Available online at https://www.cnki.net

The Journal of China Universities of Posts and Telecommunications

October 2021, 28(5): 1 – 17

https://jcupt.bupt.edu.cn

Research on cross-chain and interoperability for blockchain system

Li Ming [1,2,3] (✉), Qiu Honglin[4], Xu Quanqing[4], Song Wenpeng[1,2], Liu Baixiang[1,2]

1. Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai 200433, China
2. School of Computer Science, Fudan University, Shanghai 200433, China
3. China Electronics Standardization Institute, Beijing 100007, China
4. Ant Group, Hangzhou 310063, China

Abstract

At present, there is an urgent need for blockchain interoperability technology to realize interconnection between various blockchains, data communication and value transfer between blockchains, so as to break the 'value silo' phenomenon of each blockchain. Firstly, it lists what people understand about the concept of interoperability. Secondly, it gives the key technical issues of cross-chain, including cross-chain mechanism, interoperability, eventual consistency, and universality. Then, the implementation of each cross-chain key technology is analyzed, including Hash-locking, two-way peg, notary schemes, relay chain scheme, cross-chain protocol, and global identity system. Immediately after that, five typical cross-chain systems are introduced and comparative analysis is made. In addition, two examples of cross-chain programmability and their analysis are given. Finally, the current state of cross-chain technology is summarized from two aspects: key technology implementation and cross-chain application enforcement. The cross-chain technology as a whole has formed a centralized fixed mechanism, as well as a trend of modular design, and some of the solutions to mature applications were established in the relevant standards organizations, and the cross-chain technology architecture tends to be unified, which is expected to accelerate the evolution of the open cross-chain network that supports the real needs of the interconnection of all chains.

Keywords blockchain, distributed ledger technology, cross-chain, interoperability, Hash-locking, notary, relay, multi-chain

1 Introduction

Cross-chain technology is essentially a technique for securely and trustfully transferring data, or message, or message from one chain to the other chain and producing the desired effect on the chain. Early cross-chain technologies are represented by Ripple and Bitcoin relay (BTCRelay), existing cross-chain technologies are represented by Polkadot and Cosmos, and emerging cross-chain technologies are represented by Fusion, which implements multi-currency smart contracts.

In 2012, Ripple released the InterLedger protocol (ILP), which enables cross-ledger transfers by means of a third-party notary, proposed the first cross-ledger interoperability scheme in the blockchain field. In 2014, the pegged sidechains were proposed fistly [1],

Received date: 21-06-2021
Corresponding author: Li Ming, E-mail: 13701388822@139.com
DOI: 10.19682/j.cnki.1005-8885.2021.0029

which is a cross-chain interaction scheme that introduces a two-way peg with the main chain, which enables cross-chain asset transfers. In 2015, the Bitcoin Lightning Network used the Hash-locking scheme to achieve a fast transaction channel under the Bitcoin chain. In 2016, the BTCRelay scheme, a relay-based cross-chain solution that enables one-way cross-chain connectivity from Bitcoin to Ethereum, was introduced. In Ref. [2], a comprehensive and in-depth analysis of blockchain interoperability issues was provided. The proposal to build a cross-chain web infrastructure platform was first proposed in 2017 by Polkadot and Cosmos, where Polkadot was created by Gavin Wood, co-founder of Ethereum, and Cosmos is an open source community project built by the Tendermint team, both of which are still in development. In 2020, Gartner summarized the concept of blockchain interoperability in its latest blockchain technology research report [3]. In addition to cross-chain interactions between blockchains, blockchain interoperability also encompasses functions such as assets and other information exchanging data between the blockchain and off-chain worlds.

With the development of blockchain technology, the concept of blockchain interoperability was improved. This paper will mainly elaborated on the concept of inter-blockchain interoperability in blockchain interoperability, summarized the key technical problems to be solved in inter-chain interoperability with the existing cross-chain solutions of blockchain, described the principles of several key cross-chain technology implementations, how to solve the key cross-chain technical problems and analyzed their applications. Immediately after that, 5 typical cross-chain systems were introduced. In addition, 2 examples of cross-chain programmability and their analysis was given. Finally, the future trends of cross-chain technology based on the current progress of cross-chain technology and applications were summarized.

2 Related work

Blockchain interoperability is becoming one of the key features of blockchain technology, and some scholars summarized and elaborated on interoperability among blockchains from different perspectives. Li et al. [4] analyzed the demand and technical difficulties faced by cross-chain technology, summarized the developing cross-chain technology, comprehensively analyzed the security risks of cross-chain technology, and summarized and discussed the future development trend of cross-chain technology. Xu et al. [5] analyzed the urgent problems of cross-chaining in terms of security, connectivity, and message synchronization, discussed the solutions of existing cross-chaining projects to these problems, synthesized the limitations of cross-chaining technology, and looked forward to the development trends of cross-chaining in several areas. Belchior et al. [6] conducted a review by collecting 404 papers on blockchain interoperability and divided the research into three categories: public connectors, blockchain of blockchains, and hybrid connectors, each of which is divided into several subcategories that discuss blockchain interoperability technologies, standards, application cases, unresolved challenges, and research directions in the future.

This review mainly focused on three aspects. Firstly, focused on the 4 major cross-chain system problems, the classification of solutions, and their considerations. Secondly, the cross-chain refinement system can be based on the scenario needs for suggestions of technical solution selection. Thirdly, provided the use of cross-chain programmability in major projects and did a comparative

3 Concept of interoperability

Regarding the definition of cross-chain, Vitalik Buterin [2], the founder of Ethereum, proposed the concept of blockchain interoperability, citing three implementation schemes of cross-chain interoperability: notary schemes, side/relay chains, and Hash-locking. Gartner [7] defined blockchain interoperability as the ability of using different distributed ledgers of blockchain platforms to seamlessly interact and transfer assets and other information from one blockchain network to another blockchain and other systems. This can be done without explicitly targeting a custom platform-to-platform gateway or exchange for each use case. EU Blockchain Observatory and Forum [8] considered blockchain interoperability as the ability of blockchains to exchange data with other platforms (including platforms running different types of blockchains) as well as with the off-chain world. Blockchain interoperability refers to the ability of a blockchain system instance to exchange information with other system instances and to use the exchanged information, and is divided into application-blockchain interoperability, inter-blockchain interoperability, and blockchain and off-chain interoperability.

By combining the above definitions of blockchain interoperability, it can be concluded that the concept of blockchain interoperability is not only limited to the interoperability between blockchains, but also includes the interoperability between blockchain and external systems. The interoperability function supports not only the exchange of assets but also the exchange of information. The cross-chain technology described in this paper is blockchain interoperability, that is, interoperability between blockchains, but not interoperability between blockchains and external systems.

4 Key technical issues on cross-chain

This section focuses on the key technical problems usually solved by cross-chain solutions. The design of various cross-chain solutions was studied and four key technical problems were summarized: cross-chain mechanism, interoperability, eventual consistency, and universality. And describes the definition of the problems and the solutions of existing cross-chain respectively. Four key technical issues including framework models, asset/information interoperability, data consistency among cross chains, and universal application are discussed in follows, as to construct complete, flexible and practical cross-chain interoperability. The interoperability between blockchain system and traditional system is also important, however it will not be discussed in this paper.

4.1 Cross-chain mechanism

Cross-chain mechanism refers to the mechanism that the information (including information, assets, and other information written into the ledger) that has reached certainty by consensus in one blockchain is read by another blockchain and verifies its integrity, which solves the core problem of secure and trustworthy transmission of information between blockchains. Third-party systems are also usually required to be introduced between two blockchains to achieve cross-chain interaction processes, as if the security of a blockchain is guaranteed by a consensus mechanism.

