

Ontology Infrastructure White Paper

Version 2.0.0

2019/07

Abstract

1. Ontology is an infrastructure that supports multiple trust collaboration scenarios and will continue to extend various modules and protocols based on scenarios and application scopes of applications. This Infrastructure White Paper only describes Ontology's planning at the current stage and will be continuously updated according to the actual project process.

Through history, people have established trust from different dimensions and methods, for instances, via technology, legal system and communities etc. However, multi-source, multi-system and multi-method single-point trust collaboration will incur very high collaboration costs, hindering the depth and breadth of trust collaboration. Although internet technology changes rapidly, the pain points of trust still exist today, such as trusted source decentralization, data fragmentation, lack of individual roles, identity verification, and difficulty in identifying false information. During the collaboration process of society governance, economic collaboration, and financial service, there is a large amount of cost incurred by “trust” every day.

The decentralized and tamper-proof blockchain has built technology trust for specific scenarios from a certain mechanism. However, to integrate with more business scenarios in real world requires more integration mechanisms. How to construct a trust mechanism that combines diversified trust and integrative applications becomes the pursuit for the new “trust” infrastructure¹.

This White Paper focuses on the architecture and technology protocol of the Ontology infrastructure.

Table of Contents

1. Introduction.....	2
2. Glossary	4
3. OntologyArchitecture	10
3.1. Ontology Network Architecture.....	11
3.2. Distributed Ledger Technology	12
3.2.1. Ontlogy Consensus	12
3.2.2. Smart Contract.....	15
3.2.3. Distributed Ledger	17
3.2.4. Performance Solution	17
3.2.5. HydraDAO	19
3.3. Ontology Sharding Structure.....	20
3.3.1. Multi-Layer Cross-Shard Service	22
3.3.2. Cross-Shard Transactions	23
3.3.3. Function Shards.....	24
3.3.4. Business Shards	25
3.4. Ontology Chain Network Structure	25
3.4.1. Cross-Chain Basic Services.....	27
3.4.2. Cross-Chain Transactions.....	28
3.4.3. Chain Network Governance	29
3.4.4. Function Chain and Business Chain.....	30
3.4.5. Hyper-Converged Chain Network	30
4. Postscript.....	31
Contact Us	32

1. Introduction

This White Paper describes the multichain and multi-layer technology architecture of Ontology, covering the core layer and its chain network interaction protocol, Ontology basic function chain, and providing for tools for some service layers.

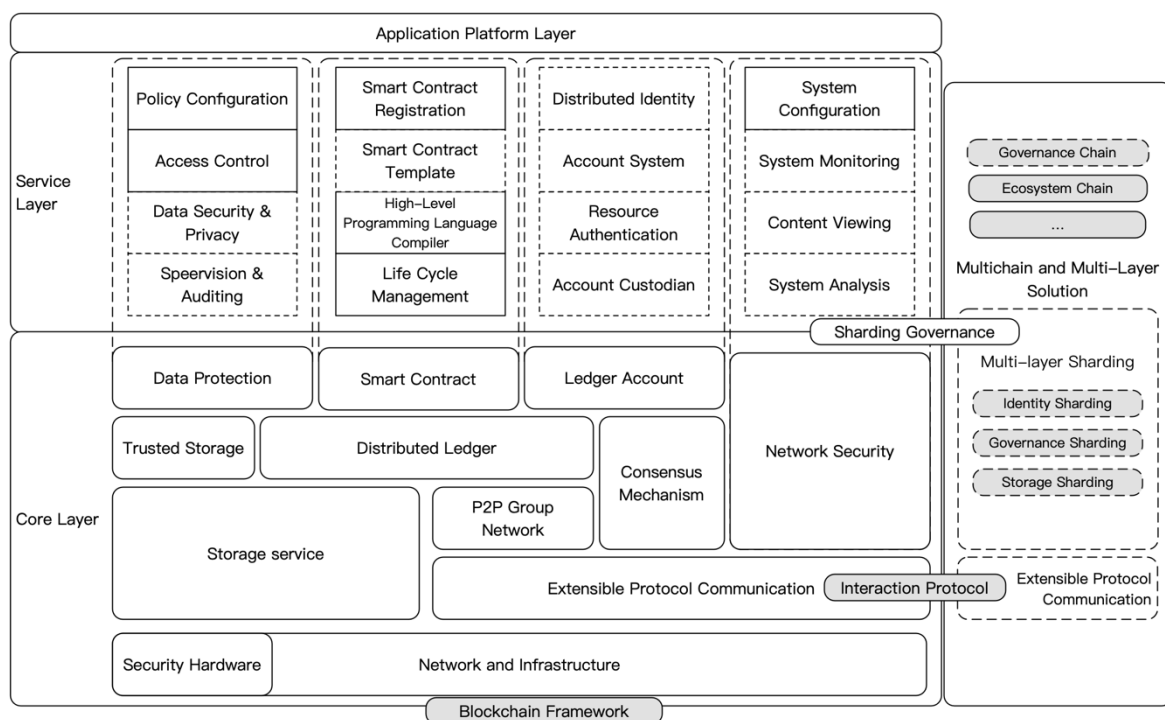


Figure: Ontology Infrastructure Technology Architecture

Ontology Core Layer. As the infrastructure of the Ontology ecosystem blockchain, the Ontology core layer provides a complete distributed ledger system, including core distributed ledger, smart contract system and security system. The decentralization, co-maintenance and tamper-resistance of distributed ledger technology serve as the key to realizing distributed multi-party trust. The

distributed ledger includes the implementation of consensus and smart contract system and provides consensus, storage and smart contract support for distributed trust framework and upper layer application. The Ontology core layer supports cross-chain and multi-layer sharding solution for Ontology homogeneous chains.

Ontology Service Layer. In order to make better use of the Ontology infrastructure services for upper layer application, Ontology conducts modularized service encapsulation for the Ontology core layer, including services like data protection, smart contract, ledger account, and network security.

