Dogechain

Whitepaper

V1.0

Doge Day (4/20), 2022

Table of Contents

Abstract

Dogecoin is a peer-to-peer digital currency which was created as a hard fork of Litecoin. Dogecoin transactions only take a minute while Bitcoin miners process transactions approximately every 10 minutes. The Scrypt hash puzzle is used for its Proof-of-Work (PoW) consensus algorithm to prevent miners from using special devices used for Bitcoin mining. Unfortunately, according to a recent crypto energy monitoring platform, Dogecoin consumes as much energy as El Salvador. Furthermore, Dogecoin does not natively support smart contracts. While Dogecoin uses the technology of Bitcoin spin-off Litecoin is an independent standalone chain, Shiba was created as a meme ERC20 token on Ethereum. Dogechain will also be a standalone chain which will be EVM compatible and be built to run smart contract-based applications. This chain will be developed by community developers and will be based on a Proof-of-Stake (PoS) mechanism designed to run parallel to the Dogecoin chain, providing a scalable, secure, reliable and flexible blockchain.

1. Introduction

Dogecoin is considered the first meme coin to gain attraction with commercial applications on social platforms such as Twitch and Reddit. Dogecoin was created as a hard fork of Litecoin (which was also a hard fork of Bitcoin), and is a peer-to-peer digital currency for sending and receiving digital payments processed through the Dogecoin blockchain network. Dogecoin has joined the cryptocurrency market by gaining a reputation as "joke money". Dogecoin is mostly driven by its enthusiastic owners as there is no CEO or an organization behind it. Dogecoin transactions only take one minute while Bitcoin miners can process transactions approximately every 10 minutes. Transaction fees of Dogecoin are also much lower than Bitcoin.

While SHA256 is used as a mining algorithm in the Bitcoin protocol, Scrypt is used in the Litecoin protocol. Dogecoin has also been using the Scrypt-based consensus mechanism in its Proof-of-Work (POW) mechanism. Therefore, it prevents miners from using special devices used to mine ASICs for Bitcoin. Just like with Bitcoin, the network participants of the Dogecoin blockchain use their computing power to secure the network and fuel block generation while verifying transactions. With each block created, miners are rewarded at a fixed rate of 10,000 after 600K-th block. Unfortunately, according to a recent crypto energy monitoring platform, Dogecoin consumes as much energy as El Salvador. While Dogecoin uses proof of work and is a chain itself, Shiba is another meme token which was created on Ethereum.

In this paper, we introduce Dogechain (Doge Chain with EVM) which is based on a PoS mechanism providing scalability, security, robustness, and sustainable blockchain that aims to run in parallel to the Dogecoin blockchain. Dogechain is going to be EVM-compatible and built for running smart contract-based applications. Note that EVM is a computation engine that facilitates the deployment and operation of smart contracts. They are generally written in higher level languages (e.g., Solidity) and then compiled into EVM bytecode. Without the EVM, it would not be possible to execute software programs on the Ethereum protocol. Therefore, the EVM is the most critical part of Ethereum's core architecture. Furthermore, an EVM compatible chain creates an environment that supports code execution in an environment similar to Ethereum's Virtual Machines. Hence, instead of developers building solutions and protocols from scratch, they can now build them faster and deploy them to the blockchain much quicker.

Dogechain will also have a large decentralization architecture and high performance. However, it will not face the issues as occurred Ethereum with PoW, such as low tps, public chain congestion, and high fees for a high degree of decentralization. Furthermore, Dogechain flexibly realizes the establishment of smart contracts without impacting the public chain itself. Dogechain will also have its own smart contracts, thus building the extensive DeFi ecosystem. Also, Dogechain relies on a system of a pre-defined number of validators with a variant of Proof-of-Stake (PoS) consensus mechanisms leading to shorter block time and lower fees. Validator candidates which have the highest number of staking can become validators and produce blocks. Slashing scenarios will also be in place such as double signing, hence leading to security, decentralization, reliability, transparency, stability, and block finality.