A semi-trusted model refers to performing cross-chain message verification by introducing a trusted third-party, and the blockchain trusts the verification results of the trusted third-party. This trusted third-party, which usually also becomes a notary public, can consist of a single recognized person, and some programs, for added security, will have multiple notaries forming a group of notaries, using multiple signatures to arrive at a scenario where individual notaries are tolerated for their misdeeds. Cross-chain implementations of the semi-trusted model are widely available, with ILP as a typical implementation for alliance chains and wrapped Bitcoin (WBTC) as a typical implementation for public chains.

There are two main implementations of the trustless model. The sidechain mechanism divides the cross-chain parties into main chain and sidechain, and the consensus node of the sidechain holds the light client of the main chain. The sidechain will periodically synchronize the block headers on the main chain by consensus, so that the sidechain can verify the existence and integrity of any transaction on the main chain by consensus. In addition, zero knowledge proof (ZKP) technology [9] can also be applied to the sidechain mechanism, because the sidechain has to carry out heavy transaction verification resulting in low performance, the verifiable computation of ZKP can be introduced to verify the existence of the transaction under the chain and generate the proof of de-trust, and the sidechain only needs to verify the ZKP lightly, so as to achieve the effect of efficient verification. The implementation is disclosed in ZKP relayer. The second type of implementation that introduces a de-trusted third-party, one implementation is the relay chain mechanism.

The design of the cross-chain mechanism directly affects security, performance, and interoperability of the overall cross-chain system. The public chain usually emphasizes security, and the implementation is mainly based on the trustless model. Many scenarios can accept the semi-trusted model implementation, but it will restrict the scale of the value flow on the cross-chain system. In the alliance chain, because the consensus node of a blockchain often consists of multiple credible parties, the chain itself is a semi-trusted model, so the main emphasis is on performance and interoperability, and the semi-trusted model is usually chosen as the main implementation.

4.2 Interoperability

Interoperability refers to the function provided by the cross-chain system for the application or smart contract. From the classification of the object of cross-chain interoperability, it can be divided into inter-blockchain asset exchange and inter-chain information exchange. And from the classification of universality, it can be divided into dedicated interoperability function and general interoperability function.

The inter-chain asset exchange function means allowing assets to flow across-chains from one blockchain to another and guaranteeing asset conservation between the two blockchains. Through the sidechain mechanism, another blockchain learns that an asset on that chain has flowed into the cross-chain account, so it issues the corresponding asset on this chain to the corresponding account in accordance with the cross-chain protocol.

The inter-blockchain information exchange function allows functional modules on the blockchain (including smart contracts) to exchange information, such as messaging between smart contracts, and cross-chain calls between smart contracts. Based on information exchange, smart contracts allow flexible customization of cross-chain logic, and it is a universal interoperability function. The public chain implementations that enable information interoperability are Cosmos, and in alliance chains, HyperLedger Cactus, Antchain open data access trusted service (ODATS), and WeCross all support information interoperability. Antchain ODATS provides a blockchain ledger data addressing function for smart contracts, allowing them to send a request for ledger data on any blockchain, including the identification of the target blockchain and ledger data (such as Hash transaction). The cross-chain system will route the address to get the ledger data, complete the data existence and integrity verification and write back to the smart contract.

The interoperability of cross-chain system affects the usage scenarios of this cross-chain system, but the generic cross-chain interoperability has high complexity implementation, which brings certain performance problems and security issues.

4.3 Eventual consistency

Eventual consistency means that when inter-chain assets are exchanged (or inter-chain information is exchanged), the assets (or state) between multiple blockchains that implement the interoperation can achieve eventual consistency within a specified time window, i.e., either everyone executes successfully or all can roll back to the state before the interoperation was initiated.

Inter-chain asset exchange means that after the assets are destroyed (or frozen) in the first blockchain, the second blockchain can eventually issue the corresponding number of assets, and there is no wrong state such as the second blockchain cannot issue the corresponding assets, while the assets of the first blockchain are frozen forever.

Generic cross-chain transactions, meaning that multiple chains perform customized multiple actions to form a distributed transaction that can either reach eventual consistency or roll back to the original state. The key point of generic cross-chain transactions is to implement a de-trusted transaction manager, as exemplified by the WeCross solution, which implements a cross-chain transaction manager through smart contracts. The contract can define multiple blockchain subtransactions to be executed for a cross-chain transaction, using a two-stage commit method. When the transaction is executed, the smart contract will record the execution status of each subtransaction, coordinate the execution of each blockchain subtransaction first, and then coordinate the execution of each subtransaction to commit the transaction after all subtransactions are completed. If there is a transaction participant in the first phase to be able to execute the completed transaction, the second phase will coordinate the execution of rollback for each subtransaction. It implements a de-trusted transaction manager by means of smart contracts, making it possible to change cross-chain transactions that only smart contracts can control and no third-party can interfere with their execution. Support similar generic cross-chain transaction implementations as InterLedger, HyperLedger Catus, and other implementations, while Cosmos, Polkadot can also achieve cross-chain transaction manager through the customization of the function of smart contract communication.

The eventual consistency affects the cross-chain business security of cross-chain and thus limits the using scenarios of cross-chain. The cross-chain implementation with security risks in the eventual consistency may lead to double-spending and other problems that do not protect the interests of cross-chain participants. In general, cross-chain solutions need to design additional mechanisms to minimize the occurrence of possible problems around their eventual consistency.

4.4 Universality

Cross-chain technology solutions usually need to meet certain universality, and generally address two aspects of universality: the universality of interoperability and the universality of heterogeneous chain support. The universality of interoperability means that not only the single scenario of cross-chain asset exchange is supported, but also more complex types of cross-chain interoperability can be extended.

There are two general approaches to achieve interoperable universality: the first is to support generic cross-chain transactions, and the second is to design generic cross-chain communication protocols. Take WeCross, HyperLedger Catus, and HyperService [10] protocols as examples, they all support universal transaction managers, which allow each blockchain to customize the logic of subtransactions through smart contracts and other means, and coordinate each subtransaction through the transaction manager and thus realizing diverse cross-chain transactions.

There are two general approaches to implement generic heterogeneous chain support. The first is to support cross-chain bridge adaptation, and the second is to use functionally customizable sub-chains. The second way is adopted by Cosmos and Polkadot's implementation, Polkadot defines the blockchain skeleton of substrate's customizable functions, and substrate implements the cross-chain protocol of Polkadot network at the bottom, providing a unified cross-chain interoperable interface to the upper modules, allowing developers to use substrate to develop personalized blockchains with different functions (called parallel chains in the Polkadot). These personalized blockchains are all interconnected through a unified interoperability interface.

The universality will affect the use scenario of cross-chain solutions, but in order to improve the universality, the complexity of cross-chain system implementation is often high, and the more universal solutions lead to the higher complexity of cross-chain protocols as well as that of adapting to heterogeneous chains. At the present stage, the main application of public blockchains is asset interaction, so the universality of interoperability is not yet so high, while the scenario of decentralized finance (DeFi) has higher requirements for the universality of heterogeneous chain support. Alliance chain is widely used in many different industries. The business processes of different industries are highly different, and there are many kinds of heterogeneous chains in business, so there is a high demand for the universality of both aspects in the scenario of alliance chain.

The above are a few basic problems listed, and the cross-chain has to solve not only the above four problems, but also other advanced problems such as performance, friendliness, security, privacy, etc. The overall cross-chain technology is still in the process of evolution. Table 1 shows the support of different cross-chain products.

