Canto Blockchain

Innovative EVM-based blockchain built on Cosmos

WhitePaper Version 1.0

Abstract: This white paper elucidates the core technology, key concepts, and principles of Canto. Canto is a public EVM (Ethereum Virtual Machine) compatible blockchain based on the Tendermint consensus mechanism, offering advantages such as low transaction fees, fast confirmations, double validation, and secure random number generation. Furthermore, the white paper provides a comprehensive overview of Canto's ecosystem layout and development roadmap, encompassing cross-chain interoperability protocols and decentralized trading platforms. Lastly, the white paper delves into the token economics of the project, outlining the token distribution, release, and issuance plans.

目录

1 Blockchain Industry Background

1.1 Explanation of Blockchain Concept

Blockchain is a distributed ledger technology based on cryptographic principles that allows transaction records, smart contracts, and other data to be stored across different nodes, enabling decentralized data sharing and exchange. It is the underlying technology of cryptocurrencies, ensuring consensus among users without mutual trust.

The most intriguing aspect of blockchain is that no individual or entity can control it. Instead, transactions are verified and confirmed by decentralized nodes, achieving a distributed operation. Blockchain offers numerous benefits, including transparency, speed, security, and has played a crucial role in various sectors such as government, finance, supply chain, logistics, and media.

1.2 Development History of Blockchain

For most people, it's hard to dissociate blockchain from the impression of cryptocurrencies, but in reality, it has always been an information technology. Its origin can be traced back to the foundational communication assumption called the "Byzantine Generals Problem" in 1982. It aimed to establish a reliable distributed system on multiple zero-trust nodes and ensure consensus and consistency in information transmission.

The solution evolved over time, incorporating cryptography, timestamps, proof-of-work mechanisms, and so on. In 2009, an individual under the pseudonym Satoshi Nakamoto released a white paper introducing a decentralized, peer-to-peer electronic cash system – Bitcoin. This marked the completion of the basic form of blockchain and elevated its prominence.

- ✓ **Blockchain 1.0**—Blockchain emerged in the form of cryptocurrencies. It provided a decentralized digital currency solution, ensuring the security and anonymity of currency transactions through blockchain technology. Bitcoin, the most famous, paved the way for digital payments, transfers, and remittances. However, Blockchain 1.0 was limited to currency transactions and couldn't accommodate more complex applications.

- ✓ **Blockchain 2.0**—Blockchain entered the phase of programmable public chains and extended to assets such as stocks, bonds, and loans. It also introduced DAO (Decentralized Autonomous Organization) and smart contracts, enabling more diverse applications. Ethereum became the first platform to implement smart contracts on the blockchain, automating various business logic. Smart contracts were not only applicable to currency transactions but also to more complex scenarios like voting and digital identity verification.

- ✓ **Blockchain 3.0**—Blockchain evolved towards achieving a "programmable business economic model." In Blockchain 3.0, assets could be placed on the chain, creating a variety of applications within a broad foundational framework. This facilitated decentralized solutions for various industries, confirming, measuring, and storing value-representing information and bytes through blockchain, enabling traceability, control, and trading of assets on the blockchain.

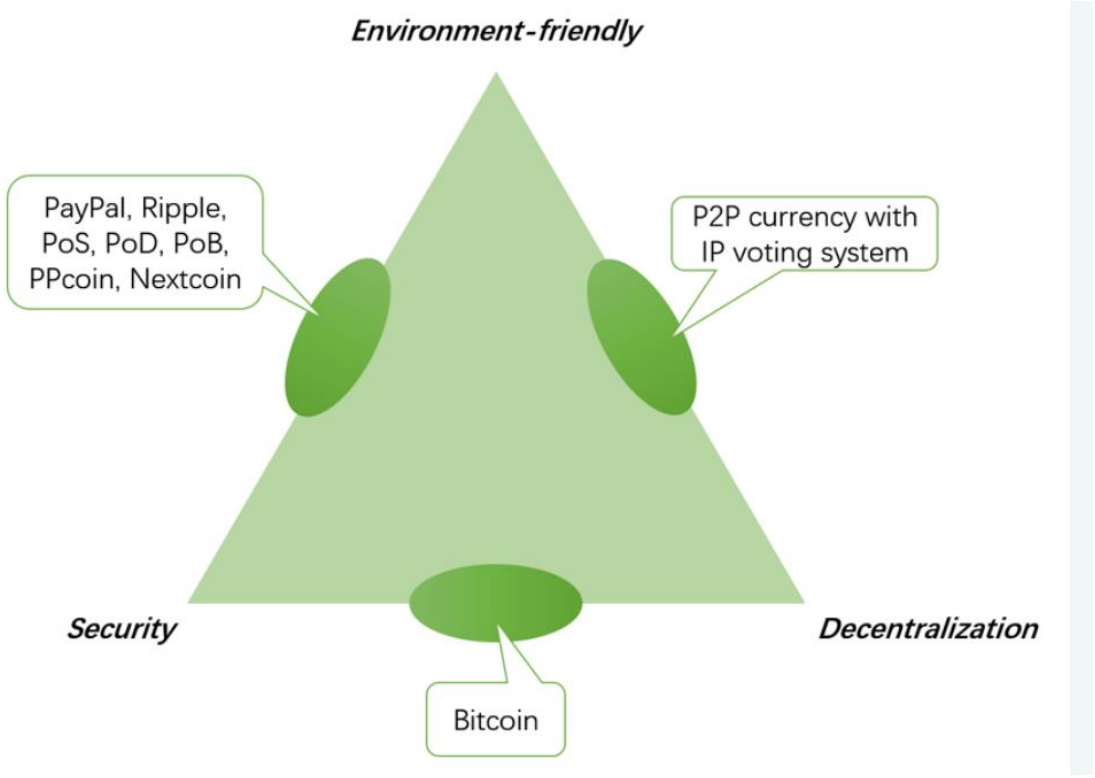
1.3 Current State of Public Chains

Public chains, refer to consensus blockchains that anyone worldwide can read, send transactions, and receive valid confirmations.

Public chains offer significant advantages, such as transparency, anonymity of

data, queryability, and traceability of on-chain data. In essence, public chains are blockchains in which anyone, without authorization, can participate. Compared to private chains and consortium chains, public chains are the most decentralized, widely used, recognized, and popular blockchains.

However, with rapid advancements in public chain technology, certain issues like the "impossible triangle" have emerged. Here are some primary development obstacles currently facing public chain technology.



1.3.1 Scalability Issue

Due to the decentralized nature of blockchain systems, the throughput and performance of public chains have always been limiting factors. As blockchain applications continue to expand, public chains require higher throughput and better performance to support more transactions and users.