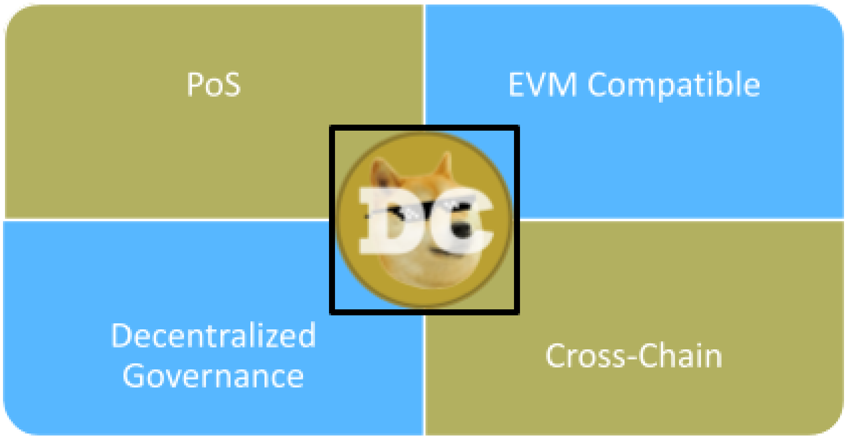

Figure 1: High-level features of Dogechain

Main Features of Dogechain:

- **IBFT Proof-of-Stake (PoS) consensus:** Community users can participate in the network which ensures a permissionless and decentralization chain.
- **EVM-compatible:** Existing Ethereum smart contracts can easily be migrated to Dogechain without requiring any further modification.
- **Decentralized Governance:** Community users (token holders) can propose proposals, delegate, vote on the blockchain parameters & events and influence governance decisions.
- **Cross-chain:** Dogecoin can be easily cross-chain transferred to Dogechain and vice versa.

2. Background

2.1 Cryptographic Hash Functions

One of the most important tools in blockchain is cryptographic hash functions which are used for integrity of transactions and guarantee immutability. It basically produces a fixed size of output (called fingerprint or digest) of arbitrarily long input data. More concretely, a hash function

$$H: \{0, 1\}^{*}\ \ \{0, 1\}^{k}$$

takes on input of any size and produces a fixed k length output. It must satisfy the following properties:

- It is easy to compute H regardless of input data size.
- Given any h, it is computationally infeasible to find an input x such that $H(x) = h$.
- Given any x, it is also computationally infeasible to find y such that $H(y) = H(x)$ and $x \neq y$.
- It is computationally infeasible to find any (x, y) such that $H(x) = H(y)$ and $x \neq y$.

SHA-256 and Keccak-256 are widely used in several blockchains, and they produce a hash (output) of size 256 bits.

2.2 Digital Signatures

2.2.1 Secp256k1 Curve

Note that all elliptic curves are equations are defined as $y^2 = x^3 + ax + b$. Secp256k1 is an elliptic curve used by several blockchains to implement public and private key pairs. Secp256k1 has been defined where $a = 0$ and $b = 7$ (i.e., secp256k1 lives on the equation $y^2 = x^3 + 7$). Basically, whenever a user is willing to generate a public and private key pair (pk, sk), he/she first generates a sufficiently large random number (which is going to be sk) and then multiplies private key by the generator point G as $sk. G$ (which is going to be pk). This number is going to be a defined point on the secp256k1 curve. Due to the underlying discrete log problem (DLP), no one can derive the private key from the given public key and the generator point (as long as the key size is sufficiently large).

Note that for each value of x, the y component is squared in this equation leading to have two symmetric points across the x-axis. Hence, there are two values of y called odd and even numbers. Therefore, public keys can be identified by the x-coordinate and the parity of the y-coordinate. In the blockchain space, this feature is quite important as it saves significant data storage.

2.2.2 ECDSA Signature Algorithm

Elliptic Curve Digital Signature Algorithm (ECDSA) is a cryptographic algorithm for creating digital signatures. More concretely,

Setup

- **Public Parameters:** Let F_q be a finite field, two parameters a and b define an elliptic curve C over F_q, a seed which validates C, a prime integer $n > 2^{255}$, and a point $G \in C$ of order n where q is either prime or a power of 2.
- **Private Key:** An integer d in $[1, n - 1]$.
- **Public Key:** $Q = dG$.

Signature generation for a given message M:

- Generate $k \in [1, n - 1]$
- Compute

$$(x_1, y_1) = kG$$
$$r = x_1 \, mod\, n$$
$$s = \frac{H(M) + dr}{k} mod\, n$$

- If $r = 0$ or $s = 0$, try again. The signature is (r, s).
- **Signature:** (M, r , s).

Verification:

- Given (M, r', s').
- Verify if r' and s' are in $[1, n - 1]$ and that $r' = x_1 \, mod\, n$ for $(x_1, y_1) = u_1G + u_2Q, u_1 = \frac{H(M)}{s'} \, mod\, n$, and $u_2 = \frac{r'}{s'} \, mod\, n$.