Table 1 Cross-chain project support for key technical issues

Mechanism	Public blockchain technology solution						Alliance chain technology solution				Innovative solutions	
	Polkadot	Cosmos	Bitcoin Relayer	Rootstock	WBTC	TBTC	InterLedger	HyperLedger Catus	WeCross	ODATS	ZKP relayer	HyperService
Trustless cross-chain mechanism	√	√	√	√		√		√			√	√
Semi-trusted cross-chain mechanism				√	√		√		√	√		
Asset exchange interoperability	√	√	√	√	√	√	√	√	√	√	√	√
Information exchange interoperability	√	√						√	√	√		√
Eventual consistency	√	√	√	√	√	√	√	√	√	√	√	√
Generic cross-chain transactions	√	√						√	√	√		√
Interoperability universality	√	√						√	√	√		√
Heterogeneous chain support universality	√	√			√	√	√	√	√	√		√

5 Implementation of key cross-chain technology

5.1 Hash-locking

Hash-locking is a form of technical implementation proposed in Lightning Network. In order to achieve Hash time lock, firstly, Hash lock, locked by Hash value, and only with the original value for the generation of this Hash value to unlock after locking. Secondly, time lock requires the entry of password of the Hash lock within a specified time. The condition to open these two locks is that the original Hash value is entered within the specified time.

The so-called one-way Hash time lock, the sender randomly generates a secret key of x, and a smart contract would be constructed through the Hash value $h(x)$ corresponding to a Hash function. In the contract, a timeout time shall be set to stipulate that the receiver's unlock through the secret key x only within that time could be considered as valid. If the timeout time is exceeded, only the sender of the asset can unlock and return the asset.

In this implementation, it is a trustless cross-chain mechanism. It is a dedicated interoperability function in terms of interoperability. It only supports asset cross-chain and can achieve eventual consistency within a certain window.

5.2 Two-way peg

Two-way peg implements mutual anchoring between two chains, sidechain and main chain, to enable the sidechain to verify the behavior of the main chain when an agreed flow of money/state occurs on the main chain, and then automatically execute the agreed flow of sidechain assets.

The principle of two-way peg is implemented based on the characteristics of simplified payment verification (SPV), Similarly, the one-way anchoring from the sidechain to the main chain is the same process.

Fig. 1 illustrates an example of two-way peg process.

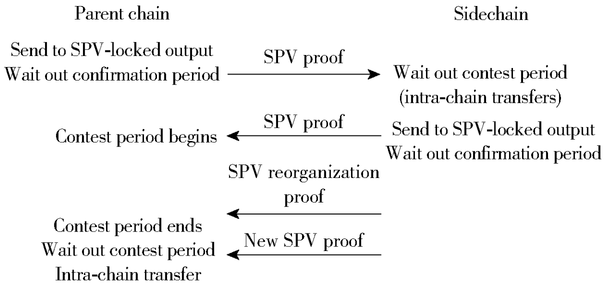

Fig. 1 Example of two-way peg protocol

The two-way peg based on SPV belongs to the implementation of trustless cross-chain mechanism, which does not need to trust any third-party and has relatively high security, but the process of anchoring directly through SPV will lead to many problems, such as the problem of Gas cost and performance problems brought by maintaining an SPV client on the blockchain, and the scalability is also limited.

Two-way peg is also classified as symmetric two-way peg and asymmetric two-way peg. Symmetric two-way peg refers to the use of the same anchoring method between the main chain and the sidechain, while asymmetric two-way peg refers to the use of different anchoring methods between the main chain and the sidechain respectively (e. g. the main chain anchors the sidechain using the trustless method and the sidechain anchors the main chain using the semi-trusted method).

Two-way peg is the basic principle of sidechain technology which solves the problem of how to complete the mutual recognition of blocks/transactions between blockchains. It is a kind of trustless cross-chain mechanism with the advantage of high security. The principle is also widely used in various cross-chain projects, such as in Cosmos system, the communication between chains is carried out by the principle of two-way peg. This scheme is also used in earlier schemes such as Bitcoin relayer, Rootstock [11]. The latest cross-chain project NEAR also adopts this scheme, which implements the verification client of Ethereum through the Rust language and the verification client of NEAR through the smart contract of Ethereum, thus realizing the symmetric two-way peg of trustless. The improvement can be applied to verify the validity of data across chains, and in addition to token exchange, it can handle more widely applied cross-chain transactions.

The two-way peg implementation has high security, because it is usually a trustless cross-chain mechanism with strong interoperability, but it cannot solve the problem of universality and has high requirements for heterogeneous chain adaptation. It takes longer to reach a eventual consistency.

5.3 Notary schemes

Notary schemes refer to the key technical issues in the process of cross-chain interaction through a trusted third-party, and can be used to solve the authentication issue of data in the cross-chain mechanism, and can also be used to provide specific interoperability functions, and the design of the notary can also meet certain universality according to the positioning of the cross-chain system. Therefore, the notary schemes generally needs to set its functional attributes and security attributes, and the functional attributes refer to the core problems that it solves, security attributes are also the key design of the notary schemes. According to the trust model of the scenario, a centralized single notary can be chosen. A group notary with multiple signatures can be used to improve security, a hardware-secured notary realized by trusted execution environment (TEE) can be used in some alliance chain solutions, and a fully distributed notary with de-trust can be realized inside the public chain. Notary schemes mainly include three types.

1) Single-signature notary schemes (centralized notary schemes). It is usually acted by a single designated independent node or institution, which takes on the tasks of data collection, transaction confirmation and verification at the same time.

2) Multi-signature notary schemes. Multiple notaries co-sign on their respective ledgers to reach consensus before completing cross-chain transactions. Each node has a secret key, and only when a certain number or ratio of notary signatures is reached can a cross-chain transaction be confirmed.

3) Distributed signature notary schemes. The idea of multi-party computation (MPC) is adopted. A secret

key can be formed based on cryptography and split it into multiple pieces for distribution to randomly selected notaries, and a complete secret key can be assembled after allowing a certain proportion of notaries to co-sign, thus completing a more decentralized data collection and verification process.

This summary analyzes the implementation mechanism of the InterLedger notary, in which the notary takes on the two core functions of transaction verification and transaction management, and uses a multi-signature group notary to ensure security.

A generic cross-ledger multi-hop payment process is designed in InterLedger [12], in which the initiator of payment determines the remittance path of the payment in advance and initiates a remittance offer to the connecting nodes on the remittance path, and when the remittance path satisfies conditions such as executability, the connecting nodes respond to the offer, and the payer collects responses from all connecting nodes on the remittance path to obtain a remittance transaction, and then the remittance transaction can be completed between multiple ledgers using a two-stage coordination. Fig. 2 shows the protocol process. The sending node submits the remittance transaction to the ledger, and the ledger executes the transaction preparation phase as defined by the transaction. In this phase, each ledger prepares liquidity funds by freezing, etc., and generates receipts for the execution of the transaction in the preparation phase, and the notary collects the execution receipts of the ledger in the transaction preparation phase which verifies the correctness of the receipts and after determining that sufficient liquidity funds are ready in each ledger, the notary sends the second stage of transaction submission instructions to the participants on the remittance path of chain, and subsequent transaction processing is executed on the remittance path as specified. The notary is responsible for two major functions of transaction verification and transaction management in the system. If the notary schemes is not available and the sidechain mechanism is used, the challenge will be huge. One of the main reasons is that the ledger supports heterogeneous implementations, and the ledger has to be anchored two-by-two and directly dependent on each other, making it difficult and complex for the implementation of project. InterLedger's notaries support multiple signature mechanisms, allowing the selection of m notaries and at least n notaries reaching a consistent conclusion can it be regarded as trustworthy.

