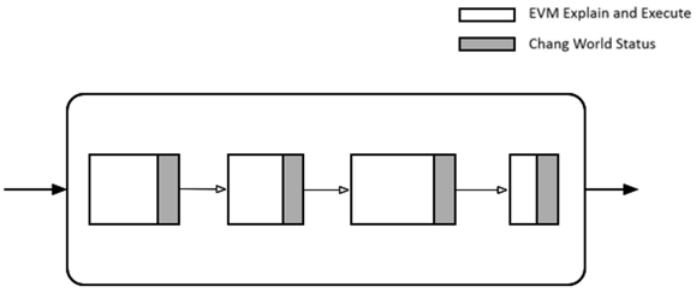

Figure 3: Serialization of Smart Contract

2.1 Parallelization of Common Transfer transactions

The account is regarded as a node. When transaction occurs from account A to account B, an arrow is added between node A and node B, the number next to the arrow indicates the transaction amount. Thus, a list of transactions within a block can form a diagram. Let's look at an example, assuming that a block contains the following transactions. {A transfers 10 tokens to B, and transfers 5 tokens to C, D transfers 3 tokens to F, E transfers 2 tokens to G }, As shown in figure 2.

From the diagram, it is obvious that the whole transaction set is divided into three sub connected graphs. {A,B,C}, {D,F}{E,G}. Tansactions in the same connected graph can only be executed sequentially because of dependencies. But in fact, transactions in different connected graphs can be executed in parallel. For example, the execution sequence between D to F and E to G is different only in the middle temporary state, but it does not affect the state that the block eventually reaches, and in the real Blockchain world, we can easily find a set of transactions without dependency.

2.2 Parallelization of Smart Contracts

For common transfer transactions, it is easy to divide the subconnected graph according to the account involved, but what if a transaction is a call to a smart contract? First let's take a look at the example shown in the figure3 how smart contract runs on chain. The execution logic of a smart contract can be considered as two parts: the first part is EVM explanation and execution marked by white, and the second part is what truly change the world state and marked by gray. The grey part is the key component which influence the final world status. A miner will collect the world state of the external account that the smart contract relies on during an execution process of the smart contract, and can also collect the world state of the external account that is affected ultimately.

The two smart contracts are considered as connected as long as there are intersections in account sets which rely on

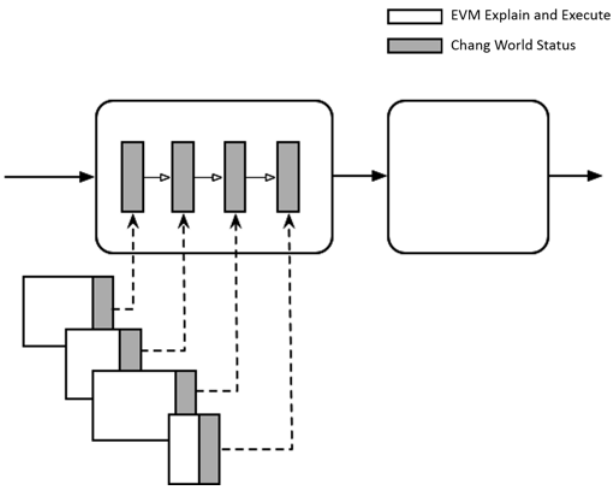

Figure 4: Parallelization of Smart Contract

and change by these two smart contracts. In this way, all the smart contract transactions in a block can be divided into different connected subgraphs, and each subgraph can be executed in parallel. As shown in figure 4, Assuming that the calls of these four smart contracts are all in different connected subgraphs, once the parallelization is executed, the generation speed of the block can be greatly enhanced, thus improving the TPS of the entire system. This parallelization can be achieved either by single machine multithread or by parallel process with multiple machines.

By analyzing the block data between block height 5592867 and 5609843 of Ethereum, the following conclusions can be drawn: 53.7% of the total transaction volume are smart contract transactions. The average execution time of each smart contract transaction is 1.29ms, the maximum execution time is 51.1ms, and the shortest is 0.14ms. That is to say, if the smart contract call is executed serially, the maximum TPS of the system can only reach 775. When executed in parallel, theoretically, there is no limitation.

The vertical expansion technology is essentially similar to the branch prediction in the CPU system, that is, the branch is executed in advance by pipelining. If the prediction is correct, then the subsequent pipeline only needs to be executed in sequence, thus greatly improving the parallelism of the system. Applying this idea in the Blockchain system is proposed by Dora. Theoretically, this idea can be applied to other public chains, thus increasing the TPS of the whole system.

A formal definition of the algorithm is given below.

2.2.1 Symbol Definition

Definition 2.1. *First, let's define some of the concepts which will be used later.*

Space of Smart Contract: $\mathbb{T}$

Block: $B_t = \{T_{1..}T_{n_t}\} \in 2^{\mathbb{T}}$*, it is a set of smart contracts.*

Address Space of Account: $\mathbb{A}$

Status Space of Account: $\mathbb{L}$

Space of World Status: $\mathbb{L}^{\mathbb{A}}$*, It is a set of mapping from account address space to state space.*

Function of World State Transition: $\Gamma : \mathbb{L}^{\mathbb{A}} \times 2^{\mathbb{T}} \to \mathbb{L}^{\mathbb{A}}$*,e.g.* $\sigma_t = \Gamma(\sigma_{t-1}, B_{t-1})$*and* $\sigma_t, \sigma_{t-1} \in \mathbb{L}^{\mathbb{A}}$*It implements a set of smart contracts based on the current world state and updates the world status.*

Serial Execution: For any pair of contracts in a group of contract, if all operations of one of the contracts are prior to another one, then the execution of this pair of contracts is called serial execution. i.e, the atomic serialization.

