Dogechain

Whitepaper

V1.0

Doge Day (4/20), 2022

Table of Contents

Abstract

This whitepaper proposes a full overview of the standalone Dogechain blockchain, its key concepts, and its core principles. The following text outlines several major pain points common to the original Dogecoin cryptocurrency and how Dogechain can solve these lingering issues. The text also details how Dogechain complements the existing Dogecoin ecosystem via its incorporation of smart contracts. In addition, this whitepaper examines the technicalities of bridging the Dogecoin blockchain with Dogechain and its capacity for interoperability. It also introduces the $wDOGE and $DC cryptocurrencies and thoroughly reviews their use cases within the Dogechain ecosystem. Finally, this whitepaper delves into the project's tokenomics and surveys the token's distribution, vesting periods, and release schedule.

1. Introduction

1.1 Dogecoin - The Original Meme Coin

Dogecoin was the first meme coin released to the public in 2014. The idea behind it was to veer away from the overtly- serious cryptocurrency investment narrative to provide a whimsical coin that mainstream users could identify with. Not surprisingly, the token achieved immediate and extensive success. The Dogecoin community began to grow exponentially as well, especially after crucial influencers such as Elon Musk and Snoop Dogg endorsed the native cryptocurrency. Today, Dogecoin is considered one of the most popular cryptocurrencies in existence along with Bitcoin and Ethereum. As the first successful meme crypto, Dogecoin has inspired a multitude of copycats, including Shiba Inu or Dogelon Mars.

From a technical standpoint, Dogecoin is a fork of Litecoin, the "silver to Bitcoin's gold". Consequently, Dogecoin is mineable, enabling more than 10,000 new coins to be released into the market every minute. Moreover, the token supply is not capped as it is with Bitcoin.

The $DOGE cryptocurrency has a single use case - to be accepted as a means of payment for exchanging goods and services online. Given its growing popularity, it seems to have achieved this goal quite admirably. People simply love using Dogecoin and participating in the meme culture that is associated with it. Conclusively, $DOGE has gained mainstream recognition and adoption.

Having said that, Dogecoin's age is starting to catch up with it. Its fundamentals have remained stagnant relative to the multitude of tokens that have evolved with the medium.

1.2 The Unfortunate Shortcomings of Dogecoin

While $DOGE continues to excel at payments, this feature remains its sole use case. At a time when blockchain technology promises extensive utility through smart contracts, Dogecoin remains a one-trick pony. As a result, Dogecoin users cannot readily use their tokens in gaming, DeFi, or NFTs. This failure is especially egregious to DeFi fans who witness Doge-like tokens

gaining traction with investors (mostly due to their passive yield opportunities). Without smart contract functionality, Dogecoin is left out of this narrative.

Moreover, as a fork of Litecoin, Dogecoin uses the proof-of-work (PoW) Scrypt mining algorithm for validating transactions and creating new coins. While Scrypt is easier to mine than Bitcoin's SHA-256, the PoW architecture remains difficult to scale for mass usage. In its current form, micro-transactions could easily create the kind of network congestion that would slow it down to a crawl.

Additionally, crypto mining is considered a notoriously wasteful process for validating blockchain transactions. A study by Digiconomist revealed that Dogecoin consumes as much as 6.54 TWh, roughly the energy needs of a small country. Unfortunately, $DOGE's increasing popularity promises to increase its carbon footprint even further.

Finally, it's worth noting that Dogecoin's PoW protocol presents insurmountable challenges to implementing smart contracts. A PoW consensus mechanism simply can't scale to meet mass demand for millions of simultaneous transactions, even if they merely fuel dApps. Even Ethereum is migrating to a more scalable PoS mechanism to alleviate this issue. Consequently, the most viable solution is to implement a complementary blockchain with a PoS token that prioritizes fast transactions and enables smart contract functionality.

2. Introducing Dogechain

Dogechain is an EVM-compatible blockchain that aims to complement the original Dogecoin cryptocurrency. As a proof-of-stake blockchain, Dogechain seeks to bring scalability, security, robustness, and utility to Dogecoin. In short, Dogechain doesn't compete with Dogecoin. Instead, it aims to harmonize with the original meme crypto and enhance it with smart contract capability.

It's important to note that the Dogechain project is a community-first blockchain that aims to empower Dogecoin holders and enthusiasts. Dogechain will ultimately provide Dogecoin users with access to blockchain games, NFTs, and the ever-growing DeFi ecosystem, one in which they can showcase their favorite meme coin for a wide range of applications. In addition, Dogechain burns DOGE, helping solve Dogecoin's infinite supply problem and making Dogecoin far more valuable.