2.3 Ethereum Virtual Machine (EVM)

A virtual machine is basically a layer of abstraction between the executable code and the executing machine. This layer is necessary to improve the portability of software and to ensure that applications are separated from each other and from their hosts. The Ethereum Virtual Machine (EVM) is a software platform that developers can use to build decentralized applications (DApps) on Ethereum. Basically, all Ethereum accounts and smart contracts live in this virtual machine.

The Ethereum virtual machine and EVM codes are designed using memory, bytes, along with blockchain concepts such as Proof-of-Work (PoW) or Proof-of-Stake (PoS), Merkle tree, and hash functions. The purpose of the EVM is to determine what the total Ethereum state will be for each block in the blockchain.

2.4 Consensus Protocols

2.5.1 Proof-of-Work (PoW)- Nakamoto Consensus

Proof-of-Work (PoW) is a decentralized consensus protocol that can be handled securely in a peer-to-peer network without requiring any trusted third party. It solves the difficulty of Byzantine general problem in an open network where miners can generate arbitrary identities to compete for the next generated blocks by solving a random hash puzzle. This is also called a Sybil attack because it requires almost no additional resources. In order to avoid a Sybil attack, PoW is used to force the miners to have and run certain computational resources. PoW also protects the security of the blockchain from the longest chain attacks. Unfortunately, PoW requires a large amount of energy and keeps increasing if more miners join the network.

2.5.2 Istanbul Byzantine Fault Tolerant (IBFT)

IBFT is another Byzantine fault-tolerant protocol based on Practical Byzantine Fault Tolerance (PBFT). On a high level, Byzantine consensus is achieved deterministically as follows: 1) a leader or bidder/proposer is selected. 2) Each proposed block goes through several stages of communication between the nodes before being added and confirmed on the blockchain.

There are four types of messages which are exchanged between the nodes:

- **Pre-Prepare, Ready, Commit**: Used through ordinary consensus algorithms operations.
- **Round robin**: Used to select a new block producer when the current producer is suspected of failing or when the block has not been created within a specific time frame.

There are two approaches in Polygon Edge framework for choosing block producers:

- **Round robin**: This is a block producer selection strategy where a different bidder is chosen for every block producing phase.
- **Attached bidd**er: A new bidder is only selected whenever a malicious behavior has been detected by the current bidder.

In these two approaches, every validator knows in advance which one of them is going to be the next block producer. This is because the decision is made through deterministic calculations based on node IDs. Similar to PBFT, IBFT also guarantees that there will be only one single bidder in each round. Moreover, the bidder is required to get responses from the other nodes in order to continue executing its further tasks. This means that in the case of a network partition with more than n nodes (at least more than $3n + 1$ nodes), the protocol does not make any decisions not to break the consensus until the partition is fixed and their communication is timely synced. This also allows immediate finality where no forks are ever allowed to occur.

2.5.3 IBFT Proof of Authority (PoA)

In PoA, validators are responsible for creating blocks and adding sequentially to the blockchain. All validators create a dynamic set of validators where validators can be added to or removed from the cluster using a decentralized voting mechanism. This means that validators can be included or excluded from a validator group if the majority (51%) of validator nodes voted to add/remove a particular validator from the set. Thus, malicious validators can be detected and removed from the network at any point in time, and new trusted validators can be added to the network. All validators in turn propose the next block (by means of the round robin leader selection), and for a block to be validated/added to the blockchain, the overwhelming majority of the validators (i.e., more than 2/3) must approve that block. In addition to the validators, there are also non-validators who do not participate in block generation but participate in the block validation process. IBFT PoA is the default consensus mechanism of Polygon Edge framework.

2.5.4 IBFT Proof-of-Stake (PoS)

The Polygon Edge Proof-of-Stake (PoS) implementation is intended to be an alternative to the existing IBFT PoA implementation by giving node operators the ability to easily select between the two when starting the chain. Epochs are considered to be specific timeframes (in blocks) during which a given set of validators can generate blocks. The epoch length can be changed, meaning that the node operators can set the length of the epoch during instance creation. At the end of each epoch, an epoch block is created, and after this event, a new epoch begins. Validator sets are updated at the end of every epoch period. Nodes request a set of validators from the staking smart contract during the creation of an epoch block and store the resulting data in local storage. This query and saving the cycle are recurring at the end of every of epoch period. Fundamentally, this allows the staking smart contract to have full control over the addresses in the validator group, leaving only one task to the nodes. Each contract query is executed only once per period to obtain the latest information about the validator set. This removes the responsibility of dealing with validator sets from individual nodes.