Security: For a parallel execution of a pair of contracts, if the change result of the world status is equivalent to the result when in some of the serial execution, it is said that the parallel execution is secure. i.e, given the parallel state transition function Γ_p*and the set* $\mathbb{S}$*of all serial transition functions, it is necessary to meet* $(\exists S)(S \in \mathbb{S}) \wedge (S(\boldsymbol{\sigma}, B_{t-1}) = \Gamma_p(\boldsymbol{\sigma}, B_{t-1})$

Set of reading account of smart contract:$T_r \in 2^{\mathbb{A}}$
Set of writing account of smart contract:$T_w \in 2^{\mathbb{A}}$
Status of account:$L(T_r) \in 2^{\mathbb{L}}$

2.2.2 Algorithm Flow

The following is the flow of the SSP algorithm 1, which can improve the performance of the system and reduce the execution time on the premise of ensuring security. It should be noted that common transfer can be regarded as a simplified smart contract, so SSP is also applicable.

Algorithm 1: Safe Speculative Parallelization(SSP)

Procedure *proposer*.`Propose()`
  repeat
    Receive $T \in \mathbb{T}$;
    Execute smart contract $\sigma = \Gamma((T_r, L(T_r)), T)$;
    Send the tuple (σ, T_r, T_w) to validator;
  until *exit*;
  return;
Procedure *validator*.`Validate()`
  Set of valid contracts $\mathbb{V} \leftarrow \varnothing$;
  Delta world state $\Delta\sigma \leftarrow$ null;
  repeat
    Receive the tuple (σ, T_r, T_w) from proposer;
    for *each valid contract* $V \in \mathbb{V}$ **do**
      if $V_r \cap T_w \neq \varnothing$ *or* $V_w \cap T_w \neq \varnothing$ *or* $V_w \cap T_r \neq \varnothing$ **then**
        Abort T;
      end
    end
    if T *is not aborted* **then**
      $\mathbb{V} \leftarrow \mathbb{V} \cup T$;
      $\Delta\sigma \leftarrow \Delta\sigma + \sigma$;
    end
  until *timeout or the* $|\mathbb{V}| > n$;
  Update world state $\sigma_{t+1} \leftarrow \sigma_t + \Delta\sigma$;
  return;

2.3 Instruction Level Parallelism in Smart Contracts

Dora is fully compatible with the Ethereum EVM instruction set and performs performance optimization at the implementation level. First, Dora performs optimization when compiling, including loop unrolling, instruction merging, etc. Second, Dora performs optimization when in runtime, including JIT and so on. At the same time, it allows programmers to mark the code (Annotation) to communicate with compiler which parts can be optimized in parallel.

2.4 Dynamic adjustment of block size and Block Compaction

In order to improve the TPS of the entire network, Dora will adjust the size of the block dynamically according to the current network congestion, and we will also introduce a block compaction technology, As shown in figure 5:

In a traditional Blockchain, a block usually consists of a block header and a series of transaction lists. E.g, in Ethereum, a transaction takes about more than 200 bytes. When a node broadcasts a block, it will pack the original transactions into the block and send it out, which is a complete waste of network bandwidth resources, since each original transaction has been broadcast to every node of the network before that, in theory we can use a compression mode, that is, put only

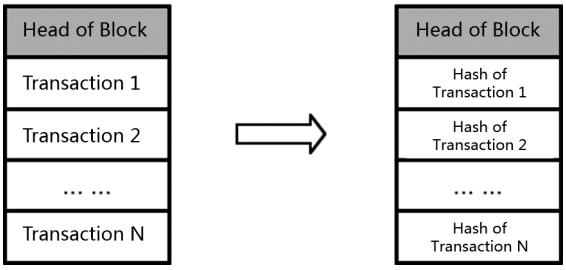

Figure 5: Block Compaction

the Hash value of the transaction in the block (only 32 bytes). When the node receives the Hash list of block transactions, only if the corresponding transaction data is not found locally, then go to other nodes to ask for the data. In this way, the bandwidth of the network can be saved about 7 times, which is almost as much as that of TPS.

3 Dora Horizontal Expansion: Hub-Zone Chain

Dora also supports horizontal expansion of the Blockchain in the way of Hub-Zone chains (Figure 6). A smart contract can be posted on either the Hub or Zone chain. A Zone chain can be dedicated to a smart contract which requires high computing performance.

The Hub-Zone chain uses an account model, account can be shared on both Hub and Zone chain. However, for the same account, the world status will be recorded on the different chains separately. E.g., a user can create an account with a wallet, which can be used to receive and send the transfer transactions of the Hub chain and Zone chain, but the number of tokens owned by the Hub chain is recorded on the Hub chain, Zone chain is same as well. At the same time, a special cross chain transaction(CrossChainTransaction) is supported to transfer tokens between Hub chain and Zone chain. The Hub chain supports multiple token transfers.

3.1 Zone Chain Security

As shown in figure 7, the Dora master chain supports a special transaction. The miners that generate blocks in the Zone chain are responsible for packing the sub block information (ChildChainBlock) that have not been recorded on the master chain and transferring it to the master chain, but it should be noted that the complete Zone block info will not be recorded on the Hub chain, only head info of Zone block will be recorded. These data are only records. The miners of the Hub chain are not responsible for verification when they perform packaging. For the data security of the zone chain, Dora Network requires that the miner of zone chain to lock up certain amount of DNT tokens, at the same time introduces a new supervisor role. Once the Supervisor discovers and verifies that the miner for the Zone chain has created a faulty block, he will receive certain amount of reward.