2.1 Solutions brought by Dogechain

The main goal of Dogechain is to **increase the use cases of Dogecoin by providing it with much-needed utility.** Dogecoin users can achieve this goal by merely wrapping their $DOGE into Dogechain smart contracts and receiving $wDOGE PoS tokens in return. $wDOGE tokens live on the Dogechain blockchain and will allow users to access an ecosystem of DeFi products, NFTs and GameFi, all indirectly powered by their original $DOGE tokens. Examples of potential use cases include:

- Participating in the NFT market through minting and exchanging NFTs by paying for gas with $DOGE.
- Partaking in lucrative GameFi opportunities and engaging with the growing blockchain gaming community.
- Joining decentralized exchanges to swap tokens and speculate on their value.
- Accessing advanced financial instruments such as staking, lending, borrowing, and liquidity mining.
- Taking part in the upcoming metaverse revolution through Dogechain-powered NFTs.
- Participating in DAOs and funding entire communities.
- And many more…

In sum, Dogechain promises to transform the single-usage Dogecoin crypto into a DeFi powerhouse. With any luck, Dogecoin will be able to readily compete with many of the top smart contract platforms in the current blockchain environment.

2.2 Characteristics of Dogechain

Dogechain relies on the Polygon Edge framework to build its standalone, EVM-compatible blockchain. EVM stands for Ethereum Virtual Machine, which means that this smart contract-capable platform will be compatible with dApps deployed on Ethereum.

EVM is at the core of the Ethereum blockchain and plays an instrumental role in creating decentralized applications. In particular, it allows developers to build and deploy solutions and protocols much more quickly (as opposed to building them from scratch). Indeed, EVM-compatible protocols incorporate a robust and proven architecture and are thus a game-changer for DeFi product developers. And in addition to existing protocols, Dogechain will propose its own smart contracts, thus building upon the extensive DeFi ecosystem.

Bitcoin and other payment-focused / store-of-value blockchains haven't been able to invoke the same demand as smart contract-capable platforms. In contrast, Dogechain's ability to improve Web3 ecosystem productivity promises to increase blockspace demand. This event will equally play a part in increasing demand for the native cryptocurrency of Dogechain, the $DC token.

Given Dogechain's capacity for high throughput and decentralization, token users will not need to suffer the same user concerns associated with many PoW tokens (including low transactions per second, public chain congestion, centralized mining, and high transaction fees). Moreover, Dogechain will conserve a high degree of decentralization due to its PoS architecture.

Dogechain relies on a predefined number of validators to facilitate its Proof-of-Stake (PoS) consensus mechanism, a setup that leads to shorter block times and lower fees. In PoS, validator candidates with the highest number of tokens staked are allowed to become validators and produce blocks. The token also employs slashing scenarios, hence leading to security, decentralization, reliability, transparency, stability, and block finality.

2.3 Main Features of Dogechain

Dogechain relies on the following key principles:

- **IBFT Proof-of-Stake (PoS) consensus:** Community users can participate in the network which ensures a permissionless and decentralized blockchain.
- **EVM-compatible**: Existing Ethereum smart contracts can easily be migrated to Dogechain without requiring any further modification.
- **Decentralized Governance:** Community members (token holders) can make proposals, delegate, vote on the blockchain parameters & events, and influence governance decisions.
- **Cross-chain compatibility:** Dogecoin can be easily utilized on the Dogechain network by wrapping the Dogecoin via the Dogechain bridge, and sent back to the Dogecoin network as needed.