1.3.2 Privacy Protection Issue

On public chains, all transactions and information are public, lacking privacy protection mechanisms. This limits the scope and use cases of public chains, necessitating more efforts in privacy protection.

1.3.3 Development Issue

Current application development on public chains primarily relies on smart contracts. However, writing and debugging smart contracts still require a certain level of technical expertise, limiting the practical development and efficiency of applications.

1.3.4 Ecological Issue

Ecological development is a crucial aspect of public chain growth. However, there are challenges in the current state of public chain ecosystems, such as incomplete ecosystem construction and a shortage of developers, impeding the healthy development of the public chain ecosystem.

1.3.5 Cross-Chain Interoperability Issue

Interacting with data and assets between different public chains still presents challenges, requiring further enhancement of cross-chain interoperability technology.

1.3.6 Governance Issue

Governance is pivotal for the ecosystem of public chains. Existing governance mechanisms need further refinement to ensure the long-term development and stable health of public chains.

1.3.7 Decentralization Issue

Distributed nodes form the fundamental basis of public chains, and an increasing number of nodes signify a broader consensus on the public chain. However, some public chains have compromised their level of decentralization to boost transaction speeds. For example, BNB Smart Chain has faced persistent questions about its centralization.

To address these challenges, we introduce Canto Public Chain, aiming to resolve the "impossible triangle" dilemma and achieve a significant leap in scalability, privacy protection, development prospects, ecological construction, cross-chain interoperability, and governance aspects.

2 Canto Smart Chain Overview

2.1 Introduction to Canto

Canto Smart Chain, or simply Canto, is a decentralized public blockchain infrastructure launched by the Canto DAO Foundation. It stands as a leading global decentralized public chain tailored for DeFi applications. Canto is a Layer1 project based on the Cosmos SDK and utilizes the Tendermint consensus mechanism, while also being compatible with the Ethereum Virtual Machine (EVM).

The Cosmos SDK serves as an architecture that empowers developers to easily create customized blockchains. Leveraging the open-source modules provided by Cosmos SDK, developers can select and assemble the required modules to construct their desired blockchain according to their specific needs.

EVM compatibility enables seamless integration between Ethereum tools, DApps, and Canto. Any developer can build decentralized DApps on the Canto Smart

Chain, spanning various application domains such as liquidity mining, DeFi protocols, anonymous exchanges, lending, cross-chain transactions, NFTs, social networks, payments, entertainment, e-commerce, and more.

Canto is poised to establish a peer-to-peer trust system, eradicating intermediaries' interference in commercial scenarios. It aims to create a new cryptocurrency system, payment method, and credit mechanism while fostering an efficient, low-cost, and more secure value ecosystem.

2.2 Canto Use Cases

2.2.1 For Users

Canto's primary objective is to establish a highly compatible, low-cost, efficient, and secure decentralized public chain. Any user can access a range of DeFi products, NFTs, and GameFi projects by holding the native $CANTO tokens. Examples of potential use cases include:

- ✓ Using $CANTO to pay gas fees for minting and exchanging NFTs in the NFT market.
- ✓ Engaging in lucrative GameFi opportunities and interacting with the growing blockchain gaming community.
- ✓ Participating in DEX to trade tokens and speculate on their value.
- ✓ Accessing advanced financial tools such as staking, lending, and liquidity mining.
- ✓ Participating in upcoming metaverse revolutions through NFTs supported by Canto.
- ✓ Engaging in DAOs and contributing to the entire community.

2.2.2 For Developers

Canto employs the Cosmos SDK, a feature-rich platform for building dApps.

Developers can create their unique blockchains using the modular framework of the SDK, streamlining the development process. This simplified approach enables the creation of custom chains derived from Cosmos SDK, thereby promoting the adoption and growth of the blockchain ecosystem. Additionally, the IBC protocol facilitates seamless communication across different blockchains. Examples of use cases for developers include:

- ✓ Developing standard tokens and non-fungible tokens (NFTs).
- ✓ Creating DApps, such as trading platforms and lending platforms.
- ✓ Participating in proposals and governance by holding $CANTO tokens.
- ✓ Staking $CANTO to deploy their own nodes and validate block transactions.
- ✓ Developing and operating blockchain-based games and metaverse-related projects.

2.3 Key Features of Canto

Compared to other emerging chains, Canto is an EVM-compatible public chain developed on the Cosmos SDK. It emphasizes high decentralization, eschewing investors and foundation establishment in favor of relying on the community. It also provides free public infrastructure (FPI) and offers contract revenue sharing (CSR) for developers.

Unlike Bitcoin and other blockchain systems primarily focused on payments and store of value, which may not generate the same demand as platforms supporting smart contracts, Canto's ability to enhance the productivity of the Web3 ecosystem promises to increase block space demand. This event will also contribute to the increased demand for the native $CANTO token.

With its high throughput and decentralization capabilities, Canto token users won't face many concerns of PoW tokens (including slow transaction speeds, severe network congestion, POW mining, and high gas fees). Additionally, due to the

Tendermint consensus architecture, Canto will maintain a high level of decentralization and reliable cross-chain interoperability.

In summary, the key features of Canto are realized based on the following fundamental principles:

- ✓ **Tendermint Consensus Mechanism:** Tendermint employs PBFT + PoS as its consensus mechanism, which is more efficient and secure compared to Bitcoin's Proof of Work (PoW) mechanism.
- ✓ **EVM Compatibility:** Existing Ethereum smart contracts can be easily migrated to Canto without any modifications.
- ✓ **Decentralized Governance:** Community members (token holders) can propose, delegate, and vote to influence governance decisions related to blockchain parameters and events.
- ✓ **Cross-Chain Compatibility:** Through the IBC cross-chain communication protocol, Canto can engage in cross-chain communication with any blockchain, eliminating the need to trust intermediaries. IBC can be used not only for blockchains developed based on the Cosmos SDK but also for other blockchains such as Ethereum, Polkadot, and more.