2. Glossary

Ontology Chain Group

Also known as the “Ontology Chain Network”, it is formed by chains of entities based in different industries and regions, together they constitute the Ontology system. Each chain uses a separate distributed ledger and collaborates through interactive protocols.

Ontology Distributed Ledger

One or more core public service chains built by Ontology’s distributed ledger or blockchain framework, providing distributed ledger and smart contract system support to all type of services on Ontology.

Distributed Consistent Ledger

An incrementally modified data storage structure, maintained by nodes in a decentralized peer-to-peer network, featuring open data and tamper-proof historical data and providing trusted storage and smart contract support for Ontology.

Consensus

Data written into the ledger by ledger nodes according to specific protocols to ensure ledger consistency.

Smart Contract

Executable codes recorded in the ledger that is executed by the smart contract engine running on ledgers nodes. The input and output of each execution can be recorded on the ledger.

Entity

Individuals who interact and identified by ONT ID on Ontology.

Oracle

A service that provides blockchain with trusted external data. With Oracle, users can predict the result of the events outside the blockchain system, which will be permanently recorded on the blockchain as facts.

ONT

Ontology Token

ONG

Ontology Gas

Ontology Function Chain

Blockchains that provide certain function in the Ontology chain group, such as trusted execution environment (TEE).

Ontology Ecosystem Chain

Blockchain projects that use different governance model and participate in the Ontology ecosystem are collectively referred to as the Ontology Ecosystem Chain.

Ontology Governance Chain

The blockchain that is responsible for the overall governance, which is the Ontology main chain.

Cross-Chain

The blockchain technology that allows information and data to flow from one chain to another, and more often it's the blockchain technology that allows interaction and exchange of assets from one chain to another.

Side-Chain

The blockchain network that complies with side-chain protocol and exchanges information with the main chain through side-chain protocol in a blockchain network.

Homogeneous Blockchain

Blockchain network that runs the same blockchain protocol.

OCE (Ontology Consensus Engine)

Ontology Consensus Engine framework that supports various consensus protocols through modular encapsulation.

VRF

Verifiable random function, an encryption solution that maps an input to a verifiable pseudo-random output.

Network Node Operator

Ontology ecosystem participants who participate in the Ontology network node operation.

Synchronization Node

An Ontology ecosystem node that provides block synchronization and transaction request forwarding services in the Ontology network. Ontology synchronization nodes do not require staking.

Candidate Node

The network node operator who joins the Ontology ecosystem node by staking.

Consensus Node

The Ontology ecosystem node that is responsible for consensus block in the Ontology network. All consensus nodes are from Ontology candidate node and change with the switch of Ontology consensus round.

PoS

Proof of Stake. In a blockchain network based on the PoS consensus, all nodes that become “validators” are able to produce (or publish) blocks, the probability of which depends on the “stake” they have.

P2P

Peer-to-peer network, a distributed application architecture that distributes tasks and workloads among peers, is a form of networking or network type formed by the peer-to-peer computing model at the application layer.

HydraDAO

Oracle algorithm on the Ontology blockchain that can scale up on-chain and off-chain business.

Sharding

Adopting the “divide and rule” approach, sharding divides the transaction and state of the current blockchain network, thereby enhancing the blockchain to increase the concurrency of transaction processing and verification, thus achieving blockchain scalability.

VBFT

Consensus algorithm on the Ontology blockchain that combines PoS, VRF, and BFT, thus striking a balance between scalability and decentralization.

Virtual Machine

Smart contract code operating environment on the blockchain, which provides deterministic execution results for smart contract.

Trusted Execution Environment

An isolated, secure execution environment that guarantees the privacy and integrity of internal code and data.

NeoVM

NeoVM is a smart contract virtual machine that has Turing completeness and can realize any logic, thus providing technical certainty and execution efficiency.

WASM

WebAssembly-based smart contract virtual machine.

Native Contract

Smart contract written by native blockchain codes and is run by native binary method, instead of virtual machine.

Native Asset

Smart contract assets managed by native contract.

Hybrid Storage

Network structure that supports both blockchain ledger storage and off-chain decentralized storage.

Transaction Fees

Fees paid to blockchain network nodes when sending transactions on the blockchain.

Two-Phase Commit

An algorithm designed to keep the consistency of all the nodes that are based on a distributed system architecture during transaction commits.

MPT

A tree structure combining Merkle Tree and Patricia Tree.

UTXO

Unused transaction output. A UTXO can be used as input for a new transaction.

ONGx

ONG asset issued on the side-chain.

CCMC

Cross-chain management contract on the Ontology platform.

Inter-Chain Transaction Relay

The Ontology cross-chain design is responsible for forwarding transaction requests on one chain to relay nodes on another chain.

ONT ID

Ontology's decentralized identity that identifies and manages an entity's network identity. An entity can correspond to multiple identities, and there is no association between multiple identities.

Asymmetric Cryptographic Algorithm

Also known as public key cryptography, it is a cryptographic algorithm system that uses a pair of keys. The key pair includes a public key that can be disclosed and a private key that needs to be kept secret.

Consensus Management

Smart contracts that are responsible for managing the stake, incentives, and roles of candidate nodes and consensus nodes on the Ontology blockchain.

3. Ontology Architecture

The Ontology architecture provides infrastructure services for the Ontology ecosystem, and focuses on decentralized data security storage and high-performance blockchain services based on smart contracts.