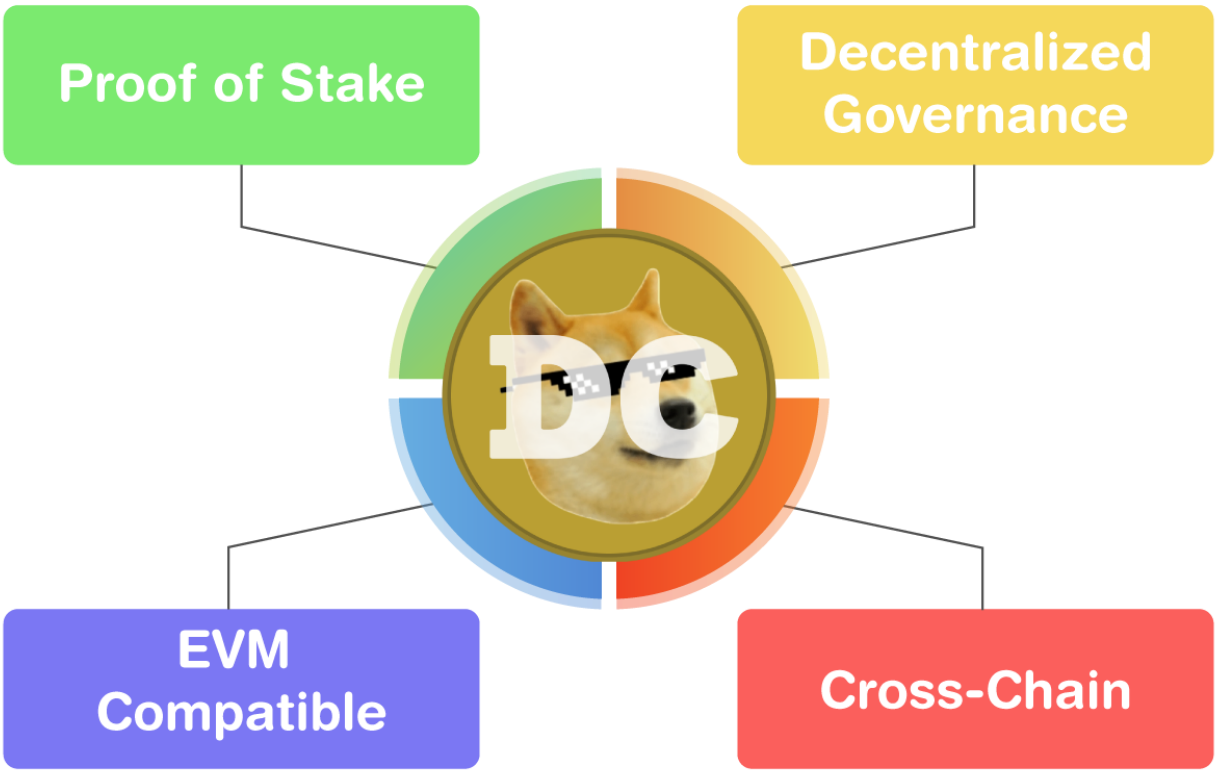


Fig1. High-Level Features of Dogechain

3. Background

3.1 Cryptographic Hash Functions

An essential tool in blockchain technology is the cryptographic function that ensures transaction integrity and immutability. An essential element of this is the hash function, a mathematical

algorithm that produces a fixed size numerical output (called fingerprint or digest) consisting of input data. More specifically, a hash function can be denoted as:

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

A hash function takes on the input of any size and produces a fixed k length output. In addition, 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 256 bits in size.

3.2 Digital Signatures

3.3.1 Secp256k1 Curve

Note that all elliptic curves are equations defined as y2 = x3 + ax + b. The code Secp256k1 is an elliptic curve used by several blockchains to implement public and private key pairs. For instance, we can define Secp256k1 as a = 0 and b = 7 (i.e., secp256k1 lives on the equation y2 = x3 + 7).

Before a user generates a public and private key pair (pk, sk), he/she must first generate a sufficiently large random number (which is going to be sk) and use it to multiply with the private key by the generator point G as sk.G (which is going to be the pk).

We use this number to define a 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 having 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 crucial, as it saves significant data storage.

3.3.2 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

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

- 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 $r' = x_1 \bmod n$ for $(x_1, y_1) = u_1 G + u_2 Q$, $u_1 = \frac{H(M)}{s'} \bmod n$, and $u_2 = \frac{r'}{s'} \bmod n$.s').

3.3 Ethereum Virtual Machine (EVM)

A virtual machine is 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. 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.

3.4 Consensus Protocols

3.4.1 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 (also called a Sybil attack) to compete for the next generated blocks by solving a random hash puzzle.

In order to avoid a Sybil attack, PoW is used to force the miners to have and run predefined computational resources. Additionally, PoW protects the security of the blockchain from the longest chain attacks. Unfortunately, PoW requires a large amount of energy which keeps increasing as more miners join the network.

3.4.2 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:

4. A leader or bidder/proposer is selected.
5. 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.

Additionally, there are two approaches in the 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 bidder**: 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.