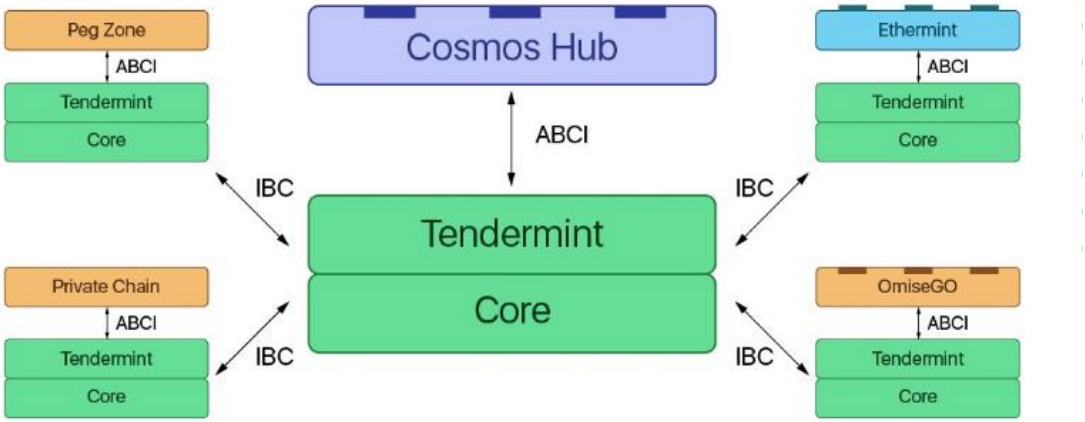
3 Canto Technical Model

3.1 Tendermint Consensus Mechanism

Distributed consensus algorithms can generally be categorized into two types: Byzantine Fault Tolerant (BFT) and non-Byzantine Fault Tolerant. Non-BFT algorithms like Paxos and Raft have been widely used in current distributed systems, while the practical application scope of BFT algorithms is relatively smaller (especially before the advent of blockchain). Tendermint falls under the category of

BFT algorithms, optimizing traditional PBFT algorithm, requiring only two rounds of voting to achieve consensus. Currently, the Tendermint algorithm is primarily applied in blockchain systems.

Tendermint aims to provide developers with the network and consensus layers of a blockchain, serving as a platform for developing various decentralized applications. This enables developers to focus solely on the application layer of the blockchain, without having to simultaneously develop the consensus and network layers. More importantly, Tendermint is responsible for sharing block transactions between nodes and autonomously establishing an immutable transaction order, facilitating the establishment of a Proof of Stake (PoS) consensus network and significantly reducing development complexity.



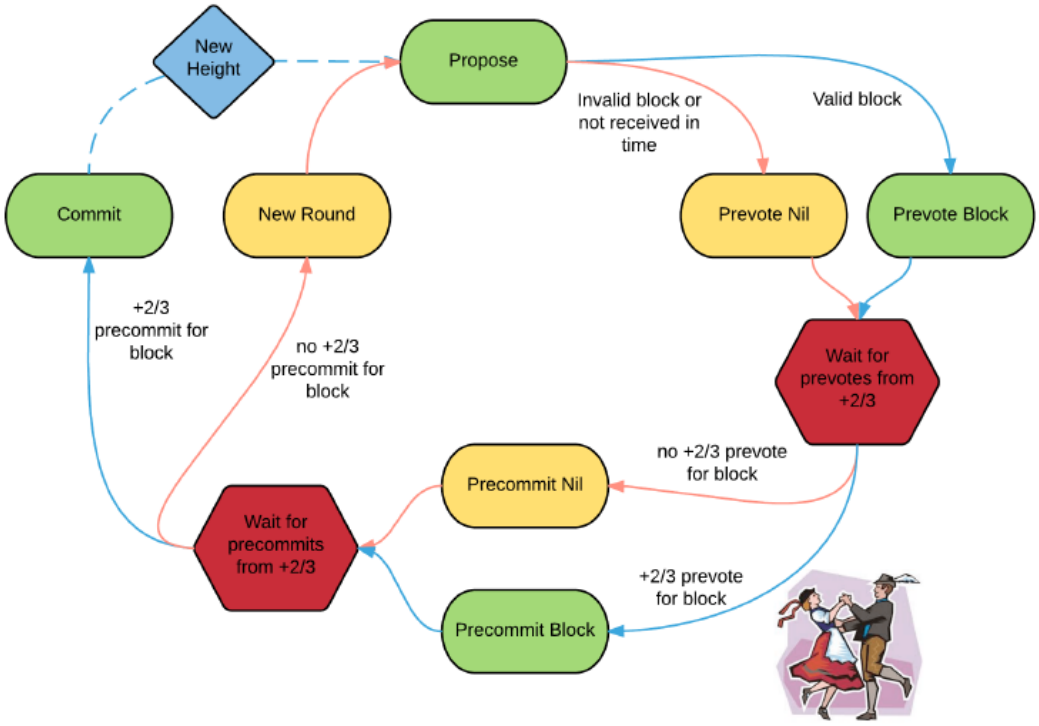
Imagine you're a chef making a pizza. First, you create the pizza crust, and then you can add any toppings you desire, such as beef, chicken, tomatoes, or cheese. In this analogy, Tendermint is like the pizza crust, while blockchain networks like Cosmos form the base beneath the crust. Developers take on the role of chefs, creating various blockchain applications on top of the base and crust (Cosmos-Tendermint) as they see fit.

3.1.1 System Model

In the system, nodes are categorized into two types: validator nodes and non-validator nodes. Validator nodes participate in consensus, which involves reaching an agreement on blocks (containing batches of transactions). This participation includes proposing blocks and voting on proposed blocks. Non-validator nodes, on the other hand, do not take part in consensus but assist in propagating block and voting messages, as well as synchronizing states among themselves.

Nodes do not necessarily have direct connections with each other. Those nodes directly connected to a particular node are referred to as peers. Whether validator or non-validator nodes, they all maintain certain consensus-related states, such as the current blockchain height, round, and step.

3.1.2 Algorithm Model



Tendermint consensus algorithm operates in rounds, where each round is composed of multiple steps. The algorithm ensures that validators propose new blocks and reach consensus on a single block in each round. The key steps in the algorithm model are as follows:

1)Propose: During this step, a validator proposes a new block containing a batch of transactions. The proposed block includes a unique round identifier and a cryptographic signature from the validator.

2)Prevote: Validators receive the proposed block and cast their prevote votes, indicating their agreement with the proposed block. A prevote consists of the proposed block's hash and the validator's signature.

3)Precommit: Once a validator receives enough prevote votes from other validators for the same block, it moves to the precommit step. In this step, validators commit to a single proposed block by casting precommit votes.

4)Commit: When two-thirds of the validators have precommitted for the same block, the block is considered committed. In the commit step, validators finalize their agreement by broadcasting precommit votes for the same block.

5)Block Addition: Once a block is committed, it is added to the blockchain. All validators now move to the next round to propose and reach consensus on the next block.

The Tendermint algorithm ensures that at least two-thirds of the validators agree on the same block within a round. This consensus process ensures the security, finality, and consistency of transactions in the network.

3.1.3 Rewards and Earnings

All Canto validator nodes within the current set of validators will earn rewards from transaction fees paid in $CANTO. Since $CANTO also serves as a utility token for other applications, both delegators and validator nodes will continue to benefit from holding $CANTO tokens.