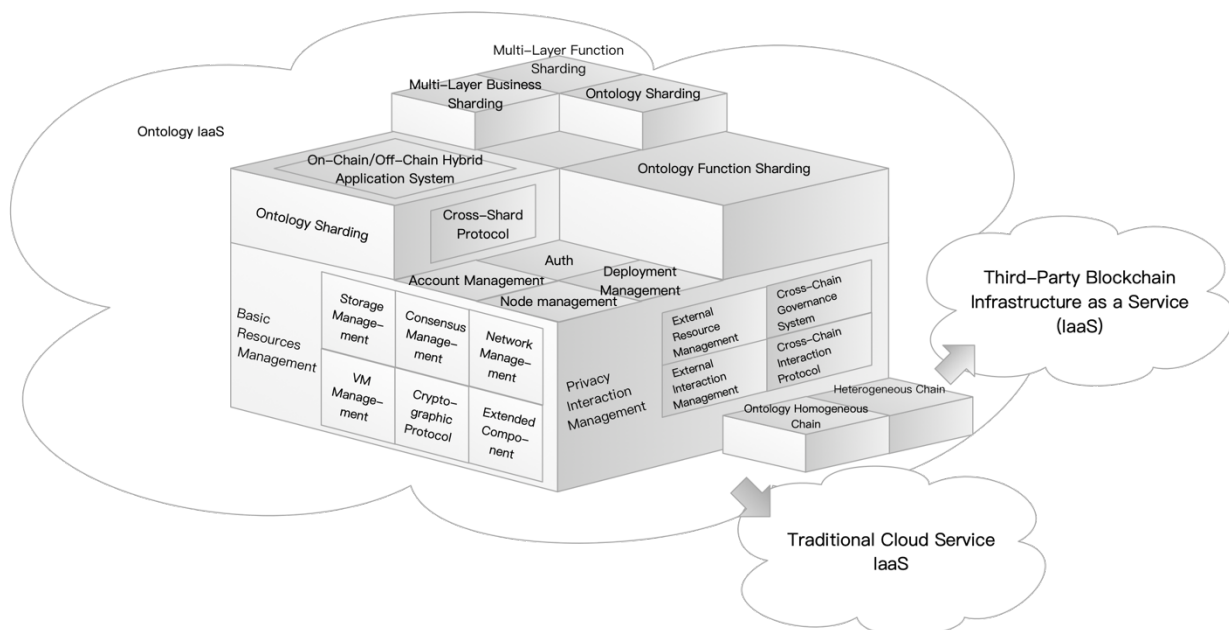


Figure: Service Framework of Ontology Architecture

The Ontology IaaS provides trusted infrastructure services.

The Ontology ecosystem needs to conform to a variety of governance models, which are supported by the Ontology chain network. Each governance model is suitable for one Ontology chain. In the meantime, the Ontology blockchain provides trusted services on the basis of meeting the requirements of governance model. Within the Ontology public chain system, the governance model

achieves stable operation through the preset token models. As for the design of the Ontology MainNet, the governance model becomes a consensus node by staking ONT, and fulfills the accounting obligation to obtain the transaction fees paid in ONG. Ontology adopts the open network hypothesis, aiming to scale up the network with the expansion of the Ontology ecosystem, in which each node operates independently and performs its own duties, thus forming a dynamic equilibrium network in proportion to ecosystem value. The Ontology consensus meets the design requirements of the network hypothesis and governance model.

From the perspective of infrastructure, Ontology laaS adopts a multi-layer sharding technology framework to meet different performance requirements of the upper layer business.

Ontology shards are made up of function shards and business shards.

3.1. Ontology Network Architecture

The design of the Ontology network serves the multichain and multi-layer design of Ontology to meet the following requirements:

1. Different blockchain networks have interoperability at the infrastructure level, and the interaction between different networks is accomplished through the preset interaction protocol. Reliable inter-chain communication is achieved by the consistency judgment of blockchains, while information security is guaranteed by cryptographic protocols;
2. Different blockchain networks have the ability to decide on their own whether or not to interoperate;
3. The same blockchain network has the ability to determine the scalability of shards;
4. Shards have the capability of continuous multi-layer scalability.

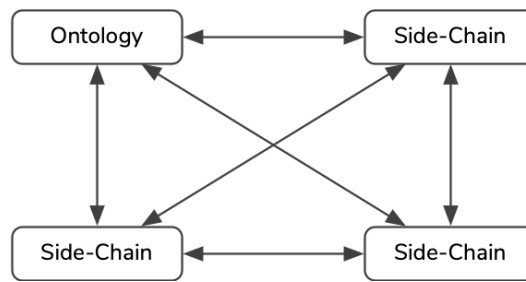


Figure: Ontology Multichain Network Architecture

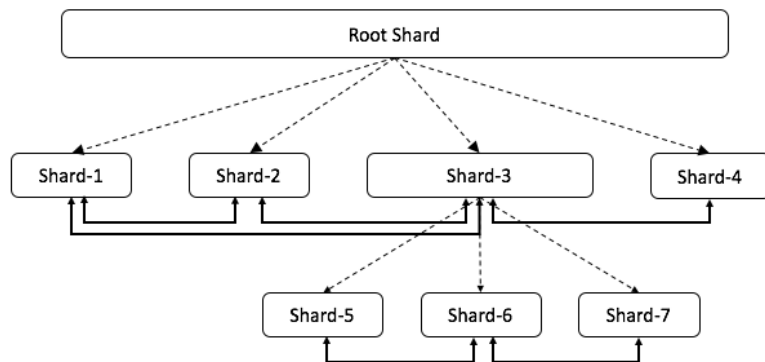


Figure: Ontology Multi-Layer Sharding Network Architecture

3.2. Distributed Ledger Technology

3.2.1. Ontology Consensus

The core ledger supports the Ontology Consensus Engine (OCE), the new-generation consensus engine. OCE is a highly efficient framework consensus engine that can provide modular encapsulation for different consensus protocols. For example, the VBFT algorithm supported by the Ontology consensus framework is a BFT-like consensus algorithm based on VRF (verifiable random function) that can achieve near-infinite scalability and can generate blockchain networks that almost will never fork with only a small amount of computing power.

OCE supports pluggable validators, online protocol repair and upgrade, and is the core of the Ontology network consensus mechanism.

OCE supports a variety of consensus algorithms to meet different network hypothesis. Furthermore, in the multi-layer sharding design, different consensus configurations are supported in different shards.

In the future, in order to optimize resources, OCE will try to differentiate the roles of nodes and participate in consensus with multiple roles so as to achieve consensus governance.

3.2.1.1. VBFT

VBFT, as an important component of OCE, achieves scalability of consensus algorithm by selecting node subsets through VRF, guarantees the anti-attack ability of the algorithm through randomness and PoS, and achieves fast state finality by BFT-like algorithm.