3.4.3 IBFT Proof of Authority (PoA)

In PoA, validators are responsible for creating blocks and adding them sequentially to the blockchain. All validators create a dynamic set of validators where validators can be added 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 propose the next block in turn (by means of the round-robin leader selection). 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 directly but take part in the block validation process. IBFT PoA is the default consensus mechanism of the Polygon Edge framework

3.4.4 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 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.

3.4.5 RAFT

Raft is a distributed consensus mechanism that relies on Paxos. The Raft protocol works with a node failure model where each error (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, 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.

3.5. Comparison and Selection

IBFT protects the blockchain against various malicious attacks, while Raft only protects against node failures. If we assume that all nodes will never be corrupted, then Raft can be used without having any concern.

However, if there is an assumption of only having partial trust in 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.**

4. Dogechain (DC) Architecture

Dogechain uses the Polygon Edge framework to build a standalone blockchain. Consequently, it doesn't use Polygon's "security as a service" features but rather relies on its own set of validators. It's worth noting that Dogechain disables two Polygon Edge features - its checkpointing mechanism and its mainchain contracts.

With this framework, our community of developers can build a blockchain network that better suits their needs and demands. They can achieve this because Polygon Edge employs a modular and extensible framework for creating EVM-compatible blockchain networks, sidechains, and global scaling solutions. After all, Polygon Edge is primarily used to launch new blockchain networks that are fully compatible with Ethereum smart contracts and transactions.

Finally, Polygon Edge uses the IBFT consensus mechanism since it provides for PoA and PoS. Likewise, the Dogechain EVM blockchain invokes IBFT PoS with built-in system contracts. With the help of Polygon Edge, Dogechain can employ the following features:

- Reuse existing Ethereum smart contract technology and its API.
    - Users can interact with standard wallets via JSON-RPC.
    - Developers enjoy Solidity/Vyper programming and full EVM support.
    - Access to popular Ethereum tools, development tools, and libraries.
    - Optimized UX when performing cross-network transactions.
- Communication between networks.
    - Completely trustless and decentralized embedded Ethereum Bridge solution.
    - Asset transfers from any EVM compatible network, particularly Polygon and Ethereum mainnets.
    - Transferring of ERC20 tokens, NFTs, or local tokens in the shell.
    - The ability to customize bridge functionality with existing plugins.
- Special Functions.
    - Building network usability via the development of plugins
    - The capacity to replace core functionalities with consensus plugins.
    - Going beyond Ethereum smart contracts by incorporating Runtime

Thanks to the underlying Polygon Edge architecture, Dogechain can achieve full compatibility with Ethereum smart contract technology. It can also use IBFT PoS to ensure high network decentralization, security, and scalability.