Validator node earnings come from the transaction fees collected in each block. Validators can decide how much of the earnings from fees collected in $CANTO they want to share with their delegators to attract more staked investments. Each validator node takes turns producing blocks with the same probability (assuming they remain 100% online), which means that over the long term, all stable validator nodes might achieve similar-scale earnings.

Additionally, the staked assets of each validator node may differ in size, leading to a counterintuitive situation where more users trust and delegate stake to the same validator node, potentially resulting in lower earnings for them. Therefore, as long as validator nodes remain trustworthy (as untrusted validator nodes could pose significant risks), rational delegators tend to delegate to nodes with lower staked amounts.

Ultimately, the distinctions between all validator nodes become smaller. This effectively prevents concentration of staking and the issue of "the rich get richer," as seen in other networks.

3.2 Cryptographic Hash Function

A fundamental tool in blockchain technology is the cryptographic hash function (also known as a hash function), which ensures the integrity and immutability of transactions. A hash function is a mathematical algorithm that produces a fixed-size

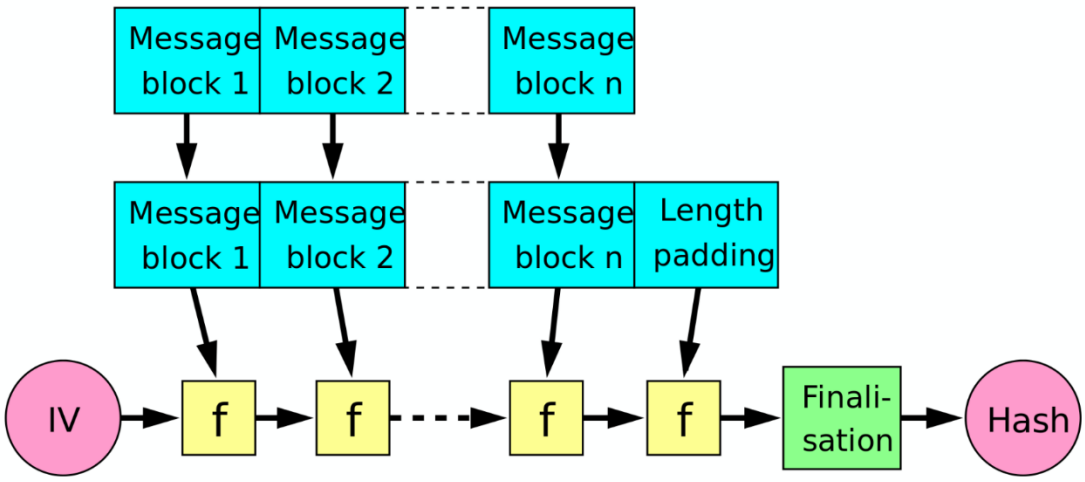
numerical output (referred to as a hash) based on input data. More specifically, a hash function can be represented as:

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

The hash function takes input of any size and generates a fixed-length numerical output of size k, known as the hash value. It must satisfy the following properties:

- ✓ It can easily compute H for any input data.
- ✓ For any h, it is computationally infeasible to calculate an input x such that $H(x) = h$.
- ✓ For any x, it is computationally infeasible to find y such that $H(y) = H(x)$ and $x \neq y$.
- ✓ Finding any (x, y) such that $H(x) = H(y)$ and $x \neq y$ is computationally infeasible.

SHA-256 and Keccak-256 are widely used in multiple blockchains and produce hash outputs of size 256 bits.



3.3 Digital Signatures

3.3.1 Secp256k1 Curve

Noticing that all elliptic curves are defined in the form of y^2 = x^3 + ax + b, Secp256k1 is an elliptic curve algorithm used by multiple blockchains to implement public-private key pairs.

For instance, we can define Secp256k1 with a=0 and b=7 (i.e., secp256k1 exists on the equation y^2 = x^3 + 7). Before a user generates a public-private key pair (pk, sk), they must first generate a sufficiently large random number (to be used as sk) and multiply it by the generator point G to obtain the private key (sk.G, used as public key pk).

We use this number to define a point on the secp256k1 curve. Due to the underlying discrete logarithm problem (DLP), no one can derive the private key from a given public key and generator point (as long as the key size is sufficiently large).

Note that for each x value in this equation, the y component is squared, resulting in two symmetric points on the x-axis. As a result, there are two y values referred to as odd and even for each x value. Therefore, the public key can be identified based on the parity of the x-coordinate and the y-coordinate. In the blockchain field, this property is essential as it saves a significant amount of data storage space.

3.3.2 ECDSA Signature Algorithm

The Elliptic Curve Digital Signature Algorithm (ECDSA) is an encryption algorithm used to create digital signatures. Specifically:

Setup:

- **Public Parameters:** Let F_q be a finite field, a and b be two parameters, define

an elliptic curve C over F_q, a validation seed C, a prime integer n > 2^555, and a point G of order n on C, where q is either a prime or a power of 2, 2^m.

- **Private Key:** An integer d in the range [1, n−1].
- **Public Key:** $Q = dG$.

To generate a signature for a given message M:

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

  $(x_1, y_1) = k\mathrm{G}$

  $r = x_1 \bmod n$

  $s = \frac{H(M) + dr}{k} \bmod n$
- If $r = 0$ or $s = 0$, retry. The signature is (r, s).
- Signature: (M, r, s).

To verify:

- Given (M, r', s').
- Verify that r' and s' are in the range $[1, n-1]$, and:

$$r' = x_1 \bmod n \text{ for } (x_1, y_1) = u_1 G + u_2 Q$$

$$u_1 = \frac{H(M)}{s'} \bmod n$$

$$u_2 = \frac{r'}{s'} \bmod n \; s'$$

The ECDSA algorithm involves generating a digital signature for a given message by using a private key and verifying the signature using the corresponding public key. It relies on the properties of elliptic curves to ensure the security and authenticity of the signature.

3.4 IBC Cross-Chain Communication

Cross-chain communication is a crucial foundation for enabling asset transfers

between different public blockchains, and it serves as an essential tool for Canto to connect with other public blockchains. The unique aspect of the IBC protocol is its layered design, which consists of the transport layer and the application layer, working together to facilitate the entire workflow.

The Transport Layer (TAO: transport layer) provides the necessary infrastructure to establish secure connections and verify data packets between blockchains. The Application Layer (application layer) builds upon the transport layer and defines how data packets should be packaged by the sending chain and unpacked by the receiving chain.

A simple analogy for understanding this concept is a mail delivery system. When you send a letter to someone through postal services, the postal service takes the envelope containing the letter and places it in the recipient's mailbox. The recipient then opens the envelope and reads your letter. The transport layer of IBC can be likened to the postal service. The postal service doesn't concern itself with the content of the letter; it simply executes the action of collecting the envelope from point A and delivering it to point B. The envelope itself can be seen as an IBC data packet being sent from one chain to another.