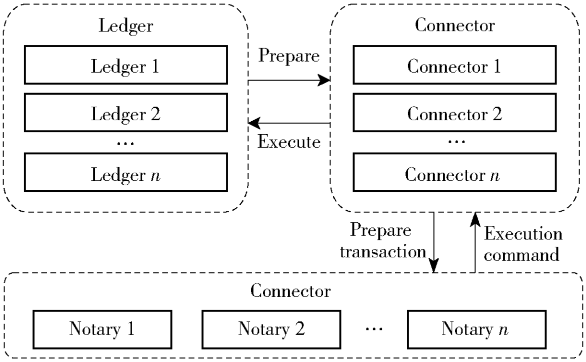

Fig. 2 Example of ILP

The InterLedger notary schemes is divided into 4 processing phases, consisting of 6 algorithmic modules. The 4 processes are: pre-preparation request processing process, preparation request processing process, notification process, and notary processing process. Some of the key algorithms processes are as follows, and in general the algorithms are relatively concise and reflect no relatively high complexity of the notary schemes implementation level.

TEE technology is also applied to implement a trusted notary, which is a hardware-secure TEE that guarantees the integrity and confidentiality of the code. In the application of the notary schemes, the notary function can be implemented as a trusted application running in TEE. TEE guarantees that notary functions are run according to the established logic, and the execution results of TEE notaries carry hardware signatures of TEE to indicate that the execution results are executed according to a predefined trusted application. Take town crier's [13] TEE notary implementation as an example, he implements the blockchain oracle function based on TEE notary. Fig. 3 shows the architecture of town crier. Town crier executes the trusted behavior of the oracle machine to obtain data source through TEE, and the smart contract on the blockchain trusts the data that comes from the specified data source and has not been tampered by the middleman by verifying the hardware signature of the TEE notary.

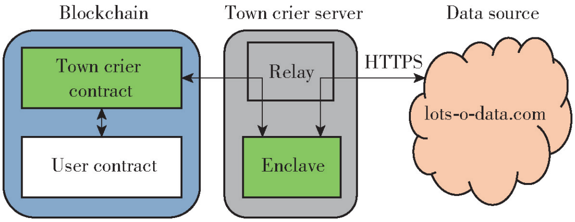

Fig. 3 Town crier system architecture diagram

This method can also be applied to cross-chain. The Antchain ODATS platform adopts the method of TEE notary, which executes the verification client of blockchain in TEE to achieve the function of trustworthy verification of transactions, is shown in Fig. 4. The trust root of TEE is neutral, so it has a certain application scenario in the alliance chain. The TEE root of trust is neutral, so it has certain application scenarios in alliance chains. TEE-based notaries do not need to deploy multiple nodes as multi-signature-based notaries do, and only need one node could complete trusted cross-chain verification.

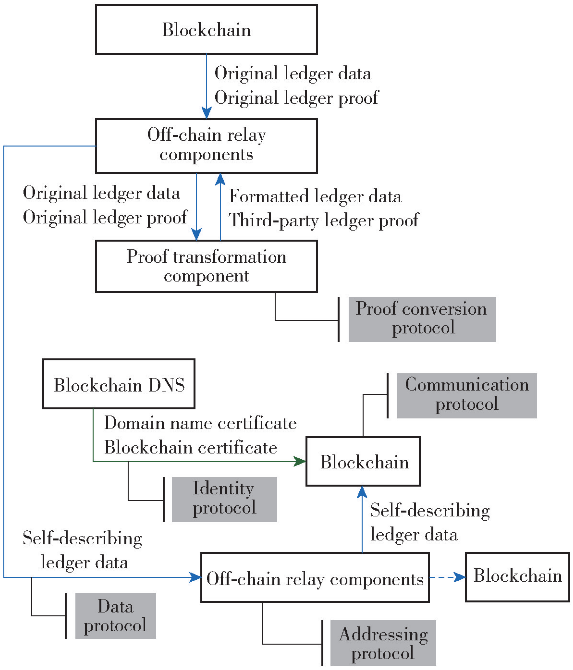

Fig. 4 ODATS cross-chain process

In the cross-chain scenario of alliance chains, because the consensus nodes of blockchains in different business scenarios have different compositions, so the trust models of different blockchains are not consistent, and when deploying cross-chain implementation of notary schemes for alliance chains, different notary schemes need to be constructed according to the trust models of blockchains. TEE is a generic notary schemes based on multiple signatures for the notaries whose election results are secure enough if they have no interest in individual blockchain consensus nodes, and a centralized notary schemes is trustworthy enough for some private chains (chains held by one organization).

The notary implementation has strong interoperability and is a universal interoperability function that can well support evenutal consistency. It has strong universality and is very friendly to heterogeneous chain adaptation. The disadvantage is that it is generally a semi-trusted cross-chain mechanism, so it is not high enough in terms of security.

5.4 Relay chain scheme

Relay chain scheme refers to the construction of a blockchain specifically for satisfying cross-chain functions by building a relay chain based on trustless trust model, providing diverse interoperability types, using blockchain to guarantee the integrity of cross-chain transactions, and satisfying universality in both protocol and architecture. In terms of cross-chain mechanism design, nodes of relay chains usually need to take on several roles (responsibilities), one is to verify cross-chain data, such as block headers, transactions, receipts, etc., the other is to bridge block chains, including listening to cross-chain requests from block chains and sending cross-chain results to the receiving chain, and the third is arbitration/governance, which can be performed when events such as security-related occur through decentralized autonomous organization (DAO). Relay chains can also be designed with more responsibilities according to the overall cross-chain scheme. The smart contract can write the cross-chain data to this queue and the bridge operator of relay chain will listen to this queue to perform the cross-chain operation. The receiving queue stores the verified cross-chain data submitted by the bridge operator of relay chain , and the smart contract reads the cross information in the receiving queue by reading or listening to the event, and completes the cross-chain function of information exchange through the mechanism of this information sending and receiving queue, and the relay chain provides data listening, routing forward and other communication link functions in this mutual operation type.

The relay chain consists of four roles: collator, validator, nominator, and fisherman. The collator helps the validator to collect, verify and submit alternative parallel chain blocks, and it is the full node of the parallel chain, responsible for collecting transactions and packing blocks in the parallel chain. The validator is the full node of the relay chain, which will assign the validator to different parallel chains by random grouping in the validator pool. The validator will accept the blocks packaged from the collator and perform validity verification, and then combine with the consensus algorithm to confirm the blocks submitted by the collator. The nominator is the holder of the governance token in Polkadot (DOT), and it will choose the validator it trusts for DOT pledging and then share the validator's earnings. The role of fisherman is to prevent the network from doing evil and is responsible for the role of reporting. Its role is mainly to regulate the system and earn bonuses by reporting illegal transactions. Although the validator serves different parallel chains through random assignment, which, from a certain perspective, raises the cost of joint doing evil by the validator, but a role like fisherman is also needed to regulate the behavior of validator.

The advantage of relay chain scheme is that it is conducive to building with trustless mechanism, high interoperability and universality. The major disadvantage is that the cross-chain delay is relatively high. The access engineering cost of heterogeneous chains is large because of the complex design of relay chain architecture and protocol. Generally, the relay chain scheme is mainly implemented by public blockchains, such as Cosmos is also implemented by the relay chain, and the beacon chain of Ethereum is also a type of relay chain implementation.