4.1 Dogechain Layering Architecture

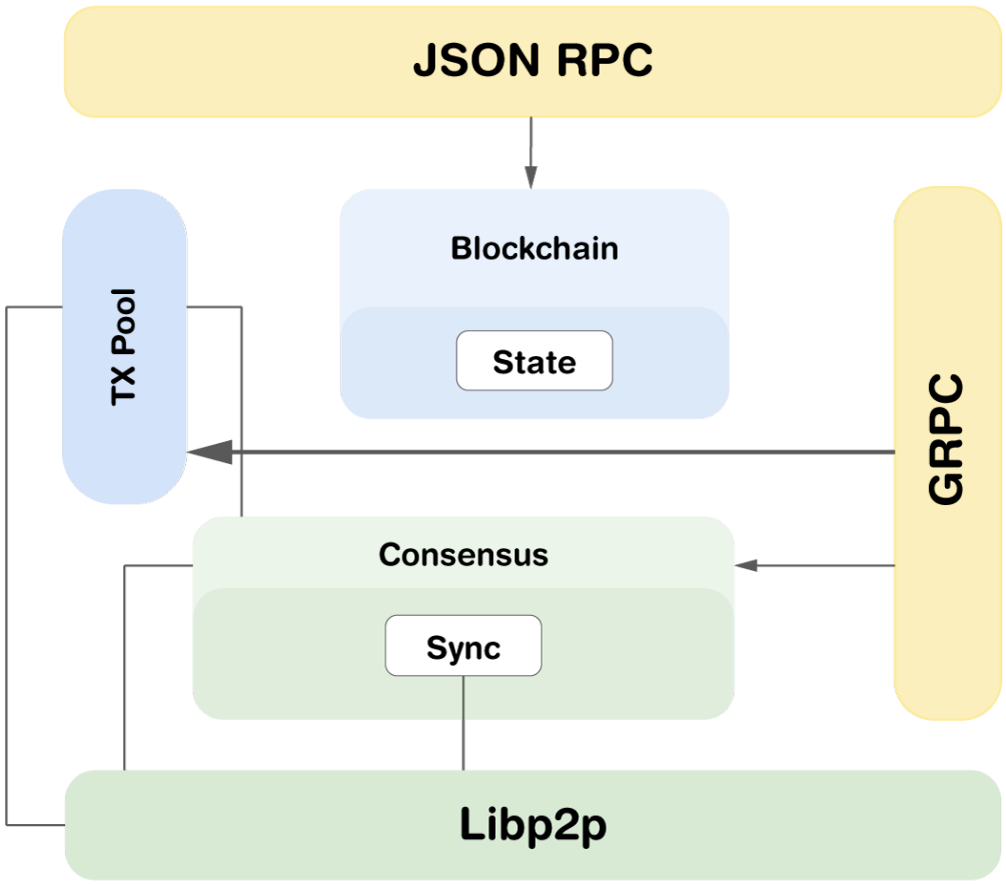


Fig 2. Dogechain Layered Architecture

- **Libp2p:** This module always begins at the underlying network layer. 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 enable modularity and the implementation of customizable synchronization and consensus mechanisms (depending on how the client operates). Polygon Edge also offers pluggable consensus algorithms out-of-the-box.
- **Blockchain:** The Blockchain layer serves as the core layer for managing tasks within the Polygon Edge system.

- **State:** The State layer provides the 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 layer is crucial for enabling interaction with the operator. This layer ensures that node operators can interact with the clients easily, providing a usable and efficient UX.

4.2 Dogechain Cross-Chain Protocol

This Dogechain Cross-Chain Protocol is essential to linking the original Dogecoin blockchain to the Dogechain. This protocol requires a ratio of 1:1 $DOGE coin to enter or exit the Dogechain. When users peg their Dogecoin to the Dogechain, the Dogechain protocol mints a wrapped $DOGE token (**$wDOGE**).

Conversely, when a user destroys a **$wDOGE** token, he can withdraw a Dogecoin from the Dogechain chain using a ratio of 1:1. In this context, a cross-chain bridge protocol module will be utilized to achieve cross-chain transactions.

The primary features of the cross-chain protocol are:

1. Decentralized and secure cross-chain support of Dogecoin to Dogechain (via Dogecoin client).
2. A trustless key generation for threshold signature schemes. Generated private shares of the signing key will be used to calculate final signed transactions.
3. The private key shares will also be managed by the community and third-party partners to eliminate any risk of a single-point-of-failure (i.e., centralization).
4. The protocol governance mechanism supports voting capabilities for organizations that run on the cross-chain protocol.