Algorithm features:

- Support generic P2P consensus network;
- Based on BFT algorithm without large amounts of computation;
- Support consensus block generation of large-scale networks;
- The generation speed of block generation depends only on the block propagation speed of consensus networks;
- Finalize the blocks based on probability in a fast way;
- Able to handle reboot of synchronization nodes in the network at any time;

- Able to process network split of P2P consensus network.

In the actual operation process of the Ontology MainNet, the algorithm performance realizes the second-level block generation finality.

The design of VBFT consensus algorithm is closely related to the Ontology governance model. Through VBFT algorithm, the rights of all stakeholders in the Ontology ecosystem to participate in the Ontology network governance, and the norms and rules of rewarding these stakeholders for their participation in network governance can be coordinated.

The stakeholders of the Ontology ecosystem include holders of Ontology stake, candidate (guardian) nodes for margin stake, and the nodes actually participating in the consensus.

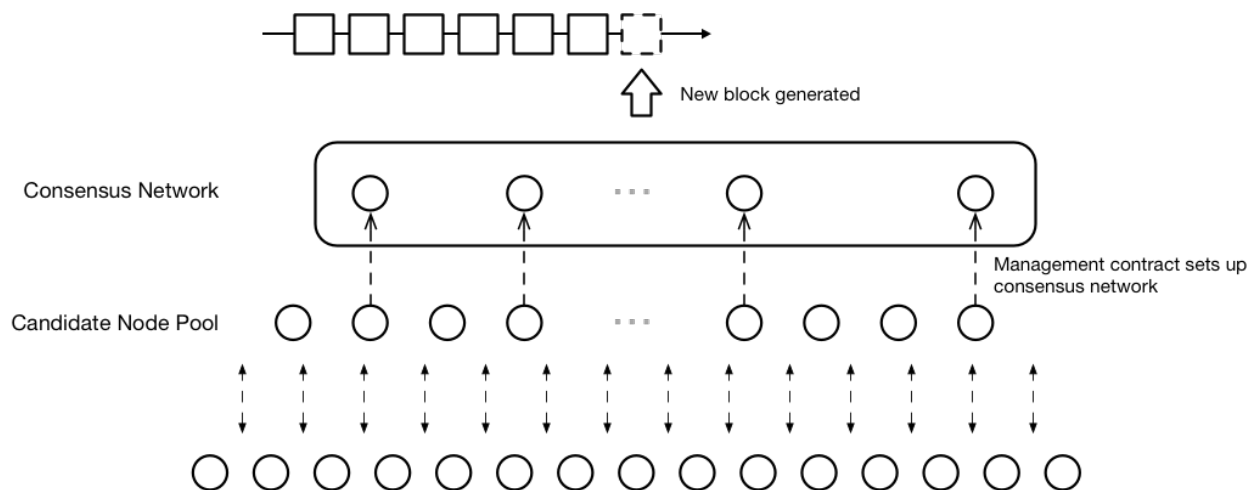


Figure: VBFT Consensus Algorithm

3.2.1.2. Consensus Management Contract

VBFT includes candidate network and consensus network, and provides a promotion channel from candidate network to consensus network, while offering high security. The switch of network nodes is accomplished by consensus management contract. The main scenario of network governance is economic governance, which is carried out through pre-built strategies after community voting.

3.2.1.3. Network Management Consensus

With the expansion of the Ontology network in the future, when emergency governance fails to be implemented as malicious nodes exceed the threshold that consensus algorithm can withstand, the P2P network can be managed by slow consensus. When the consensus network is invalid or hijacked, the malicious nodes could be temporarily isolated with the assistance of slow consensus to categorize the basic P2P network of the Ontology network.

3.2.2. Smart Contract

3.2.2.1. Smart Contract Virtual Machine (VM)

- NeoVM

The core ledger uses the NeoVM for Go as the execution environment of the smart contract, which can realize the smart logic control for the Ontology application layer framework. NeoVM, with Turing completeness, can realize any logic, and has technical certainty and high execution efficiency.

NeoVM combines with parsing transformation of high-level programming languages to support the basic application of virtual machine flexibly. The external interface of virtual machine is provided

through customized API operation, allowing flexible interaction between ledger data and external data. This mechanism not only achieves high performance of native code execution when smart contract runs, but also realizes a universal virtual machine mechanism supporting different blockchains.

With its simplicity and lightweight features, NeoVM provides various types of built-in data, such as integers, bytes, structures, arrays and dictionaries. As the memory allocation of data is completed by the host, many functions can be accomplished by a small number of bytecodes.

- Native Contract Support

Ontology supports native contract, including Auth, native asset (ONT/ONG) management, governance model management, and Ontology Identity Management Contract.

- Wasm VM

Web Assembly (Wasm) is a binary instruction set designed for stack-based virtual machines. It is designed as an open platform compilation target for high-level languages like C/C++/Rust. In addition, it is a Web standard led by W3C and supported by browser providers such as Google, Microsoft and Mozilla. Wasm has many features, including high efficiency, memory security, no undefined behavior, and platform independence.

The Ontology Wasm VM uses minimal API design without compromising its functions; at the same time, it encapsulates the basic APIs such as Rust and C++, and builds the upper common library for contract developers.

3.2.2.2. Hybrid VM Support

On the basis of NeoVM, native contract and Wasm VM, The Ontology infrastructure supports multiple virtual machines to offer services at the same time. Smart contracts supported by different virtual machines can be called to realize complex function logic. In addition, smart contract developers are allowed to choose suitable development tools according to the characteristics of different virtual machines to achieve contract optimization.

3.2.3. Distributed Ledger

Using distributed ledger technology, entity cross-chain and cross-system privacy, and specific cross-chain protocols to achieve parallel execution of processes, that is, multiple steps of a process/transaction are executed on different blockchains or systems to ensure identity privacy of different entities in different systems and on blockchains, and the consistency of the entire transaction.