5.5 Cross-chain protocol

Cross-chain protocol refers to the protocol that need to be commonly followed for inter-chain interoperability. This section focuses on how the two universal cross-chain systems, namely Cosmos and Polkadot, implement the cross-chain protocol to support universal interoperability.

Different cross-chain protocol designs have severe impact on the cross-chain models, interoperability, eventual consistency, and versatility. Generally, the realization of cross-chain systems requires different protocol designs according to different and realistic requirements.

5.5.1 Cosmos IBC

Composed of independent blockchains known as 'Zone', Cosmos is a heterogeneous multi-chain system that supports cross-chain interaction shown in Fig. 5. It provides a set of software development kit (SDK) that can complete the construction of blockchains. As an inter-blockchain system, its focus is the cross-chain protocol, otherwise known as cross-chain communication. As shown in Fig. 5, inter-blockchain communication protocal (IBC) [14] is the key to Cosmos operation.

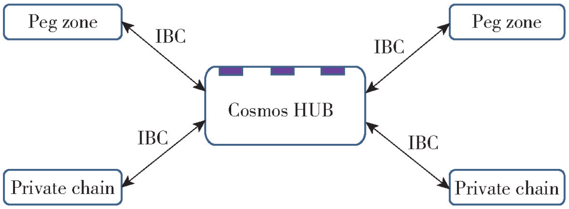

Fig. 5 Cosmos IBC protocol

If communication between blockchains is expected to be achieved, a two-way proof mechanism similar to sidechains is to be adopted. In other words, two interacting blockchains are used to enable information transfer through the two-way transmission of data packets. More specifically, a data packet generated by blockchain is transmitted to other blockchain through HUB. A proof is used to ensure the transmission of data, the receiver verifies that this proof is consistent with the sender's block header, thereby ensuring the authenticity and validity of the data.

5.5.2 Polkadot XCMP

As a heterogeneous multi-chain interchange architecture, Polkadot protocol enables customized sidechains to be linked to public blockchains, which is achieved through relay technology in a similar case with Cosmos. Cross-chain message passing (XCMP) protocol [15] is a subset of the Polkadot protocol. It defines how messages can be passed between parachains when there are no trust assumptions other than the security of shared relay chains, as shown in Fig. 6. The main content consists of message queue mechanisms, message availability, and message input and output. As the messaging protocol for parachains, XCMP relies heavily on the unique relay chain architecture and design of Polkadot. The XCMP protocol uses a simple queue mechanism based on Merkle tree to ensure the correctness of inter-blockchain transactions. The validator on the relay chain is responsible for transferring transactions from the exit queue of the parachain to the entry queue of the targeted chain.

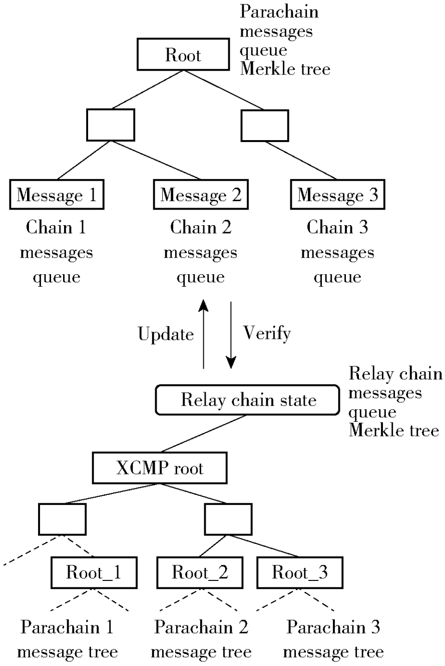

Fig. 6 XCMP protocol

5.6 Global identity system

The identity system is the basis of communication. It aims to provide blockchains with universal identification and reliable binding authentication between the identification and the blockchains in the process of inter-blockchain interoperability. The Ethereum name service (ENS) provides decentralized naming services through smart contracts to provide accounts on blockchains with reliable and verifiable naming bindings, and the authenticity of the naming can be verified through smart contracts when wallets and other interoperations are carried out. Inter-blockchain interoperability also needs naming services. In the relay chain mechanism, relay chains generally manage inter-blockchain identity identification. Yet, this identification only works in the relay chain. In contrast, if Ethereum is connected to Polkadot and Cosmos respectively, it will have two different identities, which means that there would be a greater burden of inter-blockchain identity management if universal inter-blockchain programming is implemented through smart contracts.

As shown in Fig. 7, serial blockchain name services and inter-blockchain interoperability methods concerned with blockchain naming services are designed in the Antchain ODATS solution.

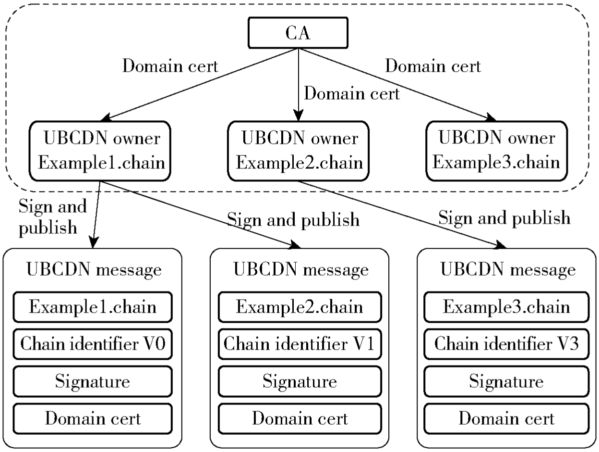

Fig. 7 Antchain ODATS

In all inter-blockchain communications, a blockchain only needs a valid identifiable domain name to unify blockchain domain name (UBCDN) applicable to all blockchains within a universal blockchain network.

The global identity system can introduce functions similar to those of the public key infrastructure (PKI) system. But it is a more lightweight distributed identity system with higher distributed autonomy and extensibility. It solves two problems by issuing two levels of certificates. The first problem is domain name ownership: domain name certificates are issued through certificate authority (CA) to determine the owners of the domain names. The second is the reliable and verifiable bindings between domain names and blockchains: owners of domain name certificates can sign and publish UBCDN certificates to describe the information and root of trust of blockchains.

In the cross-chain interactive process, the interoperability module receives the cross-chain data packet. Then the UBCDN certificate is read via the blockchain domain name contained in the data packet. The trust chain is used to verify that the UBCDN certificate is issued by a valid owner of a domain name. After the acknowledgement of UBCDN's validity, the basic information and the root of trust information of the blockchain are resolved from UBCDN. Then, the trust rootstock of the blockchain is loaded to verify the authenticity of the cross-chain data. For instance, the root of trust of some blockchains needs to specify a genesis block. In other words, the root of trust can load the SPV client based on the genesis block for SPV verification of cross-chain information. If the result is verified, it indicates that the data come from the blockchain identified by the blockchain domain name.

The global identity system is the infrastructure required for an open and connected cross-chain architecture. In the future, this system may have a large part to play where the cross-chain standards can be universally applied. It is also included in the standardized cross-chain modules in the trusted blockchain initiative (TBI) issued by the China Academy of Information and Communication Technology.

6 Introduction to the typical cross-chain system

This section gives an introduction to five typical cross-chain systems: BTCRelay, Polkadot, Cosmos, HyperService, and HyperLedger Cactus.