Just like you write the recipient's address on the envelope, the IBC data packet contains sender and recipient information. Finally, the recipient (application) receives the letter (data packet), opens it, and reads the content.

In the context of blockchain networks, IBC provides a standardized protocol for different chains to communicate and exchange information, which is crucial for enabling interoperability and expanding the capabilities of decentralized applications across various blockchains.

3.4.1 Canto Repeater

In the context of IBC, blockchains do not directly send messages to each other over the network. Instead, they rely on relayers to facilitate communication. Relayers are off-chain processes responsible for monitoring the state of each chain participating in the IBC protocol and relaying updated data packets to their corresponding counterpart chains. Using the example mentioned earlier, when Chain A sends a message X to Chain B, Chain A will submit or store the hash of the data packet containing message X in its state machine. When a relayer sees that Chain A has submitted a message X intended for Chain B in its state machine, the relayer simply picks up message X and forwards it to Chain B.

Relayers are responsible for transmitting data packets back and forth. They do not modify or validate data packets and therefore do not need to be trusted. Relayers are also used during the establishment of connections and channel handshakes. In case one end of a connection attempts to fork or engage in malicious behavior, relayers can submit evidence of such behavior.

3.4.2 Light Clients

Light clients are responsible for verifying the proof of messages within data packets. A light client is a lightweight alternative to running a full node. Unlike full nodes, it doesn't store all block data or execute transactions. Instead, it only verifies block headers. In the context of IBC, a light client is a validation algorithm within a blockchain that tracks changes in the state of another blockchain (timestamp, root hash, next validator set hash). This approach saves space and increases the efficiency of processing consensus state updates.

3.4.3 Channels

Communication between applications in IBC is conducted through channels, which serve as conduits for transmitting data packets between application modules on different chains. A connection can have any number of associated channels. However, each channel is associated with only one Connection ID, which is used to identify the light client. Additionally, each channel has a Port ID, used to identify the application on the connected channel.

4 Canto Architecture Design

Canto is a blockchain built using the Cosmos SDK framework, compatible with the Ethereum Virtual Machine (EVM), and relies on its own set of validators. Leveraging the Cosmos SDK framework and EVM compatibility, our developer community can construct a blockchain network that better suits their needs.

4.1 Canto function

Built on the architecture of Cosmos SDK and EVM compatibility, Canto offers the following features:

4.1.1 Utilization ETH Contract Technology

- ✓ Users can interact with standard wallets through JSON-RPC.
- ✓ Developers can utilize Solidity/Vyper programming and benefit from comprehensive EVM support.
- ✓ Access to popular Ethereum tools, development tools, and code libraries.
- ✓ Optimized user experience when executing cross-network transactions.

4.1.2 Inter-Network Communication

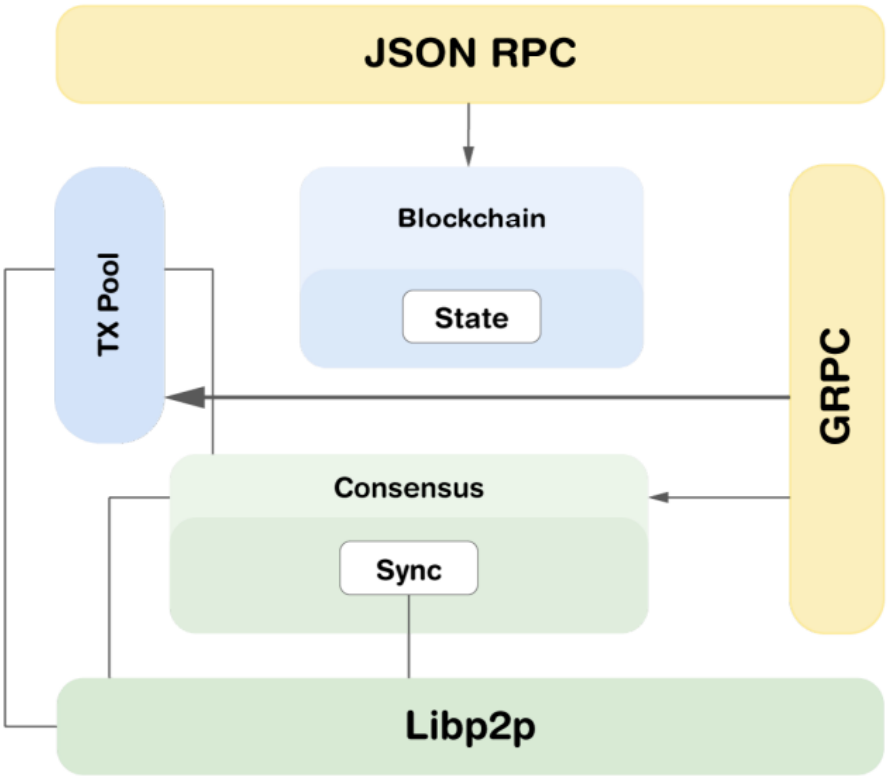
- ✓ Fully trusted and decentralized embedded Ethereum cross-chain bridge solution.
- ✓ Transfer assets from any EVM-compatible network, particularly BNB Chain and Ethereum mainnet.
- ✓ Transfer ERC20 tokens, NFTs, or native tokens from public chains.
- ✓ Customization of bridge functionality using existing plugins.

4.1.3 Special Features

- ✓ Construct network usability through development plugins.
- ✓ Capability to replace core functionality using consensus plugins.
- ✓ Surpass Ethereum smart contracts by incorporating Runtime into the system.

In summary, due to its EVM compatibility, Canto achieves full compatibility with Ethereum smart contract technology. Additionally, it ensures network decentralization, security, and scalability using PoSA (Proof of Stake Authority).

4.2 Canto Layered Architecture



- ✓ **Libp2p:** Libp2p serves as the underlying network layer, characterized by modularity, scalability, and speed, providing an excellent foundation for higher-level functionalities.
- ✓ **Synchronization & Consensus:** Separation of synchronization and consensus protocols enables the implementation of customizable synchronization and consensus mechanisms, depending on how the client operates.
- ✓ **Blockchain:** The blockchain layer functions as the core layer for managing tasks within the Canto system.
- ✓ **State:** The state layer provides the logic for state transitions, handling how the state changes when new blocks are added.
- ✓ **JSON RPC:** This layer serves as the API layer for dApp developers to interact with the blockchain.
- ✓ **TxPool:** The transaction pool layer is closely coupled with other modules in the system, as transactions can be added from multiple entry points.
- ✓ **gRPC:** Essential for operational interactions. This layer ensures that node

operators can easily interact with clients, providing a usable and efficient user experience.