2.5.5 RAFT

Raft is a distributed consensus mechanism relies on Paxos. The Raft protocol works with a node failure model where each node failure (e.g., missing messages, network partitions, or hardware-only failure) is considered a node failure. Hence, it should run $n \geq 2f + 1$ where f is the maximum number of nodes that can fail and n is the total number of nodes. The Raft protocol first selects a leader among a set of nodes and then makes the leader fully responsible for receiving transaction requests and handling the copying of logs (i.e., blocks) on other nodes. Each node can be either a candidate or a follower or a leader. The leader selection procedure is deterministic, so the protocol cannot run until the leader is selected by more than half of the nodes.

2.5.6 Comparison and Selection

IBFT protects the blockchain against various malicious attacks, while Raft only protects against node failures. Therefore, if we assume that that all nodes will never be corrupted, then Raft can be used without having any concern. However, if there is an assumption of having partially trust to the validators, then it would be better to utilize IBFT. Since Dogechain is decentralized and permissionless, it is going to run IBFT as its underlying consensus protocol.

3. Dogechain (DC) Architecture

Dogechain is using the Polygon Edge framework to build its standalone blockchain and will not use Polygon's "security as a service". The following features are disabled: checkpointing mechanism and mainchain contracts. This framework helps the community developers to build their own blockchain network, possibly with different required features. The Polygon Edge modules span the entire blockchain stack. These modules have been designed with developers' experience leading to extended multiple features. Polygon Edge is a modular and extensible framework for creating EVM-compatible blockchain networks, sidechains, and global scaling solutions. Its main purpose is to launch a new blockchain network as ensuring full compatibility with Ethereum smart contracts and transactions. Polygon Edge uses the IBFT consensus mechanism with two different approaches such as PoA and PoS. Doge EVM chain continues with IBFT PoS with built-in system contracts. With the help of Polygon Edge, the following features can be achieved:

- Reuse existing Ethereum smart contract technology and its API.
    - Users can interact with standard wallets via JSON-RPC.
    - Solidity/Vyper development, full EVM support.
    - Utilizing widely used Ethereum tools, development tools, and libraries.
    - Optimized UX when performing cross-network transactions
- Communication between networks.
    - Completely untrusted and decentralized embedded Ethereum Bridge solution.
    - Asset transfers from any EVM compatible network, in particular Polygon and Ethereum main-nets.
    - Transferring of ERC20 tokens, NFTs, or local tokens in the shell.
    - Customizing the bridge functionality with the existing plugins.
- Special Functions.
    - Building by the development of plugins
    - Developer friendly plugin system.
    - Replacing core functionalities with the consensus plugins.
    - Going beyond Ethereum smart contracts with Runtime plugins.