6.1 BTCRelay

BTCRelay (as shown in Fig. 8) enables Ethereum to have cross-chain access to Bitcoin blockchain data. That is to say, a smart contract is developed in the Ethereum network, by continuously receiving Bitcoin block header pushed by each relayer node in the network, the storage and real-time update of the Bitcoin block header in the smart contract are realized. Since the transaction information of Bitcoin blockchains is chiefly stored in block heads in the form of the Merkle tree, BTCRelay enables Ethereum smart contracts to access BTC blockchain data in a decentralized manner without relying on any third-party intermediary. Although the principle and implementation of BTCRelay are not very complicated, this system enables users to create various trigger conditions by relying on Ethereum smart contracts of information or Bitcoin blockchain events, thereby enhancing the availability of smart contracts. Currently, BTCRelay only supports cross-chain between Ethereum and Bitcoin, and Bitcoin should be prevented from reading information of Ethereum blockchains at the same time. In this cross-chain mode, communication is one-way and thus has certain limitations.

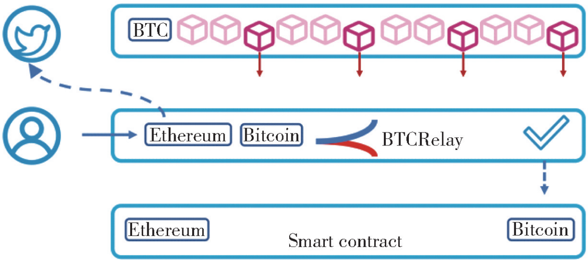

Fig. 8 BTCRelay

6.2 Polkadot

Polkadot is a scalable heterogeneous multi-chain system that provides a huge number of relay chains to connect blockchains that are currently independent of each other. The aim is to provide cross-chain communication between different blockchains.

Therefore, Polkadot is nothing more than a protocol that allows different independent blockchains to exchange information. Polkadot attempts to establish a multi-chain architecture so that all blockchains connected to this architecture can be better complete information exchange between each other. Polkadot defines a set of parachains and relay chains to address extensibility and scalability respectively. The architecture of Polkadot is shown in Fig. 9.

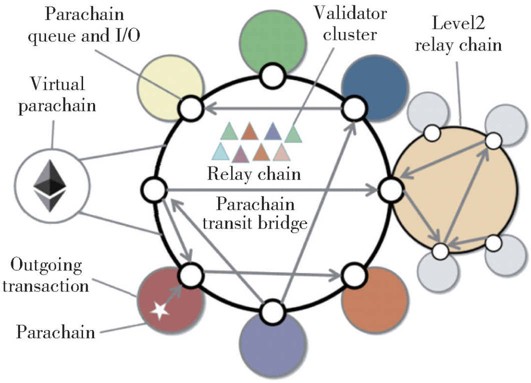

Fig. 9 Polkadot architecture

Polkadot refers to a network of heterogeneous blockchain fractions as 'parachain'. These blockchains are connected to and secured by Polkadot relay chains. They can also be connected to external networks via a bridge.

1) Relay chain. As the core of Polkadot, relay chains are responsible for network security, consensus, and cross-chain interoperability.

2) Parachain. As the sovereign blockchain, parachains can have their own token and optimize the token's functions for specific scenarios. In order to connect to relay chains, parachains can pay on-demand or rent a card slot for continuous connection.

3) Bridge. As a special blockchain, it allows Polkadot shardings to connect to and communicate with such external networks as Ethereum and Bitcoin.

Polkadot has the following five strengths. One is extensibility. Polkadot allows multiple transactions to be processed in parallel. The other is heterogeneous fragmentation. Each chain in the network can be optimized for a specific scenario. The third is upgradable. The transparent chain governance system of Polkadot allows the self-upgrade of blockchains without the need for fork chains. The fourth is transparent governance. Polkadot is managed fairly and transparently by anyone who owns DOT, the Polkadot token. All DOT holders may propose amendments to the agreement or vote on existing proposals. The fifth is cross-chain composability. Because of Polkadot's ability to connect to the blockchain, Polkadot fractions will also be able to interact with decentralized financial protocols and crypto assets that are popular on external networks such as Ethereum.

6.3 Cosmos

Cosmos is a network composed of many independent and parallel blockchains. As shown in Fig. 10, the connection between each blockchain is realized through nodes. The first blockchain in the network will serve as Cosmos HUB (partition) which connects numerous other blockchains through a new cross-chain communication protocol. All cross-partition token transfers are realized through this HUB, and any type of blockchain can connect to Cosmos.

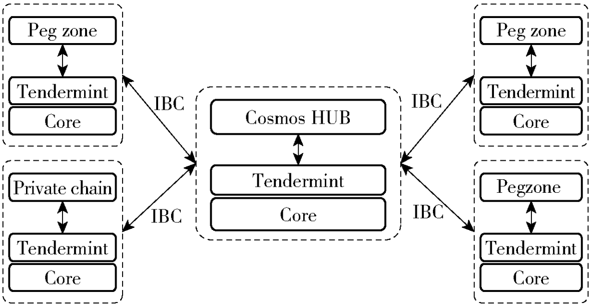

Fig. 10 Cosmos architecture

The Cosmos HUB is the first public blockchain in the Cosmos network and runs through the Byzantine consensus algorithm of Tendermint. Unlike other blockchain consensus systems, Tendermint provides instant, provably secure authentication of mobile client payment. Tendermint is designed to be completely non-forking, so mobile wallets can then receive transaction confirmations in real-time, allowing for a truly de-trusted payment method on smartphones.

Cosmos allows numerous blockchains to maintain interoperability while running in parallel. The Cosmos HUB is responsible for managing the countless independent blockchains known as 'partitions'. The fractions on the HUB will constantly submit the latest block (to synchronize the state of each partition). Each partition is consistent with the HUB in terms of state (partitions do not synchronize states with each other). Merkle proofs are issued to prove that messages have been sent and received so that messages can be passed from one partition to another. This mechanism is called IBC mechanism.

6.4 HyperService

Fig. 11 shows the system architecture of HyperService and Fig. 12 shows its details.

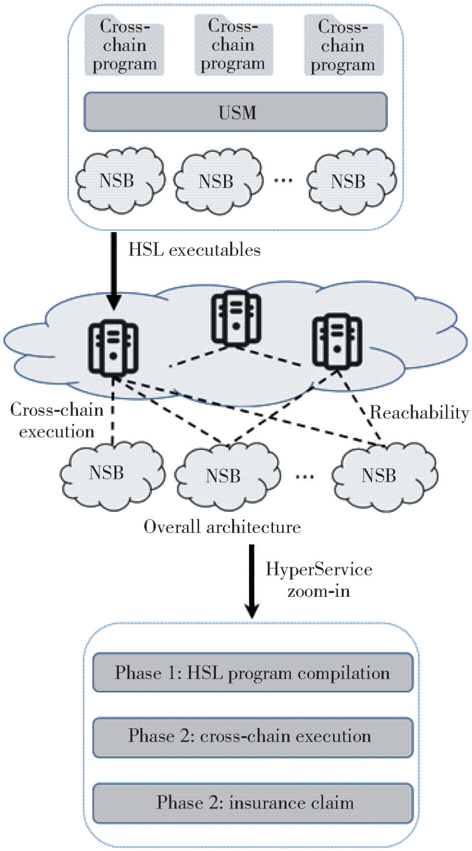

Fig. 11 System architecture of HyperService

HyperService consists of four parts.

1) The decentralized application (dApp) client is used for interaction between the dApp gateway and the HyperService platform. HyperService is designed to make the client lightweight so that it can interact with the HyperService platform through mobile and web applications.