4.3 Canto Testnet Network

The Canto Testnet is a blockchain network separate from the main network, operating as an exact replica of the original protocol, using the same technology and software to provide similar functionalities. Additionally, the Testnet simulation offers a sandbox environment that allows continuous refinement and improvement of the real-time version of the project before it is launched on the mainnet. This dynamic environment provides an ideal setting for testing smart contracts and decentralized applications (dApps).

While the mainnet is the ultimate and only valuable version of the cross-blockchain release, the Testnet also has its advantages, primarily offering an experimental environment to enhance the platform.

The experimental environment ensures ongoing development on the platform. To enhance and innovate the blockchain industry, continuous rigorous testing of smart contract functionality, transactions, and minting/mining processes is necessary. Having a Testnet enhances the spirit of development across the entire industry by providing a simulation of how the actual mainnet will operate after launch.

In short, the Canto Testnet is a crucial component of the Canto ecosystem, providing a straightforward and risk-free way for developers to test and debug their code. Whether it's writing smart contracts, building DApps in NFTs, or creating DeFi protocols, the Canto Testnet network adds significant value to the blockchain mainnet it serves.

Based on the consensus above, a test network will be provided for developers to test before the Canto mainnet goes live. Simultaneously, a testnet faucet will be created to distribute test coins—tCANTO—for developers to use.

5 Canto Smart Contracts

Canto utilizes smart contracts to manage the selection of validator nodes, reward distribution, and staking. These contracts are deployed within the genesis block and encompass six different types of smart contracts.

- ✓ **Governance Contract** - Manages validator node proposals and voting.
- ✓ **Validator Set Contract** - Ranks validator nodes and determines which nodes are elected or removed.
- ✓ **Vault Contract** - Receives all withdrawal fees from the chain bridge.
- ✓ **Staking Contract** - Manages staking operations and allocation of block rewards.
- ✓ **Slashing Contract** - Manages validator nodes that do not adhere to chain consensus rules.
- ✓ **Bridging Contract** - Manages token exchanges between the Canto blockchain and BNB Chain.

5.1 Governance Contract

Blockchain networks are autonomous platforms that self-evolve through peer-to-peer democratic community interactions, promoting transparency. On-chain governance is a method for suggesting and implementing changes to the blockchain. In this type of governance, change initiation rules are typically hard-coded into the blockchain protocol.

Community-elected validator nodes propose potential ideas through code updates and written proposals. All elected validator nodes and regular users participate in

voting to accept/reject proposed changes.

Under the governance contract, community members democratically vote to propose ideas that advance the development of the blockchain network. To be able to make proposals, users must hold a sufficient amount of $CANTO token stakes. On the other hand, individuals holding a certain amount of $CANTO tokens can vote on proposals. There will also be an option to report governance misconduct for reporting abuses of the contract. The following example options might change based on community feedback:

- ✓ Minimum staking amount to become a validator node
- ✓ Minimum staking amount for regular users
- ✓ Minimum staking amount for proposing ideas

5.2 Validator Set Contract

This contract validates and stores nodes that meet the requirements to become validators. Additionally, the contract lists the primary validators and their addresses, finalizes and approves blocks, and categorizes blocks generated by specific validators.

5.3 Vault Contract

All withdrawal fees from the cross-chain bridge are sent to the vault contract.

5.4 Staking Contract

This contract executes staking, reward calculation, and distribution of rewards to users and validators. The contract also periodically updates rewards received by validators and stakeholders.

The consensus mechanism ensures decentralization and community participation. $CANTO holders, including validators, can stake their tokens to the contract to

receive $CANTO.

5.5 Slashing Contract

Canto adopts a punishment method similar to BNB Chain's. Besides enhancing the security of the Canto chain, it is also used to protect its on-chain governance mechanism from malicious or dishonest behavior. Evidence of punishment on the Canto chain can be submitted by anyone. It's important to note that each submission of a slashing proof requires a transaction and incurs a corresponding fee. However, if successful, it also results in higher rewards.

The following two types of slashing behaviors are considered:

- ✓ **Double Signing:** Assuming two different block headers have the same height and parent block hash. If these two block headers are sealed and signed by the same validator with different signatures, the validator will be penalized and permanently jailed.
- ✓ **Unavailability:** If a validator misses 48 blocks within 24 hours, they won't receive rewards from block fees. If a validator misses more than 96 blocks within 24 hours, they will be fined a certain amount of $CANTO tokens and restricted for 3 days. During the restriction period, they can still generate or validate blocks.

5.6 Bridging Contract

Stakeholders can invoke the bridging contract for cross-chain token transfers on EVM chains. When a transaction is synchronized, multiple nodes (bridge signers) will sign and confirm the transaction, and invoke the bridging contract to encode the data. When more than half of the nodes confirm (through the digital signature process), the relevant tokens will be returned to the specified address.

6 Canto Ecosystem

6.1 Developer Ecosystem

6.1.1 Blockchain Explorer

The blockchain explorer serves as the gateway into the Canto data realm. With the Canto blockchain explorer, technical developers can access real-time information about blocks, transactions, miners, addresses, gas fees, smart contracts, and other on-chain activities. It also offers synchronized data on all Canto nodes, enabling an in-depth understanding of transaction details on the Canto chain.

6.1.2 IDE Development Environment

As an EVM-compatible chain, Canto integrates the Remix smart contract online Integrated Development Environment (IDE) tool. Developers can use Remix for a variety of technical operations, including:

- ✓ Developing smart contracts (with an integrated Solidity language editor)
- ✓ Dynamic smart contract debugging
- ✓ Accessing the state and properties of deployed smart contracts
- ✓ Code analysis, error prompts, and best practice suggestions
- ✓ Debugging and testing DApps (requires tools like Mist or other Web3-enabled solutions)

6.2 Defi Ecosystem

6.2.1 Canto DEX

Canto DEX, developed by the Canto team, is an Automated Market Maker (AMM) DEX. It employs constant and constant product formulas for stablecoins that

require concentrated liquidity and assets that require infinite liquidity, respectively.

Additionally, Canto DEX is designed to be fee-less and permanently un-upgradeable, where liquidity providers are rewarded with native $CANTO tokens as opposed to transaction fees.

Canto DEX allows users to trade tokens without the need for centralized exchanges. Any transactions made on the Canto DEX are directly executed from your wallet. There's no need to trust intermediaries, as your tokens are not handled by any third parties during the trading process.

6.2.2 CANTO Bridge

The CANTO Bridge enables token bridging between Canto and other EVM-compatible chains. Instead of directly exchanging or converting assets, the CANTO Bridge locks your source asset in a smart contract and mints new wrapped assets on the target chain. Subsequently, these wrapped assets can be exchanged for other assets on the target chain through exchanges.