The supervisor will monitor all transactions of the Hub chain and Zone chain simultaneously. When a Zone chain block package message(ChildChainBlock) is detected, it will perform verification referring to the transaction data previously received from Zone chain. Once the problem is found for Zone chain, the evidence will be submitted to the Hub chain, and the Hub chain miners will check the evidence. In the meantime, to prevent a DOS attack by the supervisors, certain amount of DNT tokens must be collected from the supervisors as collaterals when submitting evidence. Once the evidence are verified, the supervisors will receive all the collateral tokens from the zone chain miners who made the mistakes; if the evidence could not be verified, the supervisors collateral tokens for this round will be awarded to the miner of the Hub chain.

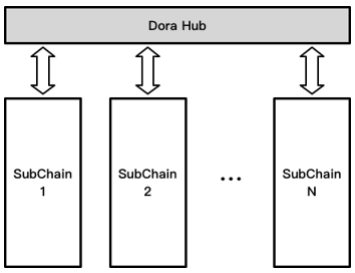

Figure 6: Hub-Zone Chain

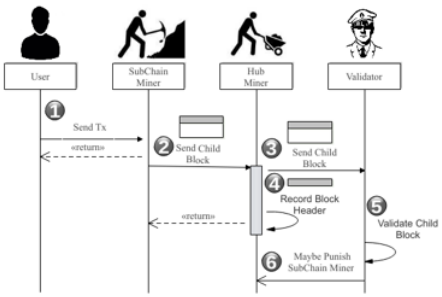

Figure 7: Zone Chain Security

3.2 Data Clipping and Snapshot

To solve the problem of data explosion, the Dora Hub chain supports the clipping of the Zone chain block information, and the data recorded on the Hub chain only retain the latest block data(proposed 2048 blocks in the early stage) of Zone chain , and the data prior to that can be removed from the Hub chain.

The Hub chain also supports the snapshot, and a snapshot contains the world status information of all the accounts of the current block, so that a new node can be synchronized quickly from a snapshot block, and even a secondary node can delete all the block data before the snapshot.

3.3 Cross Chain Token Transaction Between Hub and Zone Chain in Dora

Generally, a token will circulate only within a particular subchain, but Dora designs a special CrossChainTransaction that allows tokens in the same account to be transferred to each other between the Zone Chain and the Hub chain.

CrossChainTransaction: {
blockno: block number, rawtransaction in which block in which specific block of this transaction
blockhash:Block Hash corresponding to blockno
merklepath:Merkel tree path to verify if the rawtransaction is in block(blockno).
rawtransaction:{
owner: ""0xfbc2a4...ed"",
symbol: UT,
value: 100,
direction: 0 or 1 0 for from Zone Chain to Hub Chain 1 for from Hub Chain to Zone Chain
}
}

As shown in figure 8, how the token UT is transferred from the Zone chain to the Hub chain.

Suppose that account A transfers 100 UT to the Hub chain, when cross chain transactions occur:

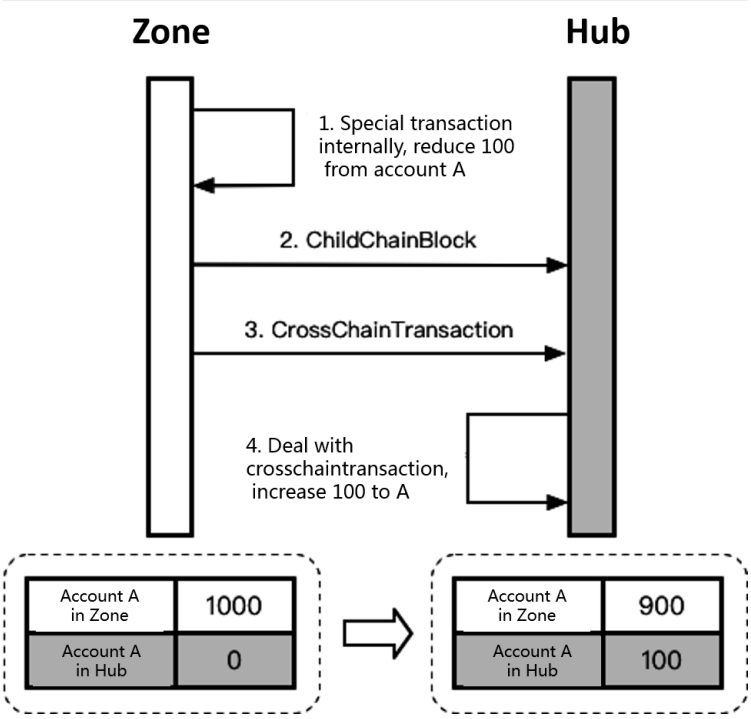

Figure 8: Token transferring between Hub Chain and Zone Chain

1.First A launches a special transfer transaction on the UT Zone chain, which requesting to transfer the 100 UT from the Zone account A to the Hub account A.

2.When the transaction is confirmed on the Zone chain, then awaiting miner of the Zone chain records the block information containing the transaction into the Hub chain.

3.Then on the Hub chain, A initiates a CrossChainTransaction from the Zone chain to the Hub chain, with transfer transaction in step 1, the block height and the Merkle path for Hub chain to verify;

4.The miner of the Hub chain verifies the CrossChainTransaction. If verified, it will be packaged into the Hub chain block, and the state record of the Hub chain account A is updated to transferred in 100 UT.

On the contrary, the token can be transferred back to the Zone chain from the Hub chain, the advantage of which is that the tokens are retained in the same account during the entire transfer process and do not require additional controlled accounts to be introduced as other cross chain solutions.,

At the same time, a special TokenSwapAction is designed on the Dora Hub chain, which allows to perform atomic exchange between two types of tokens, and the signature information of the two parties is required to carry, and determine whether the exchange can be performed after checks the balance of the two parties.