Distributed ledger can both provide data attestation and support behavior attestation, which means every data request, matching, invocation, and use will be recorded in the ledger to generate a full data record, thus ensuring the security, reliability, and privacy of data.

3.2.4. Performance Solution

Since blockchain services are the basic service infrastructure, different consensus algorithms cause relatively large performance loss, which makes the performance indicators of competing decentralized consensus algorithms and collaborative distributed algorithms quite different. Considering the different performance requirements of different services, Ontology's VBFT algorithm naturally supports consensus algorithms with different degrees of

decentralization, so as to provide a balanced security and performance solution as much as possible.

Furthermore, the Ontology ecosystem will support multi-layer shards to achieve different hardware and performance indicators on different shards, so as to balance performance, security and cost, and meet different degrees of decentralization and security. According to its own needs, business would choose shards of corresponding performance indicators, or decides whether to expand new shards in accordance with its own performance requirements.\

3.2.4.1. Hybrid Storage

The storage resources of distributed ledgers are limited and very expensive, so Ontology designed a hybrid storage solution. Such solution is mapped by data assets, decouples data and asset attributes, supports persistency of all kinds of data by distributed storage service, and provides asset reconciliation service with distributed ledger service at the same time.

3.2.4.2. TEE

Since computing resources of distributed ledgers are also limited and very expensive, Ontology introduces the solution of Trusted Execution Environment (TEE) as an execution scheme to support complex algorithms, which decouples the high value attributes of data processing, data interoperability and data processing, supports complex algorithms in a trusted execution environment, and provides data processing reconciliation service on the basis of distributed ledger service.

3.2.5. HydraDAO

HydraDAO is an off-chain scaling solution of Ontology's distributed ledger, which achieves scalability of performance, function and business, including Oracle access, Layer 2 scaling solution and cross-system communication.

Taking Oracle as an example, HydraDAO's process is described as follows:

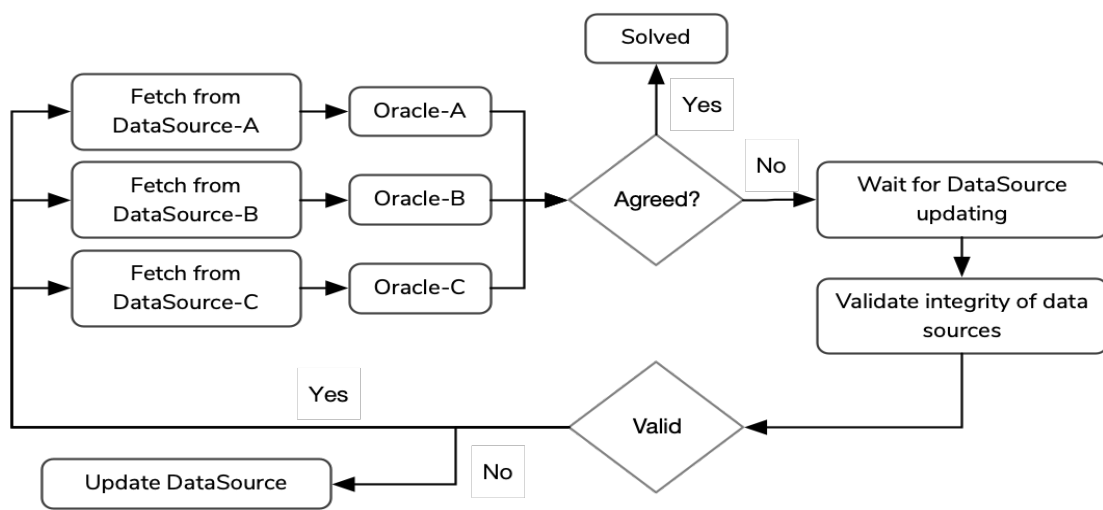


Figure: HydraDAO Execution Logic

1. When any transaction is executed at the beginning or end, it can choose to invoke the Oracle smart contract related to the transaction. Oracle smart contracts and their parameters must be set up at the time the transaction is created and stored on the blockchain as part of the transaction. After the transaction has ended the contract logic is automatically executed to determine which outcomes are the final facts;
2. For transactions that depend only on the state of the blockchain, the Oracle smart contract queries the blockchain data directly from the blockchain interface upon authorization;
3. For most transactions related to the real world, Oracle smart contracts require data to get real world transaction results. When an Oracle smart

1. The use of any data must be authorized by data owners or relevant authorizing parties and follow relevant privacy protection clauses.

contract needs to access/read data in the real world, it must follow the model below: each Oracle randomly chooses one of the multiple trust sources or specifies a threshold number for the data anchor when creating a transaction. At the end of the transaction the global state is verified and interactively updated via the specified trusted data source¹.

3.3. Ontology Sharding Structure

By means of hybrid storage, trusted execution environment, Oracle and Layer 2 solutions, the Ontology distributed ledger and off-chain system are organically combined to improve the business performance in the Ontology ecosystem while ensuring security. At the same time, as the core of the distributed ledger system, the Ontology blockchain system introduces the multi-layer sharding technology framework to further expand the capability of the core system of the blockchain. The Ontology sharding design provides sharding on three dimensions: state sharding, transaction sharding, and network sharding. Below are the three points to be considered:

- Performance

Based on Amdahl's law, the Ontology sharding design adopts smart contract as the basic unit of shards.

- Security (degree of decentralization)

With a multi-layer structure, each shard supports independent distributed deployment, and the “parent shard” is made up of “sub-shards”. Thus, the “parent shard” features the highest degree of decentralization.

- Parent shard: processes global requests in the sharding network;
- Sub-shard: processes smart contract requests in the sharding network.

- Scalability

With the development of business, new shards will be developed and network throughput will also be scaled up accordingly.