3.1 Dogechain Layering Architecture

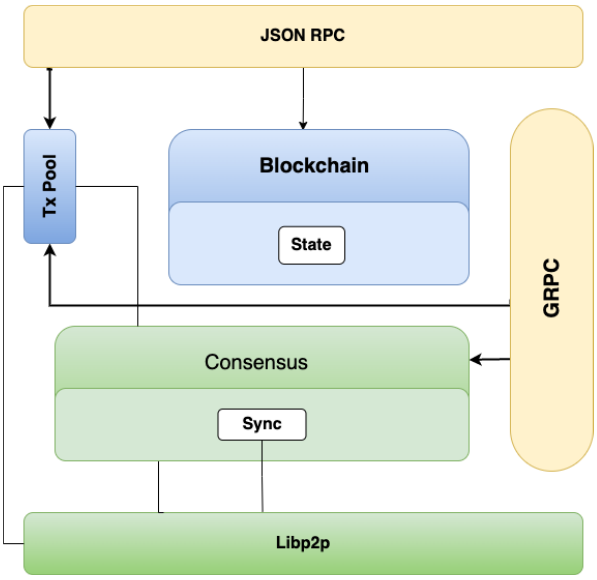

Figure 2: Layers of Dogechain

- **Libp2p:** This module always starts at the underlying network layer that libp2p uses. Libp2p is modular, extensible, and fast. In particular, it provides an excellent foundation for more advanced features.
- **Synchronization & Consensus:** The separation of synchronization and consensus protocols allows for modularity and implementation of customizable synchronization and consensus mechanisms depending on how the client operates. Polygon Edge offers pluggable consensus algorithms out-of-the-box.
- **Blockchain:** The Blockchain layer is the core layer that manages every task in the Polygon Edge system.
- **State:** The State layer contains a logic for transitioning between states. It deals with how the state changes when a new block is added.
- **JSON RPC:** DApp developers use this layer as an API layer in order to interact with the blockchain.
- **TxPool:** The TxPool layer is a transaction pool and is tightly coupled to other modules in the system as transactions can be added from multiple entry points.
- **GRPC:** The GRPC level is very crucial for interaction with the operator. In this way, node operators can interact with the clients easily, providing a usable and efficient UX.

3.2 Dogechain Cross-Chain Protocol

This protocol will play between Dogecoin chain and Dogechain chain. Dogecoin will be 1:1 to enter and exit the Dogechain Chain. When a Dogecoin is pegged to the Dogechain, the Dogechain will mint a wrapped Doge token (e.g., wDoge) for the user. On the contrary, when a user destroys a wDoge token, he can withdraw a Dogecoin from the Dogechain chain, and the pegged relationship is 1:1. In this context, cross-chain bridge protocol module will be utilized to achieve cross-chain transactions.

Dogechain Cross-Chain Protocol supports the Dogecoin cross-chain transfer. The main features of the cross-chain protocol are:

1. Decentralized and secure cross-chain support of Dogecoin to Dogechain (via Dogecoin client).
2. The protocol uses a trustless key generation for threshold signature schemes, and those generated private shares of the signing key will be used to calculate final signed transactions. The private key shares will also be managed by community, third-party partners, etc to eliminate the single-point-of-failure (i.e., centralization).
3. The protocol governance provides the support for voting on the organizations or institutions which run the cross-chain protocol.

3.3 Dogechain Design

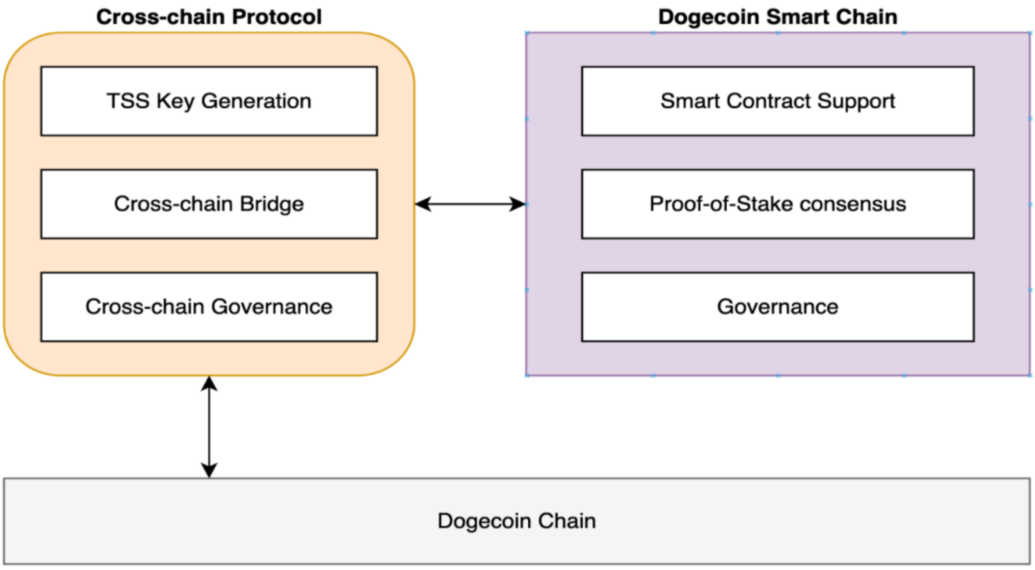

Figure 3: High-Level Architecture of Dogechain

The above diagram shows the relationship between Dogechain Chain and Dogecoin chain:

1. Users can lock their Dogecoin on the cross-chain protocol and get wDoge on Dogechain chain.
2. Users can use wDoge to deploy & interact with smart contracts, pay transaction fees, and participate in the governance of Dogechain.
3. Users can destroy wDoge and claim their native Dogecoin.