TokenSwapAction: {
taker: "LRC",
takerAddr: 0xfbc2a4...ed
takerValue: 1000,
maker: "RDN",
makerAddr: 0xabc35e...ef
makerValue: 500,
takerOrderWithSignature
makerOrderWithSignature
}

Through this atomic token exchange TokenSwapAction and cross chain operation CrossChainTransaction, it is easy

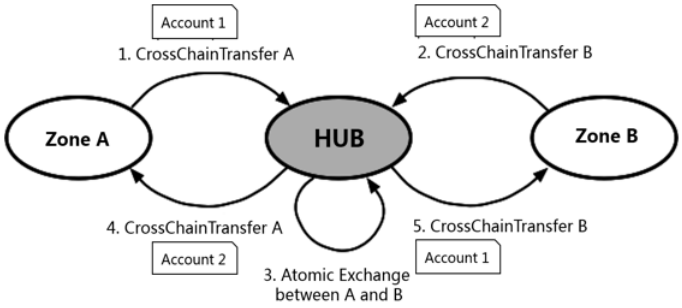

Figure 9: CrossChain Transaction

to implement cross chain transactions between the Hub chains and Zone chain (Figure 9). Dora will create some special bridging sub chains which is able to insert other existing public chains, so that the cross chain transactions of existing chains can be supported as well.

4 Consensus Algorithm DVBC

4.1 The Idea of Algorithm

Considering the balance between high performance and high security requirements, Dora proposed a hybrid consensus algorithm DVBC (Delegate Verifiable BFT Consensus). DVBC is based on DPoS[10], VRF[11] and BFT[8]. Assuming that there are Mnodes, $N+1$nodes should be selected in total, and the initial value of N is 21. At the beginning of each round of consensus, Dora first uses the first layer algorithm DPoS to select the number of top $N/3$ nodes from the number of M nodes according to the size of the vote, and these nodes will automatically become the bookkeeper of the current round. In second layer algorithm, the remaining nodes $M - N/3$ are selected as candidates. In these nodes, the VRF algorithm is utilized, the 2*N/3+1 candidate nodes are selected randomly as the bookkeepers. In the last layer, we use the BFT[8] algorithm to reach the consensus of the blocks among these selectd $N+1$ bookkeepers. This ensures that as long as there are no more than $N/3$ fraud nodes in the $N+1$ nodes, the correctness of the blocks is guaranteed and there is no fork at all. Obviously, this algorithm can prevent super-nodes from defrauding jointly. Additionally, because the candidate nodes also have the probability to be selected as the bookkeeper, this economic incentive mechanism motivates more nodes to be selected as candidate nodes, thereby improving the security and robustness of the network.

Considering each dApp has the different requirements for security and performance, it is allowed in Dora to specify which consensus algorithm for subchains to be used when creating subchains: DPoS+VRF, BFT, or both.

4.2 Formal Description

Algorithm 2 describes the overall process of DVBC. Among them, DPoS, VRF and BFT are respectively described by algorithm 3, 4 and 5.

DPoS provides a sampling algorithm based on the voting ratio as the probability of candidate nodes being selected.

The VRF algorithm first uses the public key of each candidate node and the Hash value of the previous block as the source information. The information is given to the VRF Hash function to calculate the new Hash value, and the 2*N/3+1 nodes are sorted by the value. The advantage of using VRF is that it can quickly form consensus, find candidate nodes, and be able to be verified.

The BFT algorithm is described by calculus π [12][13][14]. The basic syntax of calculus π is shown in figure 10. The advantage of using calculus π is that it is easy to analyze the feature of the algorithm, such as security and flexibility, and can also use software to do formal verification for the algorithm. The algorithm consists of 4 states: propose, prevote,

Algorithm 2: DVBC

Function `DVBC()`
  while *true* **do**
    list of delegates $L = \varnothing$;
    $L = \text{DPoS}(N/3, candidate\_votes)$;
    $L.append(\text{VRF}(2 * N/3 + 1, candidates))$;
    $\text{BFT}(L)$;
  end

Algorithm 3: DPoS

Function `DPoS(`*m, candidate_votes*`)`
  list of delegates $L = \varnothing$;
  $status = candidate\_votes$;
  $sum = summation(candidate\_votes)$;
  for $i = 0;\ i < m;\ ++i$ **do**
    $status\ += candidate\_votes$;
    $top = sort(status).top\_index())$;
    $L.append(top)$;
    $status(top)$ -= sum;
  end
  return L;

Algorithm 4: VRF

Function `VRF(`*m, candidates*`)`
  list of delegates $L = \varnothing$;
  $info = candidates.pub\_key + previous\_block\_hash$;
  $candidates\_hash = VRF\_hash(sec\_key, info)$;
  $topn = sort(candidates\_hash).topn(m))$;
  $L.append(topn)$;
  return L;

Algorithm 5: BFT

Procedure `Propose()`

$P_i^{k,p,v}$ ::= if $i == round\_robin(k)$ then

$\overline{cp_i} < pr > |V_i^{k,pr,v}$, where $pr = getProposal(p)$

else if $p \neq \varnothing$ then

$V_i^{k,p,v}$

else

$cp_{round\_robin(k)}(pr).V_i^{k,pr,v} + timeout_{round\_robin(k)}.V_i^{k,\emptyset,v}$

Procedure `Prevote()`

$V_i^{k,p,v}$ ::= $\overline{cv_i} < p > |(new\ c)(\prod_{j=1}^{N} cv_j(pr).\bar{c} < cv_j, pr > |VS_i^{k,p,v}(c))$

Procedure `PrevoteSub()`

$VS_i^{k,p,v}(c)$ ::= if $max_b(|\left\{pr \in v_k^{'} : pr.block == b\right\}|) > \frac{2}{3}N$ then