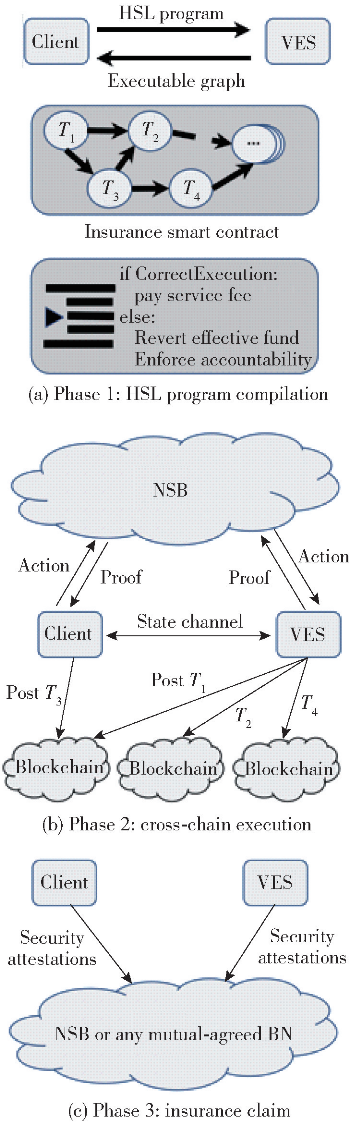

Fig. 12 Details of HyperService architecture

2) Verifiable execution systems (VESs) are conceptually the driver of blockchains and allows access to blockchains to conduct transactions through the dApp client. In other words, files can be executed while HyperService is running. VESs and the dApp client utilize the underlying universal inter-blockchain protocol (UIP) encryption protocol to secure the execution of transactions across different blockchains.

3) Network state blockchain (NSB).

4) Insurance smart contract (ISC). As an extensible model in no relationship to blockchains, it is used to describe state transitions across different blockchains, thereby essentially defining the cross-

chain dApp. Universal state model (USM) realizes virtualization layers to unify underlying heterogeneous blockchains. Its virtualization includes: first, blockchains are abstracted into objects with universal state variables and functionality, regardless of the implementation methods. Second, developers program the dApp by specifying required operations on these objects and specify the relative order of these operations.

To properly execute the dApp, all the transactions included in the executable files must be published on blockchains for execution. Meanwhile, preconditions and deadlines must be complied with. Although this execution is conceptually simple, it can be extremely challenging to enforce the correct execution in the entire absence of trust: first, no trusted authority coordination is allowed to be executed on different blockchains. Second, there is no need to establish mutual trust between VESs and the dApp client. To meet this challenge, HyperService designs an encryption technology, namely UIP. The protocol between VESs and the dApp client can securely execute executable HSL files on blockchains. UIP can work on any blockchain without imposing other requirements.

HyperService is a platform to provide interoperability and programmability across heterogeneous blockchains. It is supported by two innovative designs: one is HSL, which is a programming framework of writing inter-blockchain dApp, different languages are uniformly used to write smart contracts. The other is UIP, which aims to safely design universal interoperability protocols of blockchains so as to implement the complicated operations defined in these dApps. The author of HyperService wrote approximately 35 000 lines of code to use the prototype of HyperService to demonstrate its utility. The end-to-end execution latency of dApp application and the throughput of integration platforms were reported based on this prototype.

HyperService believes that the currently dominant cross-chain technologies are confined to atomic swaps of tokens. Nevertheless, apart from token scenarios, the existing blockchains extend many scenarios through contract programming. Overall, HyperService has the following major constraints in the overall implementation. Firstly, there is a relay chain, relying on the security of the relay chain. Secondly, for each role on the relay chain, the access block chain needs to be publicly accessible. Thirdly, only multiple transactions can be executed on each chain, that is to say, only multiple operations on multiple chains can be executed. Fourthly, cross-chain transactions are 'static' and need to be defined in advance. Fifthly, cross-chain programming is complicated, and its complexity is even higher than the universal implementation of the state channel. One reason is that HSL must be used to write transactions on relay chains and only static transactions can be executed, since there are generally many logic branches in a business, leading to the presence of various transactions which limit application scenarios. The other reason is the virtual absence of cross-chain communication function, which also limits many scenarios.

6.5 HyperLedger Cactus

Cactus was largely contributed by Accenture and Fujitsu. Cactus is designed to support specific application scenarios. Its core idea is to support as many application scenarios as possible through the interoperability of multiple ledgers, particularly in some dominant or other application cases, assess transfer from Ethereum to Quorum, for example.

The goal of Cactus is to allow information to flow between different blockchain platforms. It defines cross-platform blockchain communication protocols and leads to the implementation of Node.js which is a computer language. The client gains access to Cactus through application interface (API). Cactus communicates with various blockchains interactively through ledger connector plugin. Currently, Cactus supports Fabric, Quorum, Corda, and Besu, in addition to its support for public blockchains in terms of architecture design.

The Cactus architecture is shown in Fig. 13. Business logic plugin is responsible for verifying and logging verified transaction information. Verification and logging can be done through local encoding, or message logging and signature verification can be done through smart contracts connected to blockchains by ledger connector plugin.

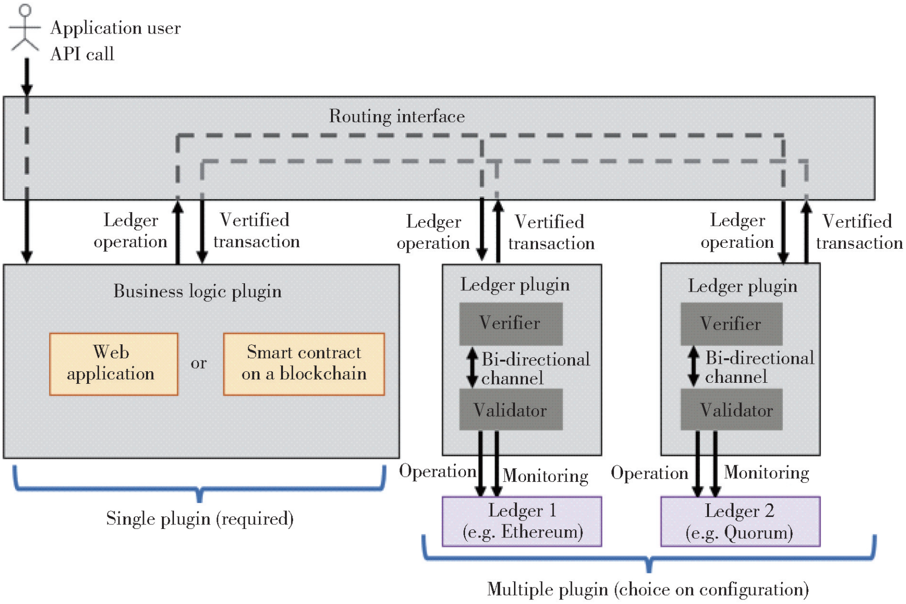

Fig. 13 System architecture of HyperLedger Cactus

7 Cross-chain programmable cases

This section will detail two cross-chain programmable cases that have been implemented in the cross-chain system, analyze the strengths and shortcomings of the programmability, and find out the design focuses of cross-chain programmability.

7.1 Cosmos IBC programmable cases

The Cosmos IBC module provides packets (cross-chain packets) for transmission and authentication between different blockchain systems. It allows definitions of packet structures and deploys sending and receiving modules of packets across multiple blockchains so that the smart contracts on the blockchains can interact with these IBC modules to enable cross-chain communication.

To use IBC modules to customize cross-chain interoperability programming, seven steps are required.

Step 1 To define packet structures.

Step 2 To use SDK to generate interoperability modules of packets.

Step 3 To introduce the concerned IBC packet modules in the smart contracts.

Step 4 To construct packet objects and to dispatch sending interfaces for sending.

Step 5 To enable receiving validation interfaces of packets.

Step 6 To enable receiving interfaces of packets.