3.4 Native Currency of Dogechain: DC

Dogechain has its own network coin called DC token. There is also wDoge token which are all crossed from the Dogecoin blockchain using bridge. All DC tokens are pre-mined but a tiny amount of wDoge is minted in the genesis (only 1000 wDoge will be mined for bridge signer gas fee). Dogechain in principle will have two options for transaction gas fee payment. It can be charged in either DC or wDoge. During the initialization phase, wDoge will be used to bring and motivate more wDoge to the system. In the short term, DC will also be utilized once the chain is stabilized.

3.5 Dogechain Configurations

- IBFT PoS with built in systems contracts will be used as a core consensus algorithm by Dogechain.
- Average block time is expected to be 2 seconds.
- Initially, 15 nodes will be running to comply with BFT (Byzantine Fault Tolerance).
- Block size is going to be dynamic and will be decided by the Validator set. The initial block gas limit is 30,000,000.
- Expected number of validator nodes in the chain will be minimum 21 in the long run.
- Any account staking more than 10,000,000 DC token and passing the community authority and authentication, could join the Validator Set.
- Dogechain has pre-deployed contracts for Staking, which would yield DC token for staking profit.
- If the block is not produced or accepted within the expected time, the next validator would take over the proposer duty.
- There is no newly minted block reward for block production. All transaction fees will be valued in DC and will be distributed to block producer.

4. Smart Contracts of Dogechain

The management of the validator along with their selections, reward distributions, and staking are all performed by the contracts. These contracts are deployed in the genesis block. In Dogechain, there are basically six different types of smart contracts.

- **Governance Contract:** It manages validator proposals and votes.
- **Validator Set Contract:** It is responsible for ranking validators and decides which are to be elected or removed.
- **Vault Contract**: All withdrawal fee on chain-bridge sent to vault contract.
- **Staking Contract:** It is responsible for staking and unstaking operations, and also distribution of block rewards.
- **Slashing Contract:** It is responsible for punishing operations against validators who do not follow the expected rules of the chain.
- **Bridging Contract:** This contract manages token exchange between Dogecoin blockchain and Dogechain.

Validator Set Contract, Slashing Contract, Bridging Contract and Vault Contract will be system contracts and they are deployed in the genesis block.

4.1 Governance Contract

Blockchain networks are autonomous platforms that evolve on their own and provide transparency through one-to-one democratic community interaction. On-chain management is an approach for recommending and making changes to blockchains. In this type of governance, change initiation rules are commonly pre-hard-coded into the blockchain protocol. Community-selected validators suggest possible ideas through code updates and written suggestions. All chosen validators and regular users vote to accept/reject the proposed change.

Under the governance contract, community members democratically vote on proposals that will advance the development of the blockchain network. To be able to recommend a proposal, the user must have a sufficient number of DC token shares. On the other hand, people with a certain amount of DC tokens can vote on proposed proposals. There will also be an option to report management commitments to report misuse of contracts. Changing the sample options below is subject to change by the community.

- Minimum staking amount for being a validator
- Minimum staking amount for general user
- Minimum staking amount for giving a proposal
- …

4.2 Validator Set Contract

This contract validates and stores the nodes that meet the requirements of becoming a validator. Also, the contract can list the main validators and their addresses, the last created and approved block, and classify the blocks produced by specific validators.

4.3 Vault Contract

All withdrawal fee on chain-bridge sent to vault contract.

4.4 Staking Contract

This contract performs staking, reward calculation, and distribution of rewards to either users or validators. This contract also periodically updates the rewards received by the validators and shareholders. The IBFT PoS consensus mechanism ensures decentralization and community participation. DOGE holders, including validators, can stake their tokens “pegged” to a share.

4.5 Slashing Contract

Like the Binance Smart Chain slashing mechanism, Dogechain also adopts the slashing methodology for enhancing the security of the Dogechain chain. Slashing is part of the on-chain governance to ensure that malicious or dishonest behaviors are being punished. Dogechain chain slash evidence can be submitted by anyone. The transaction submission demands a slashing proof and cost fees but also brings a higher reward once it is successful. Two types of slashing behaviors such as double signing and unavailability are considered below.

- **Double Signing**: Let us assume that two block headers have the same height and the same parent block hash. If these two block headers are sealed by the same validator and different signatures are created, then this validator will be punished for jailing forever.
- **Unavailable**: If a validator misses 48 blocks per 24 hours, then it cannot receive any reward from a block fee. If the validator misses more than 96 blocks for 24 hours, then it will be punished for 10K DC token and it will be jailed for 3 days. During the jail, it can still both produce or validate a block.