The structure of the Ontology shards is shown in the following figure:

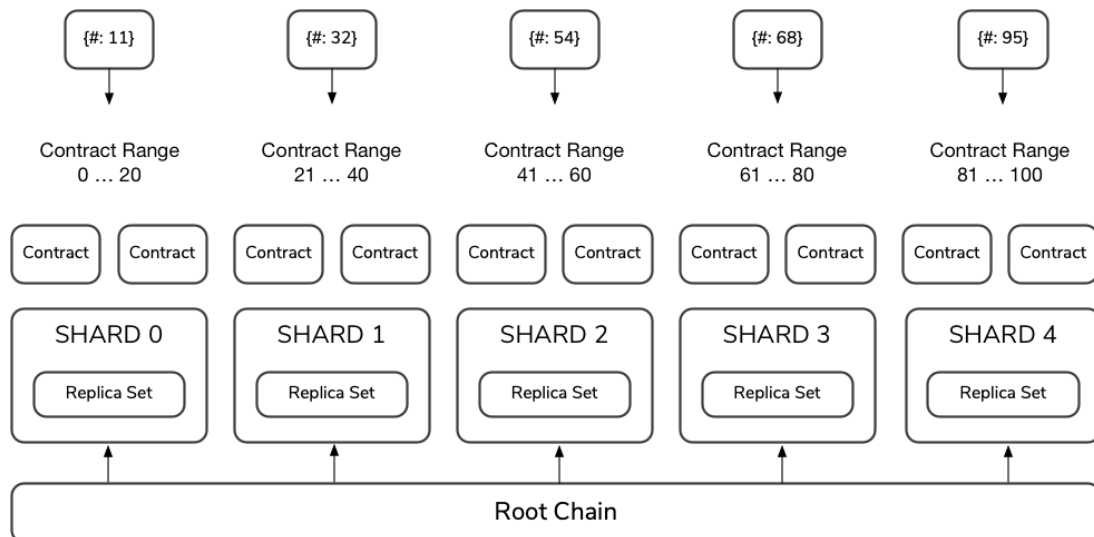


Figure: Ontology Smart Contract Shards

The Ontology sharding focuses on serving various scenarios of cross-shard transactions. It can reduce the performance pressure and storage load of a single shard (chain), and can be adjusted according to function characteristics and business performance requirements. Ontology provides many function modules to serve shards, such as safer transaction storage model MPT, asset atomicity-oriented UTXO model, more powerful Wasm VM support, etc., which can be deployed independently in different shards.

3.3.1. Multi-Layer Cross-Shard Service

The Ontology multi-layer sharding protocol mainly serves sharding structure, sharding governance, cross-shard performance and security.

The sharding structure complies with the following specifications:

- The nodes in the sub-shards are from the nodes in the parent shards;
- Sub-shards should include blocks in the parent shards, and have anchoring relationship with the corresponding blocks;
- Sub-shards should include ledgers in the parent shards, and they are shared among the sub-shard ledgers.

Sharding governance complies with the following rules:

- Governance is completed by sharding governance contracts;
 - Sharding governance contracts are deployed in parent shards;
 - Sharding governance contracts are executed in parent shards;
 - The execution structure of a sharding governance contract is applied to the sub-shard.
- Sub-shards are created by parent shards;
- Managing the life cycle of sub-shards;
- The genesis block of a sub-shard is created by and stored in the parent shard;
- Nodes participate in the sub-shard by staking in the governance contracts on the parent shards;
- The processing cost of sharding transactions in the consensus cycle is forwarded to the corresponding sharding node by root shard after the current sub-shard consensus cycle comes to the end.

The sharding state flow under the sharding management contract is shown in the figure below.

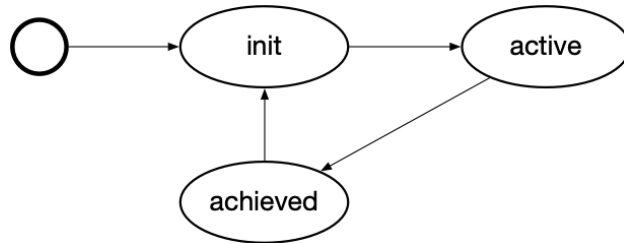


Figure: Ontology Sharding State Flow

3.3.2. Cross-Shard Transactions

Cross-shard transactions adopt two main technical means as shown below:

- Cross-shard message queue. This message queue is designed in the form of system contracts in the Ontology sharding system;
- Smart contract is applied to send cross-shard messages by calling such cross-shard message queue contract to realize the message transmission of cross-shard transactions

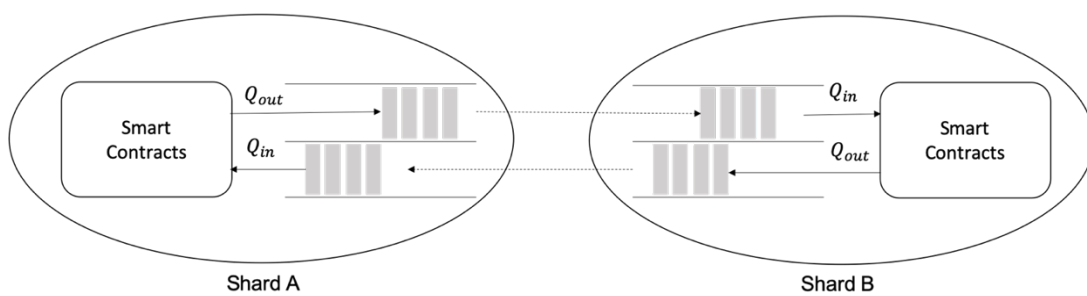


Figure: Ontology Cross-Shard Messaging

Cross-shard transactions use message queue, and adopt a two-phase submission process to complete transaction execution and submission, so as to realize Ontology cross-shard transactions.