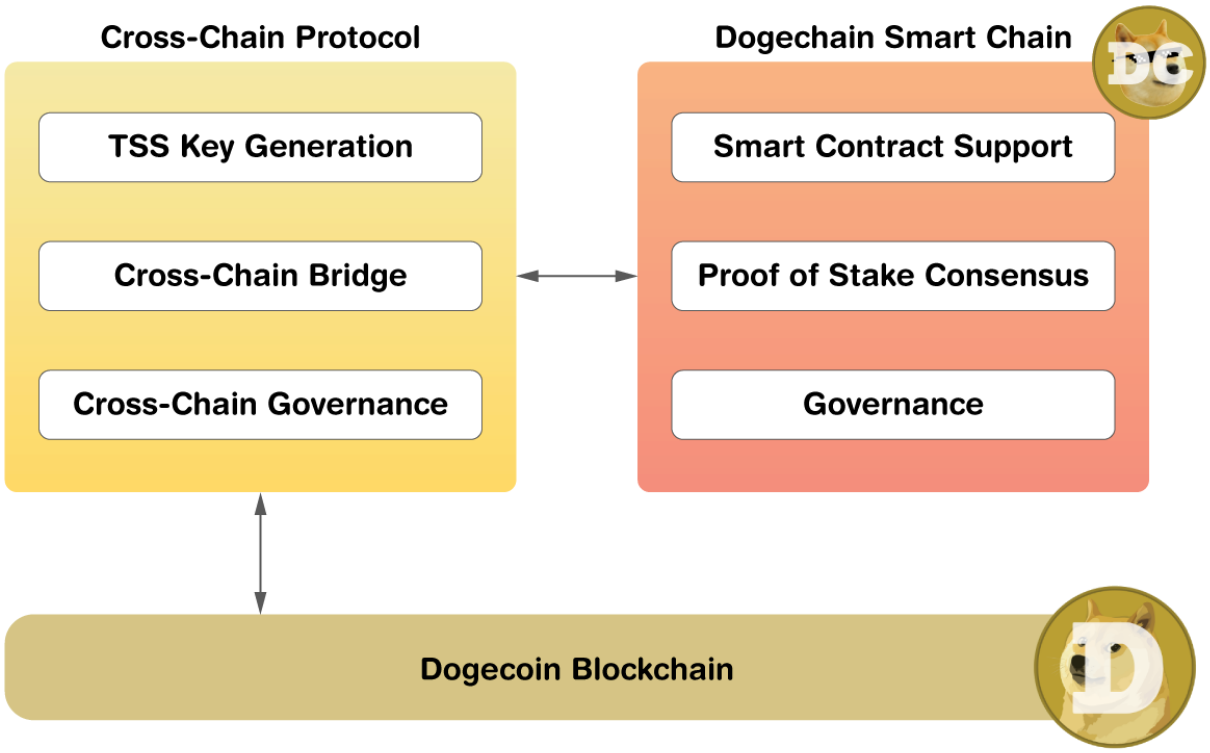


Fig 3. Dogechain Cross-Chain Protocol

4.3 Dogechain Design

As shown above, the Dogechain Chain and the Dogecoin chain have a symbiotic relationship. In particular,

- Users can lock their Dogecoin on the cross-chain protocol to receive $wDOGE on the Dogechain blockchain.
- Users can use $wDOGE to deploy and interact with smart contracts, pay transaction fees, and participate in the governance of Dogechain.
- Users can destroy $wDOGE and reclaim their native Dogecoin.

4.4 Native Currency of Dogechain: the $DC token

In addition to $wDOGE, Dogechain introduces a native cryptocurrency - **the Dogechain token ($DC).** This community-focused token serves as a primary governance token for the Dogechain blockchain and comes with various use cases. It's worth noting that the entirety of the $DC tokens supply will be pre-mined upon the release of the mainnet. The protocol will simultaneously mint a small amount of $wDOGE (1000 tokens) to serve as fuel for signing the initial bridging gas fees.

Users will have two distinct options to pay for transaction fees on Dogechain - $wDOGE and $DC. During the initialization phase of the Dogechain, it's expected that users will commonly use $wDOGE for this purpose. As the blockchain stabilizes, $DC will become the "go-to token" for fueling transactions, smart contracts, and dApps.

4.5 Dogechain Configurations

- An IBFT PoS with built-in systems contracts will be used as a core consensus algorithm by Dogechain.
- The average block time is expected to be 2 seconds.
- Initially, 21 nodes will be running to comply with BFT (Byzantine Fault Tolerance).
- Block size will be dynamic and decided by the Validator set. The initial block gas limit is 30,000,000.
- The expected number of validator nodes in the chain will be 21 at a minimum.
- Any account staking more than 10,000,000 $DC tokens and passing the community authority and authentication, will be allowed to join the Validator Set.
- Dogechain has pre-deployed contracts for staking. This allows for the staking of $DC tokens, providing rewards to holders.
- 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 either $wDOGE or $DC.

5. Smart Contracts of Dogechain

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

- **Governance Contract -** manages validator proposals and votes.
- **Validator Set Contract -** ranks validators and decides which are to be elected or removed.
- **Vault Contract -** receives all the withdrawal fees from the chain bridge.
- **Staking Contract -** manages staking operations and the distribution of block rewards.
- **Slashing Contract** - manages disciplinary actions against validators who do not follow the predetermined rules of the chain.
- **Bridging Contract** - manages token exchange between the Dogecoin blockchain and the Dogechain.

5.1 Governance Contract

Blockchain networks are autonomous platforms that evolve on their own and provide transparency through peer-to-peer 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 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 proposals. There will also be an option to report management commitments to report misuse of contracts.

The following sample options are subject to change following community feedback:

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

5.2 Validator Set Contract

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

5.3 Vault Contract

All withdrawal fees from the chain bridge are sent to the Vault Contract.

5.4 Staking Contract

This contract performs staking, reward calculation, and distribution of rewards to both users and 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 $wDOGE share.