The underlying technology of the CANTO Bridge ensures transfer speeds similar to regular transactions on the target blockchain. Cross-chain transfers involve two transactions: one on the Canto blockchain and the other on the target chain. For the CANTO Bridge, cross-chain waiting time is determined by the confirmation time required on the target blockchain. With sufficiently fast confirmations, cross-chain transactions can be nearly instantaneous.

6.2.3 Canto Lending Market (CLM)

CLM is a lending market forked from Compound v2, with governance controlled by $CANTO stakers. LP tokens obtained from providing liquidity on the Canto DEX can be used as collateral to borrow other assets in CLM, but the LP tokens themselves cannot be borrowed.

For deposits, depositing on the CLM platform is similar to placing funds in a bank. Depositers lock their crypto assets in a smart contract and earn interest generated from it. Additionally, depositors can withdraw their principal and interest from CLM at any time.

For borrowers, borrowing from CLM requires over-collateralization of supported tokens, which then provides a borrowing limit for lending out other tokens. The over-collateralization significantly reduces the risk of default. Once borrowers repay the loan along with interest, their locked collateral will be automatically returned. However, due to price fluctuations of the collateral assets, if their price falls below a certain threshold compared to the loan amount, borrowers may need to add more collateral or risk automatic liquidation by the smart contract. In the case of liquidation, borrowers would still hold the borrowed assets but lose their collateral.

6.2.4 Decentralized Stablecoin

In times of high price volatility, stablecoins offer value preservation without exiting the cryptocurrency market. Typically, stablecoins are issued through asset-backed mechanisms, such as USDT and TUSD, which are pegged to the US dollar using assets like USD as collateral.

However, with increasing centralization risks, there's a need for a decentralized stablecoin to meet the demands of the DeFi world. $NOTE is an over-collateralized stablecoin within the Canto ecosystem, obtainable only through borrowing from CLM. Currently, it is backed by $USDC and $USDT assets.

The price of $NOTE is controlled by a smart contract, and its interest rate is periodically adjusted algorithmically to keep its price stable around $1. Therefore, $NOTE's price is co-integrated with the US dollar but not directly pegged to it. Serving the Canto ecosystem, $NOTE primarily functions as a "soft-pegged" stablecoin to the US dollar.

6.3 Free Public Infrastructure (FPI)

Canto has made improvements at the application layer: it develops these infrastructure-level protocols, known as Free Public Infrastructure (FPI), and designates them as public utility protocols. Specifically, Canto's public chain has integrated a lending market forked from Compound, a DEX forked from Solidity, and a stablecoin called NOTE.

These established "public utilities" are naturally meant to serve the public. The DEX protocol of Canto cannot be upgraded and is not subject to control. It will run permanently on Canto without future fee increases. The lending market of Canto is governed by Canto stakers. Given the emphasis on the ecosystem value of the entire chain, Canto holders are not inclined to extract additional profits from individual applications. As for the stablecoin $NOTE, the protocol does not charge any fees.

Most importantly, these core public infrastructures will not have governance tokens, preventing the possibility of future rent-seeking from users. Additionally, the core protocols will follow the principle of "minimum user capture," not featuring user

interfaces. Users can only interact with the protocols through third-party aggregators, minimizing the impact of centralization.

Canto believes that existing DeFi protocols are more akin to fee-based private parking lots serving their own communities. Canto's FPI, on the other hand, resembles a free roadside parking open to everyone.

6.4 Contract Revenue Sharing (CSR)

Contract Revenue Sharing (CSR) is a fee-sharing model within the Canto network that allows contract developers to extract a certain percentage of the transaction fees paid by users when interacting with their contracts, thereby earning revenue. Contract developers registered as CSR holders are entitled to receive some NFTs.

As contract revenue accumulates, holders of CSR NFTs can claim this revenue at any time. The NFTs themselves can be traded and composited within DApps. This composability offers a variety of use cases, including trading, bundling, investing, loan collateral, and more.

6.5 Potential Applications

6.5.1 NFT Ecosystem

Canto will offer users the ability to mint their own NFTs following the ERC-721 standard. Since this well-established NFT standard is widely accepted in the market and metaverse, Canto NFT owners will be able to integrate their NFTs into existing NFT ecosystems.

6.5.2 Gamefi Ecosystem

Canto will provide developers with the capability to build entire virtual worlds and blockchain games on the Canto smart contract framework. As a result, the $CANTO cryptocurrency will enable users to engage in the virtual gaming economy and share their digital assets within their favorite metaverse.

7 Token Economics

$CANTO is the native token of the Canto network, serving as a token to incentivize users and third-party partners to engage in ecosystem development and other activities. It possesses internal value, resources, and rights within the Canto smart chain that are exchangeable. Additionally, it functions as the fuel fee for transactions and development, and is designed for limited issuance. $CANTO is used to pay gas fees for transactions and can also be staked with validators to help secure the network.

Furthermore, as an infrastructure that integrates various digital assets, the Canto smart chain has the potential to derive more intelligent assets through financial smart contracts. In the future, Canto aims to drive the value growth of the $CANTO token through various innovative models.

7.1 Issuance Plan

- ✓ The issuance of $CANTO will occur in 30-day cycles.
- ✓ In the first cycle, approximately 16 million $CANTO tokens will be minted at an annual rate of 19.84%.
- ✓ Subsequent cycles will see a 35% reduction in issuance each cycle, gradually approaching zero.

7.2 Token Allocation

- ✓ Name：$CANTO
- ✓ Initial Supply：1000000000
- ✓ Blockchain：BNB Smart Chain
- ✓ The token allocation plan is as follows:

Proportion	Allocation
45%	Long-Term Liquidity Mining
35%	Mid-Term Liquidity Mining
13%	Contributors
5%	Future Network Grants
2%	Airdrop to Early Active Users
Leftover	Reserved in Community Pool

7.3 Token Staking Rewards

Canto's smart chain enables users to enjoy the borderless DeFi infrastructure at low costs. This infrastructure is supported by a group of 21 community validators. These validators are responsible for processing transactions, providing computational power and hardware, and maintaining network security. In return, they receive $CANTO token rewards from transaction fees and block rewards.

Validators need to operate a hardware node that meets specified requirements, run a full Canto node, and stake a minimum of 999 $CANTO tokens. However, even meeting these requirements only gets them into the pool of validator candidates.

To truly begin producing blocks, validator candidates need to be elected as active validators. The elected validators are the top 21 validator candidates with the highest

voting power. Through a continuous election process, validators are updated every 24 hours.