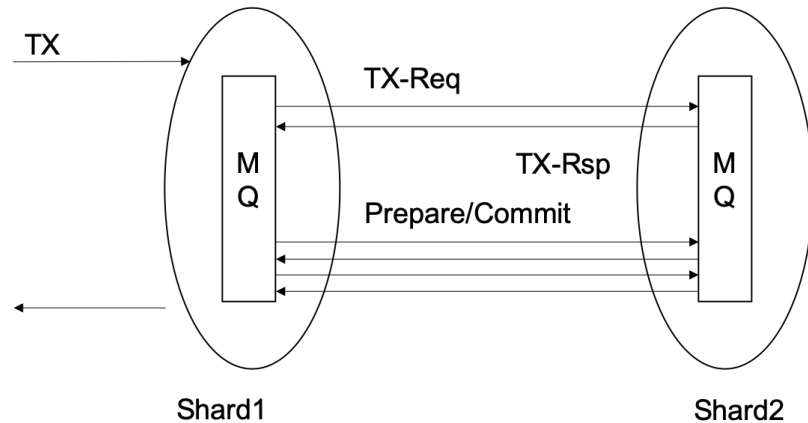


Figure: Ontology Cross-Shard Transaction

3.3.3. Function Shards

Furthermore, Ontology shards also support functional business shards. For example, in order to grow the Ontology ecosystem, there is a plan for traceability shards. At this stage, the Ontology function shards are centered on the functions of identity and data storage, expansion of computing power and so on.

3.3.3.1. Ontology Identity

People, assets, things, and affairs can all have identifications. The Ontology identity provides identification criteria that are able to serve multiple business shards.

Businesses serve users, and users in turn can choose a variety of businesses on their own, thus forming the user ecosystem. At the

same time, users can create information, which has value, thus leading to the Internet of Value.

3.3.3.2. Ontology Storage Resources

Trust service, as the core value of blockchain service, still belongs to cloud service technically. Storage resource is one of the basic services of cloud service. Ontology provides the function chain for resources storage and supports the use of stored resources and the data exchange framework.

3.3.3.3. Ontology Computing Resources

The Ontology infrastructure supports hybrid computation on and off the chain. Combining with existing technology, there are Layer 2 scaling solution and infrastructure solution of the trusted execution environment. Each shard can deploy the solutions independently according to their needs.

3.3.4. Business Shards

The Ontology core team provides blockchain infrastructure, while the Ontology ecosystem provides effective solutions for moving businesses onto the chain. The same type of business under the same governance model can use the same business shard. Business shards can continue to expand sharding to meet their scaling needs, which is the application scenario of “multi-layer” sharding.

3.4. Ontology Chain Network Structure

Based on the existing technology architecture, a chain can only support a single type of governance model. Since systems in the real world are more complex, a single governance model cannot meet the real needs. Therefore, Ontology’s design proposed the chain network

structure from the very beginning in order to meet different governance needs of different chains. Ontology can be the complex trust network through a secure and trusted cross-chain mechanism.

The technology architecture of Ontology shards is complementary to the Ontology chain network architecture. Ontology shards form a three-dimensional scaling system, which pays more attention to the performance of cross-shard transactions; while the Ontology chain network solves the problem of business collaboration under diversified governance models, and attaches great importance to the needs of different governance models, compatibility of the economic model and business consistency in the Ontology ecosystem.

Similar to the Ontology sharding technology, the Ontology multichain solution also supports function scalability, for example, MPT, UTXO, etc.

The Ontology multichain solution supports cross-chain interaction, achieves reliable cross-chain communication, cross-chain atomic transaction, and ensures asset security in cross-chain transactions.

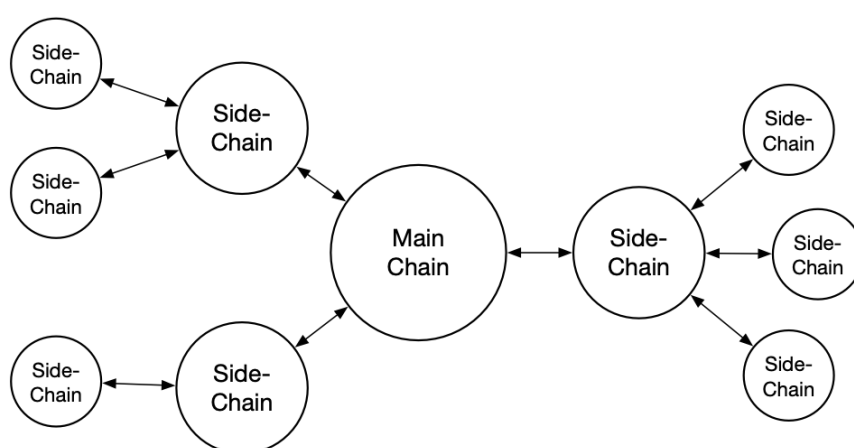


Figure: Multichain Network Topology

The Ontology multichain design uses side-chain technology and a multi-layer star network structure, under which each side-chain can be extended to “more side-chains”.

3.4.1. Cross-Chain Basic Services

The cross-chain service protocol mainly includes three parts: ecosystem economic tools, cross-chain security mechanism, and interaction standards.

The protocol uses ONG as the universal value anchoring tool in the ecosystem. The free circulation of ONG in the ecosystem is guaranteed by the related contracts of the main chain and side-chains of ONT.

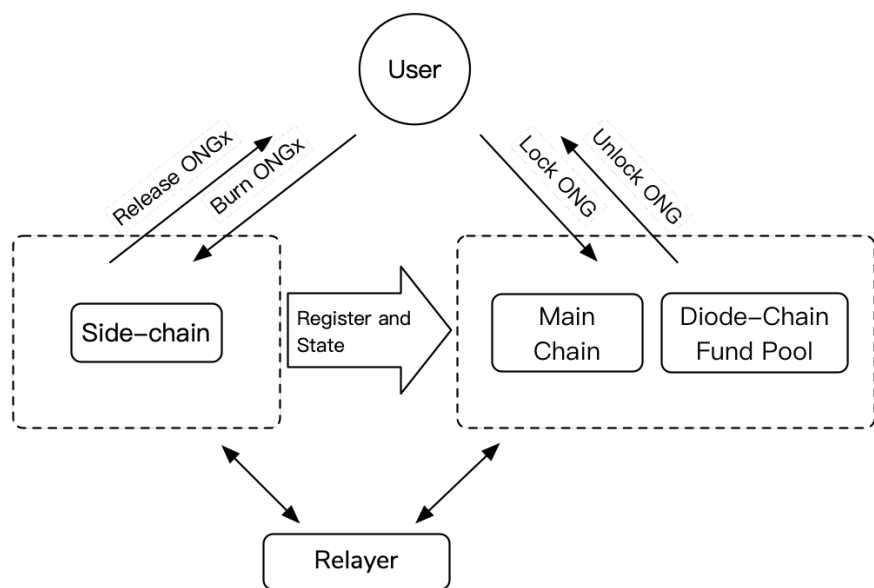


Figure: Cross-chain Economic Model - Utility Token Value Anchoring and Circulation

Cross-chain economic service protocol and security protocol are embodied in the process of side-chain registration:

- The side-chain freezes a certain amount of ONG;

- Determining and locking the supply of ONGx on the side-chain and its exchange ratio to ONG;
- Registering the genesis block of the side-chain to the corresponding main chain;
- Initializing side-chain properties.