$C_i^{k,b,v}$

else if $|v_k^{'}| > \frac{2}{3}N$ then

$C_i^{k,\emptyset,v}$

else

$c(cv, vote)$. if $vote.round < k$ then

$cv(pr).\bar{c} < cv, pr > |VS_i^{k,p,v}(c)$

else if $vote.round == k$ then

$VS_i^{k,p,vote \cup v}(c)$

else

$P_i^{vote.round,p,vote \cup v}$

Procedure `Precommit()`

$C_i^{k,p,v}$ ::= $\overline{cc_i} < p > |(new\ c)(\prod_{j=1}^{N} cc_j(pr).\bar{c} < cc_j, pr > |CS_i^{k,p,v}(c))$

Procedure `PrecommitSub()`

$CS_i^{k,p,v}(c)$ ::= if $max_b(|\left\{com \in v_k^{"} : com.block = b\right\}|) > \frac{2}{3}N$ then

$\overline{cw_i} < b >$

else if $|v_k^{"}| > \frac{2}{3}N$ then

$P_i^{k+1,\emptyset,v}$

else

$c(pc, vote)$. if $vote.round < k$ then

$pc(com).\bar{c} < pc, com > |CS_i^{k,p,v}(c)$

else if $vote.round == k$ then

$CS_i^{k,p,vote \cup v}(c)$

else

$P_i^{vote.round,p,vote \cup v}$

$$
\begin{array}{lrl}
P ::= & 0 & \textit{empty summation} \\
\mid & P_1|P_2 & \textit{composition(parallelization)} \\
\mid & P_1 + P_2 & \textit{summation(nondeterministic choice)} \\
\mid & (new\ x)P & \textit{channel creation} \\
\mid & \alpha.P & \textit{prefix form} \\
\mid & \tau.P & \textit{silent action} \\
\mid & x(y).P & \textit{input} \\
\mid & \overline{x} < y > .P & \textit{output} \\
\mid & (x).P & \textit{restriction} \\
\mid & timeout_i.P & \textit{timeout of process } i \\
\mid & P^s(y) & \textit{function to which pass variable } s \textit{ and } y
\end{array}
$$

$$
\begin{array}{lrl}
\sum_{i=1}^{n} P_i ::= & P_1 + P_2 + .. + P_n & \textit{multiple summation} \\
\prod_{i=1}^{n} P_i ::= & P_1|P_2|..|P_n & \textit{multiple composition}
\end{array}
$$

Figure 10: Description of Calculus Symbol π

precommit and commit. P, V and C are the abbreviations of propose, prevote and precommit, which correspond to the 3 main functions in the algorithm respectively. For convenience to describe recursion relations, we also define subfunctions PrevoteSub and PrecommitSub, which are simplified as VS and CS. For getProposal function, it will return p directly when p is not empty, otherwise integrate the latest transaction information into a proposal and return it. The function $round\_robin(k) = k \mod N$ indicates selecting a process responsible for propose according to the current turn, similar to the role of leader. The consensus protocol between N nodes can be expressed as $Consensus ::= \prod_{i=1}^{N} Y_i$. The status of each node is recorded as $s = \{k, P, v\}$. k indicates the round. p represents the block information of the proposed round, and v represents the voting information of all rounds. $v_k^{'}$ and $v_k^{''}$ respectively represents prevote and precommit voting information in the k round. The process transfers information through a communication channel, where cp_i, cv_i, cc_i, and cw_i represent the communication channels for used for $i$process for propose, prevote, precommit, and commit.

The function Propose first judges whether it is the current proposer or not, if so, package the latest transaction to the propose PR to send PR through the *cp* channel to other processes; and enter the V state. If not, and there is propose already, it will also enter the V state; otherwise, accept the propose from the *cp* channel or go into the V state after the timeout.

The function Prevote will broadcast the received proposals to other processes through the *cv* channel and accept the proposal from the *cv* channel of other processes; then create a shared channel C, and send the received proposals through C; and enter the PrevoteSub subfunction. In the PrevoteSub function, if the same proposal is received by more than 2/3 processes, it enters the C state. Otherwise, if the received proposal of the process is over 2/3, but not consistent, it will enter C state carrying the empty proposal; otherwise, if the proposal of the 2/3 process has not been received yet, it will be handled according to concrete situation. After read the information of the channel C, it will keep waiting if the message is old; if it is the current round information, the voting information is merged and it will keeps waiting; if its round is backward, it returns to the P state and overtakes the current round.

The function Precommit broadcasts proposal to other processes from the *cc* channel and accepts a proposal from the *cc* channel of other processes; meanwhile, the channel C is created to send the received information . Similar to function Prevote , if the same proposal is received from more than 2/3 processes, the proposal is formally submitted; otherwise, if the proposal is inconsistent, go to the state of P and start the next round; otherwise, continue to wait, if you find that