5.5 Slashing Contract

Dogechain adopts a slashing methodology similar to the one used by the Binance Smart Chain. In addition to enhancing the security of the Dogechain chain, slashing is used to safeguard its on-chain governance mechanisms from malicious or dishonest behavior via disciplinary actions.

Dogechain chain slash evidence can be submitted by anyone. It's worth noting that each transaction submission demands a slashing proof and is subject to fees. That said, it also produces a higher reward if it is successful.

Two types of slashing behaviors are considered below:

- **Double-Signing:** Let us assume that two different 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 and jailed permanently.
- **Unavailable:** If a validator misses 48 blocks per 24 hours, it will be unable to receive rewards from the block fees. If a validator misses more than 96 blocks for 24 hours, the validator will be punished for 10,000 $DC tokens and will be jailed for 3 days. During jail time, it will still be able to produce or validate blocks.

5.6 Bridge Contract

Stakeholders can call upon the Bridge contract to withdraw their native $DOGE and destroy the native token of the EVM chain. The protocol will then transfer the redeemed token to the designated address of the original Dogecoin chain. The minimum reclaim value of the native token is 100 $DOGE.

When the transaction is synchronized, multiple operators (of the bridging signers) will sign and confirm the transaction and call upon the bridge contract to write data. After more than half of the operators confirm (by means of a digitally signing procedure), the native token will be added to the reclaim address which is specified by the user.

6. Dogechain Staking

The Dogechain project will enable users to access two different token staking models:

- Staking $wDOGE tokens on the Dogechain blockchain will allow stakeholders to secure the native blockchain and receive $DC rewards.
- Staking $DC tokens into the Dogechain Ve model will allow users to receive $veDC tokens. They can select a vesting time between 1 week and 4 years, with longer vesting periods granting higher $DC rewards and more $veDC in return.

The process of staking goes as follows:

1. Users wrap $DOGE onto the Dogechain to receive $wDOGE.
2. During the airdrop window, users receive a 1-time airdrop for this action in $DC tokens in an amount equal to their $wDOGE.
3. Users can now stake $wDOGE on Dogechain and receive $DC rewards.

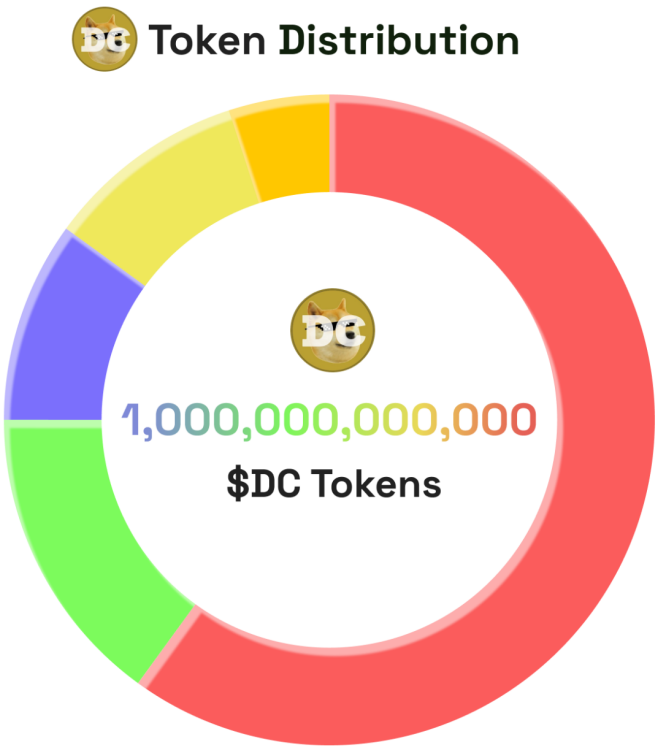
4. Users can now stake the $DC they received as rewards on the Ve model and receive additional $veDC rewards.

7. Token Economy

The protocol will release a total supply of 1 trillion $DC tokens. Moreover, the entirety of the supply will be pre-mined at launch.