According to the basic service protocol, the multichain structure can be extended to support three features: layered star network, no closed loop, and scalable.

Cross-chain protocols, like sharding protocols, also support cross-chain interaction between nodes and functions (such as cross-chain identity system). Furthermore, business applications on different chains integrate with distributed financial technology to form the Ontology multichain asset ecosystem.

3.4.2. Cross-Chain Transactions

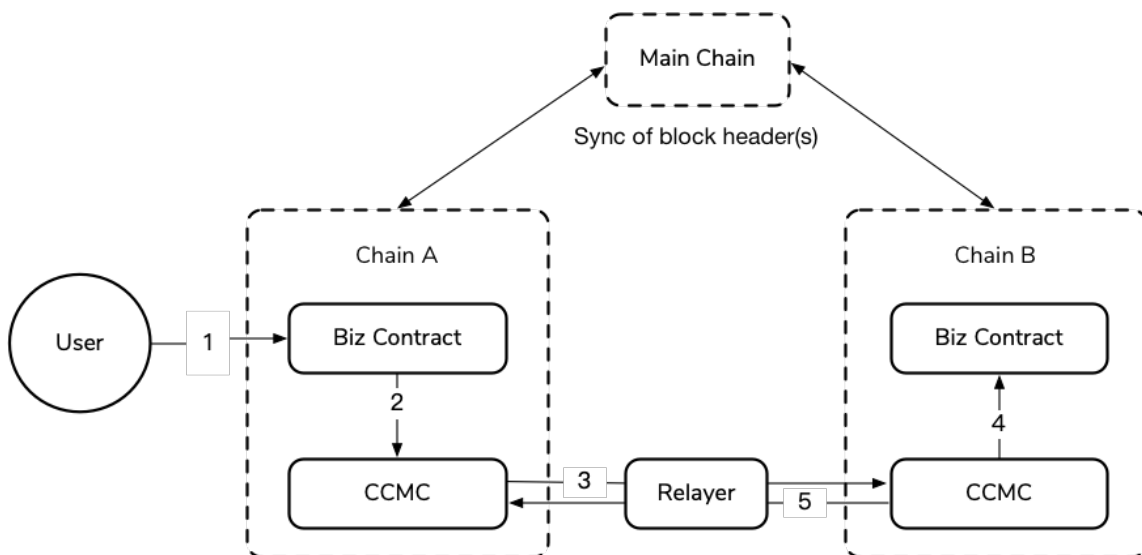


Figure: Cross-Chain Transactions

Ontology adopts simple models to realize multichain transactions with the following solution:

- Realizing cross-chain interaction through inter-chain transaction relay.
- Obtaining each other's key block headers through the main chain;
- Ensuring security using MPT (Merkle-Patricia tree);
- Providing incentives for "honest" relay behaviors.

3.4.3. Chain Network Governance

3.4.3.1. Network Governance

Group network governance. Malicious nodes will be excluded from the network when the ecosystem chain exceeds consensus fault tolerance, and honest nodes will be selected to participate in the consensus.

3.4.3.2. Consensus Governance

In a single ecosystem chain, the consensus network should be managed in coordination with the governance model.

The nodes of different ecosystem chains in the chain network should be deployed to different servers on the basis of the differences of server computing ability. The same server can participate in multiple ecosystem contributions.

3.4.3.3. Economic Governance

Each single ecosystem chain has full autonomy over digital assets and can apply its own economic model to achieve economic governance.

As for Ontology ecosystem, ONG serves as a functional token tool for basic services.

3.4.4. Function Chain and Business Chain

Similar to Ontology shards, the Ontology multichain design also supports the scalability of the customized function chain and business chain under different governance models.

3.4.5. Hyper-Converged Chain Network

Ontology proposed the concept of “Hyper-Converged Chain Network”, which is based on the design concept of “multichain compliance, multi-layer scalability, on-and-off-chain hybrid application model”, and strives to provide infrastructure support for cross-regional and complex ecosystems.

Furthermore, in the business ecosystem of hyper-converged chain network, Ontology, as the infrastructure provider, tries to provide more targeted service for node and network contributors, allowing more contributors to participate in the collaboration in different roles, and to receive due economic returns, which will accelerate the development of Ontology ecosystem.

Businesses look for chains or shards that conform to its characteristics, and provide support for cross-shard and cross-chain interaction as well as shard-to-shard and chain-to-chain business transfer.

The “Hyper-Converged Chain Network” serves as an open technology management framework for the application ecosystem of “competition and collaboration”.

4. Postscript

This White Paper introduces the technology and protocols adopted by the Ontology infrastructure, and will be updated as more application scenarios are being supported.

Ontology is committed to building an open, collaborative, and innovative technology ecosystem. The Ontology team welcomes developers from around the world to join the Ontology family and help us promote the advancement of Ontology's technology.

Contact Us

Email:
contact@ont.io

Telegram:
[OntologyNetwork](#)

Twitter:
[OntologyNetwork](#)

Facebook:
[ONTnetwork](#)

Reddit:
[OntologyNetwork](#)

Discord:
<https://discord.gg/vKRdcct>

Medium:
[OntologyNetwork](#)

LinkedIn:
[Ontology](#)