7.4 Token Use Cases

The $CANTO token operates within the Canto ecosystem, similar to how ETH functions on Ethereum, making it the "native token" of Canto. This means that besides being used to pay for most fees on the Canto chain, it can also be used for:

- ✓ Paying "fees" for deploying smart contracts on the Canto chain.
- ✓ Paying Gas fees during transactions and transfers.
- ✓ Staking $CANTO to vote for a node and receive corresponding rewards.
- ✓ Performing cross-chain operations, such as transferring token assets between other chains and Canto.
- ✓ Participating in proposal governance by staking a certain amount of $CANTO tokens.

7.5 Token Value Interpretation

- ➢ From a value perspective, the $CANTO token embodies the carriers of both "trust value" and "consensus value."
- ➢ From an incentive perspective, the $CANTO token serves as an economic reward to incentivize "validators" to participate in the network.
- ➢ From a governance perspective, the $CANTO token acts as proof of stake in participating in the Canto smart chain network.
- ➢ From a security perspective, the presence of value incentives enhances the network security of the $CANTO smart chain.

7.6 Token Ecosystem Cycle

Built upon the Canto smart chain, various applications will be derived, such as mining wallets, DEX exchanges, blockchain payments, and more. Simultaneously, the $CANTO token can be exchanged with all digital currencies, supporting circulation and payments in all aspects of the ecosystem, including payment and receipt, transfers, fiat trading, deposits, withdrawals, listing voting, STO gateways, currency distribution, lending, charitable activities, gaming, shopping, and more.

All circulation transactions are facilitated by the $CANTO token and settled in global fiat currencies. In addition to circulating within the Canto smart chain ecosystem, it will also circulate in third-party applications developed based on public chain technology, serving as the sole value token. This will accelerate the circulation of the $CANTO token, adding more attributes of circulation value to the scarce $CANTO token, thereby increasing overall value and price.

7.7 Other Standard Tokens

Similar to the ERC standards on the Ethereum chain or the BEP standards on the Binance Smart Chain, on the Canto smart chain, we have also developed the CRC standards, such as CRC20 and CRC721. Anyone can issue tokens based on these standards on the Canto chain and conduct fast transfers or transactions.

8 RoadMap

Phase 1: Genesis and Foundation

- ✓ Launch Canto Genesis
- ✓ Announce Token Allocation Details
- ✓ Conduct smart contract audits
- ✓ Bridge Over $12 million USD from Ethereum

- ✓ Announce Canto's First Hackathon
- ✓ Optimize Liquidity Mining Incentives
- ✓ Listing on CMC and CG

Phase 2: Network Upgrade and Expansion

- ✓ Implement Mainnet Upgrade to Version 3.0.0
- ✓ Proposals to Reduce 30% of $CANTO Issuance
- ✓ Pass Proposal to Sustain Liquidity Mining Rewards
- ✓ Launch the decentralized exchange
- ✓ Enhance Interoperability with Other Blockchains
- ✓ Partner recruitment and ecological layout

Phase 3: Community Empowerment

- ✓ Enable decentralized governance
- ✓ Increase international community building
- ✓ Integrate DeFi Applications and NFTs
- ✓ Expand Outreach and Awareness Internationally
- ✓ Develop New Protocols and Solutions
- ✓ Community members exceeded 10,000
- ✓ Collaborate with Regulatory Authorities

Phase 4: Sustainability and Impact

- ✓ Launch Innovative DeFi Products
- ✓ Strengthen global marketing strategy
- ✓ Conduct Workshops
- ✓ Forge a global alliance
- ✓ High-Level Community Autonomy
- ✓ Drive Sustainability Initiatives

9. Team and Financing

9.1 Team Members

Canto is a community project initiated by @scott_lew_is, and its team members mostly come from the cryptocurrency industry with years of technical experience.

@scott_lew_is is the co-founder of DeFi Pulse and Slingshot; @RobinWhitney_ was a core member of Acala and Karura; @0xzak was a co-founder of Slingshot and an advisor to DeFi Pulse.

9.2 Financing Information

In 2022, Canto successfully completed its first round of financing, raising a total of $5 million. The funding was provided by renowned cryptocurrency venture capital fund "Variant Ventures," which has an excellent investment track record in the blockchain and crypto space. Variant Ventures is well-known for its deep background and expertise in the crypto field, offering valuable support and guidance not only in investment but also in technology, strategic planning, and market expansion.

10 Disclaimer

This whitepaper has been prepared by the Canto Foundation and is provided for informational purposes only. It should not be considered as an offer, invitation, or solicitation to buy or sell or subscribe for any cryptographic tokens, securities, financial instruments, or any other rights or products. Before participating in any activities related to this whitepaper, you should consult your own legal, financial, business, or other professional advisors. Community members, project development team members, third-party development organizations, and service providers shall not

be held liable for any direct or indirect damages or losses resulting from the use of this whitepaper.

This whitepaper is provided for general informational purposes only and does not constitute a prospectus, offering document, securities offering, investment solicitation, or sale of any product, item, or asset (whether digital or otherwise). The information provided herein may not be comprehensive and does not imply any elements of a contract. The whitepaper does not guarantee the accuracy and completeness of information, nor does it provide any statement regarding the accuracy and completeness of information. If the whitepaper contains information obtained from third parties, the community and the team have not independently verified the accuracy and completeness of such information.

This whitepaper is a conceptual document describing the long-term development goals of Canto and may be subject to occasional modifications or replacements. There is no obligation to update the whitepaper and provide information beyond the scope of this whitepaper to the audience. Any part of this whitepaper does not constitute and will not constitute any offer from the community, distributors, and any sales teams (as defined in this agreement), nor should any statement in this whitepaper be used as a basis for any contract and investment decision.

Below are some terms and statements from this project's whitepaper. These terms and statements are intended to inform potential investors about the risks and responsibilities of this project, and they need to be carefully read and understood:

- ✓ Any statement in this whitepaper should not be considered as investment advice, and you should not base any investment decision on any statement in the whitepaper.
- ✓ You will voluntarily bear all relevant costs and compliance requirements and ensure compliance with applicable laws, regulatory requirements, and restrictions.

- ✓ You acknowledge, understand, and agree that the tokens may have no value, are not guaranteed or represented to have any value and circulation attributes, and cannot be used for speculative investments.
- ✓ The community and its affiliated companies and team members are not responsible or liable for the value, transferability, liquidity, or any market that offers $CANTO (through third parties or otherwise).

The above content also includes other terms and statements that need to be read and understood carefully, including all risks and uncertainties described therein, including financial, legal, and other risks. The community and its team do not make any statements, warranties, or commitments to any individuals or entities, nor do they assume any responsibility for the content of this whitepaper (including but not limited to completeness, timeliness, and reliability). It does not have legal binding force.