As a community first project, 60% of the total supply of $DC tokens will be directly distributed to the community. This distribution will respect the following schedule:

- 2% of the supply will be allocated to existing $DOGE holders. Users will be able to receive their airdrop by wrapping their $DOGE and bridging it onto the Dogechain network. This will occur over a 10-day period, most likely from May 20 until June 1, 2022.
- For the subset of Robinhood users who cannot yet withdraw, Dogechain will reserve 1% of the $DC tokens. Robinhood users will be able to claim their $DC when Robinhood unlocks their $DOGE holdings.
- 28.2% will be given out over the following 4 years. Each month, the protocol will proceed with a linear distribution as follows:
    - 2/3 of the unlocked tokens will be airdropped to people who **stake $wDOGE** on Dogechain.
    - 1/3 will be airdropped to $DC holders who lock their tokens in the **Dogechain Ve model**. These users will receive $veDC tokens that will provide passive yields in $DC. Moreover, only $veDC holders will be able to vote on the reward changes.
- The remaining 28.8% goes to community members who lock up their $veDC tokens in the Ve model. Every year, a pre-defined percentage will be distributed to $veDC holders.



Allocation	Amount	Vesting
Airdrop/Community	60% (600BN $DC)	4-Year Linear Vesting
Foundation	15% (150BN $DC)	15% Upfront, 2-Year Linear Vesting
Treasury	10% (100BN $DC)	15% Upfront, 2-Year Linear Vesting
Contributing Team Members	10% (100BN $DC)	5-Year Linear Vesting
Marketing Funding	5% (50BN $DC)	50% Upfront, 50% VE Lock

Table 1: Token Economy of Dogechain

8. Potential Applications on top of Dogechain

8.1 NFTs

Dogechain will provide its users with the capability to publish their own NFTs following the ERC721 protocol. Since this proven NFT standard is widely accepted by marketplaces and metaverses, Dogechain NFT owners will be able to integrate their NFTs into the existing NFT landscape.

8.2 DeFi

As an EVM-compatible blockchain, DeFi protocols such as Uniswap and SushiSwap can be seamlessly integrated with Dogechain. $wDOGE and $DC are DeFi-capable cryptocurrencies that can be locked in various liquidity pools and provide rewards to their holders. Moreover, they will be able to use them as collateral on decentralized lending platforms, exponentially increasing the utility of their original Dogecoin.

In addition, several Layer 2 solutions found within the Polygon Edge architecture (including both ZK Rollups and Optimistic Rollups) will enable Dogechain to make improvements on their existing transaction speeds in DeFi and address some privacy concerns.

8.3 GameFi

Dogechain will provide developers with the ability to build entire virtual worlds and blockchain games on the Dogechain smart contract framework. As a result, the $wDOGE and $DC cryptocurrencies will enable users to participate in virtual gaming economies and share digital resources in their favorite metaverses.

9. Implementation details

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

10. References

- 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.
- Crypto Energy Consumption. https://www.moneysupermarket.com/gas-and-electricity/features/crypto-energy-consumption/, 2021.
- Bitcoin vs. Ethereum vs. Dogecoin: Top cryptocurrencies compared. https://www.bankrate.com/investing/bitcoin-vs-dogecoin-vs-ethereum-crypto-comparison/, Dec 2021.
- First NFT ever minted on Dogecoin Blockchain. https://twitter.com/inevitable360/status/1470414541490110472/, Dec 2021.

https://blockchair.com/dogecoin/transaction/19aeaa88859c04a333257f1119a77438ac08feec424c6ad3645a0679c8be9882, Dec 2021.
- 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.
- Dogecoin. https://Dogecoin.com/.
- Ethereum Virtual Machine. https://ethereum.org/en/developers/docs/evm/.
- Jury. https://docs.Dogechain.community/docs/overview
- Polygon Edge. https://github.com/0xPolygon/polygon-edge https://polygon.technology/solutions/polygon-edge/
- 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.
- An Introduction to Binance Smart Chain (BSC), https://academy.binance.com/en/articles/an-introduction-to-binance-smart-chain-bsc, Sep 2021.
- Shiba Token, https://shibatoken.com/, 2021.
- The Raft Consensus Algorithm, https://raft.github.io/, 2021.
- Paxos consensus for beginners, https://medium.com/distributed-knowledge/paxos-consensus-for-beginners-1b8519d3360f, May 2020.
- 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.
- Optimistic vs. ZK Rollup: Deep Dive, https://blog.matter-labs.io/optimistic-vs-zk-rollup-deep-dive-ea141e71e075, Nov 2019.
- Bitcoin Whitepaper. https://www.ussc.gov/sites/default/files/pdf/training/annual-national-training-seminar/2018/Emerging_Tech_Bitcoin_Crypto.pdf, Oct 2008.
- M. Castro and B. Liskov, “Practical byzantine fault tolerance,” in Proceedings of the 13th Symposium on Operating Systems Design and Implementation, vol. 99, 1999, pp. 173–186.
- 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.
- 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.