Step 7 To enable timeout interfaces of packets.

7.2 ODATS programmable cases

As an cross-chain service platform for Antchains, ODATS deploys on blockchains cross-chain contracts which provide cross-chain API. It enables smart contracts to send and receive cross-chain messages. Smart contracts and cross-chain contracts interface with each other, thereby enabling cross-chain communication of various smart contracts on different blockchains.

To use ODATS to customize cross-chain interoperability programming, three steps are required.

Step 1 To introduce API interfaces of cross-chain contracts in smart contracts.

Step 2 To dispatch inter-blockchain contracts to send messages in smart contracts.

Step 3 To enable message receiving interfaces in smart contracts.

7. 3 Summary of programmability

Both Cosmos and ODATS provide flexible cross-chain programming functions for message communication between smart contracts. However, there are some differences between them. The ODATS platform provides protocol interfaces that are similar to user datagram protocol (UDP) of TCP/IP. It provides underlying communication interfaces, while smart contracts design specific business protocols as well as contents and formats of messages. On the other hand, Cosmos IBC provides functions that are similar to remote procedure call (RPC). Using code generation, it provides encoding and decoding protocols and definitions of message formats. The IBC framework also enhances cross-chain interaction frameworks for packet sending, response, and timeout, thereby providing developers with greater convenience.

Platforms like ODATS are the most flexible in providing interoperability of low-level API. The existing business protocols widely applied in the Internet field are all based on TCP or UDP. Nonetheless, when it comes to some simple scenarios, design with underlying messages would lead to the experience of high entry barriers. Cosmos IBC provides cross-chain interoperability frameworks for code generation. This type of high-level API would significantly reduce application barriers. However, it can be constrained in application scenarios outside of the framework. A desirable means is to provide stacked interfaces which are available for both low-level and high-level API with the latter constructing design based on the former. In this way, interoperability demands for various scenarios can be fulfilled under the premise of maintaining the extensibility of cross-chain interoperability.

8 Conclusions and outlook

The current state of cross-chain technology was summarized from the two perspectives of the realization of key technologies and the implementation of cross-chain application. Key technologies saw mature advancements, particularly in dedicated cross-chain protocols for cross-chain asset exchange. On the other hand, application solutions of universality were implemented in multiple universal platforms in alliance blockchains. Cross-chain application is comparatively mature, and public blockchains entered the first year of cross-chain application. Although cross-chain technologies used by underlying blockchains are still dominated by the earlier dedicated asset anchoring technology, universal cross-chain protocols and cross-chain platforms will make continuous efforts to promote the public cross-chain ecology and to implement the universal application of cross-chain contracts. Security issues behind universal cross-chains will be the largest challenge for public cross-chain technology. In terms of the application of alliance blockchains, mature industry application will move from single-chain interaction to cross-chain collaboration. In particular, cross-industry cross-chain collaboration of supply chain finance will implement validation of trusted assets and cross-chain interaction of financial services, thereby promoting value circulation of digital assets on a larger scale. On the whole, cross-chain technologies have formed a centralized and fixed mechanism with the trend of modular design. Among them, some of the protocols with mature applications have been approved by relevant organizations responsible for the development of standards. The architecture of cross-chain technologies tends to be more uniform, which is expected to accelerate the evolution of an open cross-chain network that supports the real demand for universal interconnection of all blockchains.

References

1. Back A, Corallo M, Dashjr L, et al. Enabling blockchain innovations with pegged sidechains. https://www.blockstream.com/sidechains.pdf. 2014
2. Buterin V. Chain interoperability. R3 Reports. 2016
3. Hype cycle for blockchain technologies, 2020. Gartner Research, 2020

To p. 26

combined with distributed ledger technology, the audit log data right confirmation method is established. In order to further improve the efficiency of data right confirmation, a parallel data right confirmation processing method is presented.

Trusted data access and authorization protocol is used to realize end-to-end encryption of data application and authorization and whole process ciphertext storage by data storager. The data can be decrypted only after the delegatee was authorized by the delegator. The protocol can easily build a complete data processing system on the premise of protecting data privacy based on public cloud storage system or distributed storage system.

References

1. Nuñez D, Agudo I, Lopez J. Proxy re-encryption: analysis of constructions and its application to secure access delegation. Journal of Network and Computer Applications, 2017, 87: 193 - 209
2. Nuñez D. Umbral: a threshold proxy re-encryption scheme. Malaga, Spain: NuCypher Inc and NICS Lab, University of Malaga, 2018
3. Egorov M, Wilkison M L, Nuñez D. Nucypher KMS: decentralized key management system. arXiv Preprint, 2007, arXiv:1707.06140
4. Zyskind G, Nathan O, Pentland A. Enigma: decentralized computation platform with guaranteed privacy. arXiv Preprint, 2015, arXiv:1506.03471
5. Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search. Advances in Cryptology: Proceedings of the 2004 International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'04), 2004, May 2 - 6, Interlaken, Switzerland. LNCS 3027. Berlin, Germany: Springer, 2004: 506 - 522
6. Shao J, Cao Z F, Liang X H, et al. Proxy re-encryption with keyword search. Information Sciences, 2010, 180(13): 2576 - 2587
7. Waters B R, Balfanz D, Durfee G, et al. Building an encrypted and searchable audit log. Proceedings of the 11th Annual Network and Distributed System Security Symposium (NDSS'04), 2004, Feb 5 - 6, San Diego, CA, USA. Reston, VA, USA: Internet Society, 2004: 1 - 10

(Editor: Wang Xuying)

From p. 17

4. Li F, Li Z R, Zhao H. Research on the progress in cross-chain technology of blockchains. Journal of Software, 2019, 30(6): 1649 - 1660 (in Chinese)
5. Xu Z Y, Zhou X. Survey on cross chain technology. Application Research of Computers, 2021, 38(2): 341 - 346 (in Chinese)
6. Belchior R, Vasconcelos A, Guerreiro S, et al. A survey on blockchain interoperability: Past, present, and future trends. arXiv Preprint, 2020, arXiv:2005.14282
7. Hype cycle for blockchain technologies. 2019 - Chain Interoperability Gartner Research, 2019
8. Scalability, interoperability, and sustainability of blockchains. EU Blockchain Observatory and Forum, 2019
9. What is a zero-knowledge proof (ZKP)? https://blog.chain.link/what-is-a-zero-knowledge-proof-zkp. 2021
10. Liu Z T, Xiang Y X, Shi J, et al. HyperService: Interoperability and programmability across heterogeneous blockchains. Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS'19), 2019, Nov 11 - 15, London, UK. New York, NY, USA: ACM, 2019: 549 - 566
11. Lerner S D. Rootstock platform: Bitcoin powered smart contracts. https://wenku.baidu.com/view/f648f8bcfe4733687f21aa4f.html. 2015
12. Thomas S, Schwartz E. A protocol for interledger payments. https://interledger.org/interledger.pdf? ref = hackernoon com. 2015
13. Zhang F, Cecchetti E, Croman K, et al. Town crier: An authenticated data feed for smart contracts. Proceedings of the 23rd ACM Conference on Computer and Communications Security (ACM CCS), 2016, Oct 24 - 28, Vienna, Austria. New York, NY, USA: ACM, 2016: 270 - 282
14. IBC Ecosystem Working Group. Inter-blockchain communication protocol(IBC). https://github.com/cosmos/ics/tree/master/ibc. 2020
15. Cross-chain Message Passing (XCMP). Polkadot Wiki. https://wiki.polkadot.network/docs/en/learn-crosschain. 2019

(Editor: Wang Xuying)