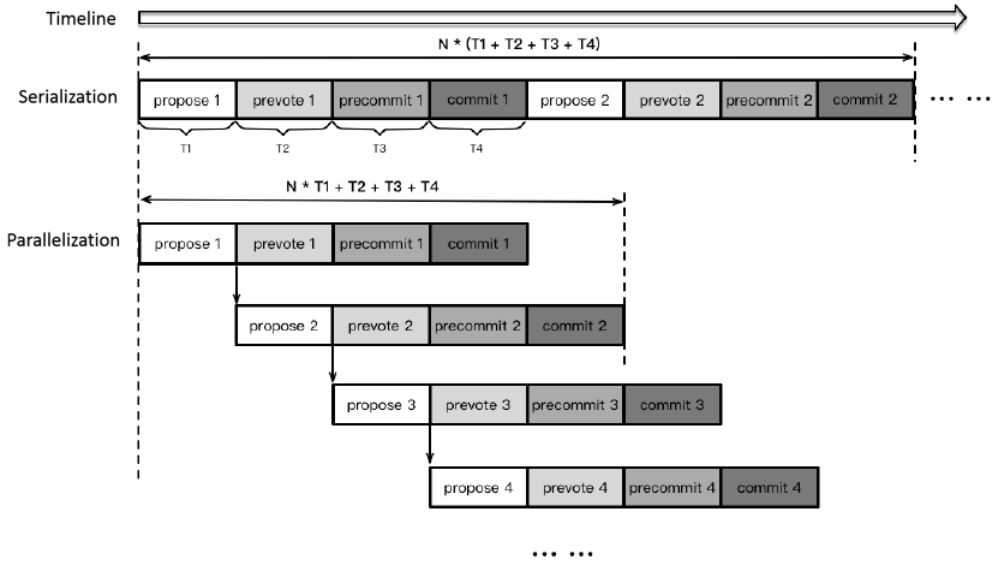

Figure 11: Consensus Algorithm Parallelization

your round is backward, then back to the state of P.

4.3 Parallelization of BFT algorithm

For four states of a block, propose, prevote, precommit, commit, assuming their corresponding time are $T1, T2, T3, T4$, the time needed to genterate and confirm N blocks by serial execution is $N * (T1 + T2 + T3 + T4)$.

From the point of view of pipeline, the four states can be executed in an orderly and parallel manner. Ideally, as shown in Figure 11, the time required to generate and confirm N blocks in pipeline mode is $N * T1 + T2 + T3 + T4$. When N is large enough, after this pipelining operation, the time is approximately $N * T1$.

5 Dora Features

5.1 Horizontal and Vertical Expansion

Dora supports not only conventional Multi Chain horizontal expansion, but also unique vertical expansion mechanism based on branch prediction , so it can achieve higher TPS than other systems.

5.2 Consensus Algorithm Parallelization

Dora also fully utilized the concept of pipelining in the design of the consensus algorithm, so that the block generation and confirmation process are parallelized, therefore it can also enhance the TPS.

5.3 Instant MintAction and SendMultiTransation

At present, the biggest application in Ethereum is the ICO, that is, to offer coin by creating an smart contract. In essence the transfer of the ICO token is also the interface calling of smart contract , which will lead to network congestion and resource waste when there are massive transfer transactions. Dora supports a special seigniorage transaction, MintAction, which solidify the behavior of the currency into the parent chain, making it easy to operate as the ordinary transfer

transaction. Dora supports a special mint transaction, MintAction, which solidify the behavior of minting into the Hub chain, making it as easy to operate as the ordinary transfer transactions.

MintAction: {
name: "UToken",
totalSupply: 100000000000000000000000000,
decimals: 18,
owner: "0xfbc2a4...ed",
symbol: "UT"
}

For the present issue of slow airdrop of tokens, Dora will also support instant multiple transfer transaction SendMultiTransaction, enable users to specify multiple transfer addresses at a time, and package multiple transactions ,and send it out together to save network resources.

SendMultiTransaction: {
from: ""0xfbc2a4...ed"",
symbol: UT,
toLen: 4,
toList: {
"0xfbc2a4...e1", 100
"0xfbc2a4...e2", 500

}
}

6 Dora Economic Model

Dora Network Limited will introduce DNT (Dora Network Token) Token to activate its internal ecosystem. The DNT token will be issued by Dora Network Limited, a Hong Kong company. DNT is intended to be the native crypto utility token to be used within the Dora Network and on decentralized applications that integrate within the Dora Network, circulating within its own system. DNT is the fuel for Dora Network. The initial offering of DNT token will be 1 billion. DNT Token is the GAS for transaction fee payment internal to Dora Network. When a user needs computation on Dora Network to send tokens, to interact with a contract and so on, the user must pay for that computation regardless of whether the transaction succeeds or fails. If it fails, the user must pay for that computation just like it would otherwise pay for a successful transaction because the miners must validate and execute the transaction in either scenario (compute). That payment is calculated in GAS and its paid in DNT. GAS is a unit that measures the amount of computational effort that is required to execute certain operations. Generally speaking, there are 2 types of accounts for Dora Network: External accounts and Contract accounts. All External accounts are controlled by private keys, without codes. Dora Network users can initiate transactions from External accounts through creating and signing transactions. Contract accounts are controlled by contract codes. Whenever contract accounts receive a transaction, contract codes will be executed, including read/write to internal storage, send Transactions or create contract, etc.

6.1 Rewards for Miners

Dora Network will allocate 5% of the total DNT tokens as the rewards for miners to create blocks for the first 5 years, 1% per year. After 5 years, the rewards mechanism and quantity will be determined by community votes. Dora Network will allow nodes to decide on how they will split on the rewards received for creating blocks in order to incentivize token holders to vote on accounting nodes. Assume the token rewards for creating a block is W, certain node is willing to use ratio p token as the voting reward, then when this node creates a block, all users who voted for this node will split the rewards of $W * p$ based on their individual votes. Every single voter will receive $I_u = \frac{V_u}{\sum_{k=1}^{N} V_k} * W * p$. The reward received

by this node consists of 2 parts: 1 part is the voting reward from voting for itself as a common user, and the other part is the reward as the accounting node. The formula is:

$$I_u = \frac{V_u}{\sum_{k=1}^{N} V_k} * W * p + W * (1 - p)$$

6.2 Free transaction for common transfers

All transactions occurred on Dora network will produce transaction fees. These fees are related to the consumption of computing or storage resources. These resources are represented by the quantity of GAS. The formula for calculating transaction fees (GAS cost) for common transfer or smart contracts is: Quantity of GAS consumed x Price for GAS. DNT is the unit for both GAS cost and GAS price.