4.6 Bridge Contract

In Bridge contract, the stakeholder can call withdraw of the bridge contract to destroy the native token of the EVM chain, and the redeemed native token will be transferred to the designated address of the Dogecoin chain.

- The minimum recharge value of the native token is 100 DOGE.
- When the transaction is synchronized, multiple operator (of the bridging signers) will sign and confirm the transaction and call the deposit of the bridge contract to write data.
- After more than half of the operator confirm (by means of digitally signing procedure), the native token will be added to the recharge address which is specified by the user.

5. Token Economy

The total supply of DC tokens will be 1 trillion. 60% of the total tokens are considered community tokens and will be freely given to the community.

- 2% of the supply will be allocated for existing DOGE holders. People will be able to receive their airdrop by wrapping their DOGE by bridging it onto the Dogechain network. This will occur over a 10-day period, most likely May 20 till June 1 2022.
- For the subset of Robinhood users who cannot yet withdraw, Dogechain will reserve 1% of the DC tokens for when Robinhood unlocks their DOGE holders.
- 28.2% will be given out over the following 4 years. Each month, of the amount unlocked, 2/3 will be airdropped to people who stake DOGE on Dogechain. 1/3 will be airdropped to DC holders who lock their tokens. This ratio can be changed by a vote (Locked DC holders are the only ones who can vote).
- The remaining 28.8% goes to community members who lock up DC holders as rewards. Every year, a pre-defined percentage (which can be changed by DC holders) will be given to DC holders.

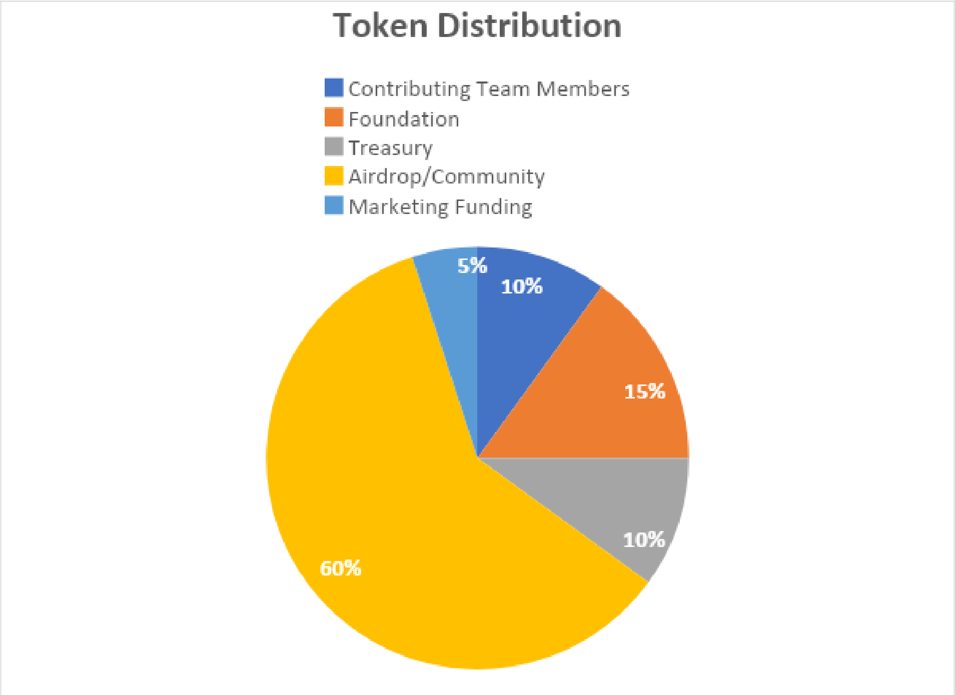

Figure 4: Token Economy of Dogechain

OVERALL TOKENOMICS

Allocation	Amount (DC Token)	Vesting
Contributing Team Members	10%	5-year linear vesting
Foundation	15%	15% upfront, 2-year linear vesting
Treasury	10%	15% upfront, 2-year linear vesting
Airdrop/Community	60%	4-year linear vesting
Marketing Funding	5%	50% upfront, %50 VE Lock
Total	**100%**	

Table 1: Token Economy of Dogechain

6. Potential Applications on top of Dogechain

6.1 NFT

On Dec 2021, a developer has minted the first non-fungible token (NFT) on the Dogecoin blockchain where the transaction id is 19aeaa88859c04a333257f1119a77438ac08feec424c6ad3645a0679c8be9882. The hash of the NFT has been written as OP_RETURN 064019ad26590b8b7a43caa2ee75a309a2d28269dbbb1d33c6e6306188ab9400.