When Dora Network users make common transfers, the transaction fees can be waived if they lockup their DNTs. The more DNTs are locked up, the more free quota a user will receive. Once the lockup period is up, the DNT tokens will be returned to its owners. To avoid 0-cost DDOS attack, Dora Network will calculate the GAS fee which has been accrued by this user based on the DNT tokens which has been locked up. Transaction fees exceeding the free quota will be charged for the actual GAS fees. Using N days as a cycle, the maximum free quota for account U within this cycle is calculated as follows:

$$F_u = \frac{U\text{'s locked tokens}}{\text{the locked tokens of all accounts at present}} * \text{The total } GAS \text{ cost accumulated to the current block in the cycle}$$

6.3 Community Regulations and Planning

The following Dora Network events must be decided by community voting. DNT token holders vote matches to the numbers of tokens they hold:

- Making strategic decisions
- Changes to miners reward mechanism and quantity
- Emergency events, such as, but not limited to, software security, system upgrade, etc.

Any decisions made by Dora communities must receive approval from 60% or above of the DNT tokens (Any change to this ratio must be approved by community vote as well). DNT Tokens will be managed by selected escrow company for digital assets liquidity, and the community will be updated openly and transparently.

Dora Foundation will provide a dApp Incubation program to encourage dApp development and ecosystem building.

7 Summary and Further Work

Dora proposes a SSP algorithm to parallelize smart contracts. It fully utilizes the parallelism at the instruction level, and designs a DVBC consensus algorithm that takes both security and efficiency into account, thus enabling our system to support commercial grade applications. In addition, Dora is fully compatible with Ethereum EVM, which reduces the transplant costs of dApp developers, and the free trading model can allow the general public to enter the Blockchain area at low cost as well.

Technology has always been progressing, and Dora will continue to focus on the latest research directions in the area of the Blockchain, absorbing and improving existing research results, such as more efficient and secure consensus algorithms. In addition, some trusted smart contracts will be introduced to accelerate the efficiency of parallel execution by specifying preconditions in advance. Dora will also consider to dynamically extend the type of transactions supported on the chain based on the need for upper level applications.

In addition to supporting EVM, Dora also plans to support WebAssembly, so that more developers can quickly develop dApp on Dora.

Appreciation

We sincerely appreciate the investment by Loopring foundation and the Hongkong Loopnest incubator into our project. Many of the ideas in this article come from Mr. Wang Dong, the CEO of Loopring[15]. He also provided a lot of help and guidance in this thesis writing, and we would like to express our heartfelt thanks to him.

References

[1] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. `https://bitcoin.org/bitcoin.pdf`, Dec 2008.

[2] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform. `https://github.com/ethereum/wiki/wiki/White-Paper`, 2014.

[3] Serguei Popov. The tangle. `https://www.iotatoken.com/IOTA_Whitepaper.pdf`.

[4] Ardor whitepaper. `https://www.jelurida.com/sites/default/files/JeluridaWhitepaper.pdf`, 2016.

[5] Jae Kwon and Ethan Buchman. Cosmos, a network of distributed ledgers. `https://cosmos.network/resources/whitepaper`.

[6] Asch whitepaper. `https://github.com/AschPlatform/asch/blob/master/docs/whitepaper/index.md`.

[7] PCHAIN Foundation. Pchain whitepaper. `https://pchain.org/#whitepaper`.

[8] Ethan Buchman. Tendermint: Byzantine fault tolerance in the age of blockchains. `http://atrium.lib.uoguelph.ca/xmlui/bitstream/handle/10214/9769/Buchman_Ethan_201606_MAsc.pdf`, Jun 2016.

[9] Joseph Poon and Thaddeus Dryja. The bitcoin lightning network. `https://lightning.network/lightning-network-paper.pdf`, 2016.

[10] dantheman. Dpos consensus algorithm - the missing white paper. `https://github.com/liuchengxu/blockchain-tutorial/blob/master/content/misc/dpos-consensus-algorithm-this-missing-white-paper.md`, 2017.

[11] Silvio Micali, Michael Rabin, and Salil Vadhan. Verifiable random functions. In *Proceedings of the 40th Annual Symposium on the Foundations of Computer Science*, pages 120–130, New York, NY, October 1999. IEEE.

[12] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, I. *Inf. Comput.*, 100(1):1–40, 1992.

[13] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, II. *Inf. Comput.*, 100(1):41–77, 1992.

[14] Robin Milner. *Communicating and mobile systems - the Pi-calculus*. Cambridge University Press, 1999.

[15] Daniel Wang, Jay Zhou, and Alex Wang. Loopring(loopring): A decentralized token exchange protocol. `https://github.com/Loopring/whitepaper/raw/master/zh_whitepaper.pdf`, 2018.

GENERAL INFORMATION

This whitepaper describes the initial sale in which the Dora Network Token (**DNT**) is sold. DNT is a cryptographic token that is designed to be used as outlined in this whitepaper. DNT is not, nor is it intended to, constitute a security, an investment scheme, financial instrument or any other regulated product in any jurisdiction. This white paper is not, nor is it intended to constitute, a solicitation, prospectus, offer document for investment and does not pertain in any way to an offering of securities, an investment scheme, a financial instrument or any other regulated product in any jurisdiction. Please note that purchases of DNT are final and non-refundable. Individuals, businesses, and other organizations should carefully weigh the risks, costs, and benefits of acquiring DNT.

LIMITATION OF THE PURCHASERS

You are not eligible to and you shall not purchase DNT through the Dora Network token sale if you are a citizen or resident (tax or otherwise) of any country or state where the purchase of DNT or similar cryptocurrencies or tokens, may be prohibited or the token sale is deemed to be non-compliant with the applicable laws and regulations. For clarity, natural persons and entities that are a resident of (tax or otherwise), domiciled in, or have a connection to, the United States of America, Japan, the Peoples Republic of China are expressly prohibited from participating in the token sale and purchasing DNT. Purchases of DNT should be undertaken only by natural persons, entities, or companies that have significant experience with, and a sophisticated understanding of, the usage and intricacies of cryptographic tokens and blockchain based software systems. Purchasers should have functional understanding of storage and transmission mechanisms associated with other cryptographic tokens. Any entities of Dora Network Limited and officers and employees thereof will not be responsible in any way for loss of any cryptographic tokens, DNT or fiat currency resulting from actions taken by, or omissions of, the purchasers. If you do not have the required experience or expertise, then you should not purchase DNT or participate in the token offering. You should carefully consider the risks, costs, and any other demerits of acquiring DNT, and, if necessary, obtain your own independent advice in this regard. If you are not in the position to accept nor to understand the risks associated with this token sale, or any other risks as indicated in this whitepaper, you should not acquire DNT, until such that you have received the necessary independent advice.

RISKS

The purchase of DNT carries with it risk. Prior to purchasing DNT, the purchaser should carefully consider the risks listed below and, to the extent necessary, consult a lawyer, accountant, and/or tax professional prior to determining whether to purchase DNT.

(a) DNT will be stored in a wallet, which can only be accessed with a password selected by the purchaser. If a purchaser of DNT does not maintain an accurate record of their password, this may lead to the loss of their tokens. If your password protection is weak and it is cracked or learned by somebody else, this may also lead to the loss of tokens. As a result, purchasers must safely store their password in one or more backup locations that are well separated from the primary location.

(b) The purchaser recognizes that some of the services in the Dora Network ecosystem are currently under development and may undergo significant changes before release and/or made available for use.

(c) The purchaser understands that while Dora Network Limited will make best efforts to release the Dora Network on time, it is possible that delays to the official release may occur.

(d) As with other cryptocurrencies and cryptographic tokens, value of DNT may fluctuate significantly and become reduced in value (including to zero value) for any number of reasons, including but not limited to, supply and demand, overall market conditions, political or geographical reasons, changes of regulations in any jurisdictions, and technical reasons.

(e) DNT will be issued on the Ethereum blockchain. As such, any malfunction or unexpected functioning of the Ethereum protocol may impact the purchasers ability to transfer or securely hold DNT. Such impact could adversely affect the value.

DISCLAIMER

To the maximum extent permitted by the applicable laws, regulations and rules, Dora Network Limited, any entities of the Dora Network Limited ecosystem, and officers and employees thereof shall not be liable for any direct, indirect, special, incidental, consequential or other losses of any kind, in tort (including negligence), contract, statute or otherwise (including but not limited to loss of revenue, income or profits, and loss of use or data), arising out of or in connection with any acceptance of or reliance on this whitepaper or any part thereof by you. Dora Network Limited and any entities of Dora Network Limited and officers and employees thereof shall not be liable for any loss of DNT after it is transferred to you by any reason including but not limited to your failure to maintain or backup an accurate record of your password or password cracking by somebody due to your poor maintenance of your password. Any person undertaking to acquire DNT acknowledges and understands however that Dora Network Limited does not provide any warranty as to the release of the Dora Network or any of the other technical features or services contemplated under this whitepaper. You acknowledge and understand therefore that Dora Network Limited (including its associated bodies corporate, officers and employees) assumes no liability or responsibility for any loss or damage that would result from or relate to the incapacity to use DNT. Regulatory authorities are carefully scrutinizing businesses and operations associated to cryptocurrencies and tokens in the world. In that respect, regulatory measures, investigations or actions may impact future business and may limit or prevent it from developing its operations in the future. Any person undertaking to purchase DNT must be aware that the Dora Network Limited business model or Dora Network may change or need to be modified because of new regulatory and compliance requirements from any applicable laws in any jurisdictions. In such case, purchasers and any person undertaking to acquire DNT acknowledge and understand that neither Dora Network Limited nor any of its affiliate shall be held liable for any direct or indirect loss or damages caused by such changes. This white paper and any other materials or explanations made by Dora Network Limited and its officers and employees shall not and cannot be considered as an invitation to enter into an investment. They do not constitute or relate in any way nor should they be considered as an offering of securities, a financial instrument an investment scheme or any other regulated product in any jurisdiction. This white paper does not include nor contain any information or indication that might be considered as a recommendation or that might be used as a basis for any investment decision. Neither Dora Network Limited nor any of its officers and employees are to be or shall be considered as advisor in any legal, tax or financial matters. Acquiring DNT shall not grant any right or influence over Dora Network Limited organization and governance to the purchasers.

NO REPRESENTATIONS AND WARRANTIES

Dora Network Limited does not make or purport to make, and hereby disclaims, any representation, warranty or undertaking in any form whatsoever to any entity or person, including any representation, warranty or undertaking in relation to the truth, accuracy and completeness of any of the information set out in this white paper. Further, no representation or warranty is given by Dora Network Limited as to the achievement or reasonableness of any plans, future projections or prospects set out in this white paper and nothing in this document is or should be relied upon as a promise or representation as to the future functionality, utility or availability of the Dora Network and/or its associated services. To the fullest extent permissible by law, Dora Network Limited excludes all liability (and is not liable for) any loss or damage of whatsoever kind (whether foreseeable or not) which may arise from any person acting on any information and opinions contained in this white paper or any information which is made available in connection with any further enquiries, notwithstanding any act or omission, negligence, default or lack of care, by Dora Network Limited, its entities, officers and/or employees.

The English version shall prevail in the white paper. In the event of a legal dispute or inconsistency, the English version shall prevail.