By Dogechain, we can now publish our own NFT following the ERC721 standard which has a set of rules making it easy to work with NFTs. Unlike ERC20, ERC721 is unique and indivisible and can be used to proof of ownership of a digital good or a real-world asset.

6.2 DeFi & GameFi

All EVM compatible DeFi (such as Uniswap) can be integrated with Dogechain. Furthermore, by means of Layer 2 solutions (both ZK Rollups and Optimistic Rollups), Dogechain can provide significant improvements over existing transaction per seconds and provide different applications including privacy preserving systems.

7. Implementation details

The source codes and further information are available on https://github.com/Dogechain-lab.

References

1. Marco Mazzoni, Antonio Corradi, Vincenzo Di Nicola. Performance evaluation of permissioned blockchains for financial applications: The ConsenSys Quorum case study, Blockchain: Research and Applications, Volume 3, Issue 1, 2022, 100026, ISSN 2096-7209, https://doi.org/10.1016/j.bcra.2021.100026.

2. Crypto Energy Consumption. https://www.moneysupermarket.com/gas-and-electricity/features/crypto-energy-consumption/, 2021.

3. Bitcoin vs. Ethereum vs. Dogecoin: Top cryptocurrencies compared. https://www.bankrate.com/investing/bitcoin-vs-dogecoin-vs-ethereum-crypto-comparison/, Dec 2021.

4. First NFT ever minted on Dogecoin Blockchain. https://twitter.com/inevitable360/status/1470414541490110472/, Dec 2021.

5. https://blockchair.com/dogecoin/transaction/19aeaa88859c04a333257f1119a77438ac08feec424c6ad3645a0679c8be9882, Dec 2021.

6. Optimistic Rollups vs ZK Rollups: Examining Six of the Most Exciting Layer 2 Scaling Projects for Ethereum, https://limechain.tech/blog/optimistic-rollups-vs-zk-rollups/, Aug 2021.

7. Dogechain. https://Dogechain.com/.

8. Ethereum Virtual Machine. https://ethereum.org/en/developers/docs/evm/.

9. Jury. https://docs.Dogechain.community/docs/overview

10. Polygon Edge. https://github.com/0xPolygon/polygon-edge

11. https://polygon.technology/solutions/polygon-edge/

12. Paxos, Raft, EPaxos: How Has Distributed Consensus Technology Evolved? https://www.alibabacloud.com/blog/paxos-raft-epaxos-how-has-distributed-consensus-technology-evolved_597127, Jan 2021.

13. An Introduction to Binance Smart Chain (BSC), https://academy.binance.com/en/articles/an-introduction-to-binance-smart-chain-bsc, Sep 2021.

14. Shiba Token, https://shibatoken.com/, 2021.

15. The Raft Consensus Algorithm, https://raft.github.io/, 2021.

16. Paxos consensus for beginners, https://medium.com/distributed-knowledge/paxos-consensus-for-beginners-1b8519d3360f, May 2020.

17. Ongaro, J. Ousterhout, In search of an understandable consensus algorithm Proceedings of the 2014 USENIX Conference; 19–20; Philadelphia, PA, USA, USENIX Association, pp. 305-320, June 2014.

18. Optimistic vs. ZK Rollup: Deep Dive, https://blog.matter-labs.io/optimistic-vs-zk-rollup-deep-dive-ea141e71e075, Nov 2019.

19. Bitcoin Whitepaper. https://www.ussc.gov/sites/default/files/pdf/training/annual-national-training-seminar/2018/Emerging_Tech_Bitcoin_Crypto.pdf, Oct 2008.

20. M. Castro and B. Liskov, “Practical byzantine fault tolerance,” in Proceedings of the 13rd Symposium on Operating Systems Design and Implementation, vol. 99, 1999, pp. 173–186.

21. Polygon Edge. D. Ongaro, J. Ousterhout, In search of an understandable consensus algorithm, in: Proceedings of the 2014 USENIX Conference; 19–20 Jun 2014; Philadelphia, PA, USA, USENIX

Association, 2014, pp. 305–320. L. Lamport, The part-time parliament, ACM Trans. Comput. Syst. 16 (2) 133–169, 1998.

22. Leslie Lamport. 1998. The part-time parliament. ACM Trans. Comput. Syst. 16, 2, 133–169. DOI: https://doi.org/10.1145/279227.279229, May 1998.