

InterChain: A Framework to Support Blockchain Interoperability

Donghui Ding

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
dingdonghui@ict.ac.cn

Tiantian Duan

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
duantiantian@ict.ac.cn

Linpeng Jia

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
jialinpeng@ict.ac.cn

Kang Li

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
likang@ict.ac.cn

Zhongcheng Li

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
zcli@ict.ac.cn

Yi Sun

Institute of Computing Technology,
Chinese Academy of Sciences
Beijing, China
sunyi@ict.ac.cn

ABSTRACT

Blockchains have supported applications in a large set of areas, and the demand for a framework by which blockchains interact with each other has emerged. We propose a new blockchain interoperability framework, called *InterChain*, which supports scalable and secure inter-operabilities between any pair of blockchains. We present the architecture of the InterChain, and give a three-handshaking method to achieve the asset transfer between separated blockchains. The framework can be easily extended to other applications of cross-chain transactions besides asset transfers.

1 INTRODUCTION

The blockchain is a distributed database maintaining a growing list of blocks. A timestamp and a link to the previous block are assigned to each block. Blockchains are featured by its ability to resist data tampering, *i.e.*, it is impossible to alter the data once the data is recorded into a block. Because of this property, a blockchain is used as an open distributed ledger that reliably records transactions happening between two parties. This ledger is therefore used for a large set of applications. Blockchains have already supported applications such as cryptocurrencies and asset management. A large variety of independent blockchains have been setup in the past years and are under use nowadays.

However, blockchains need to interact with each other, and thus cross-chain transactions are increasingly demanded. For example, users may need to exchange bitcoins with litecoins by cross-chain transactions. This situation requires different blockchains, whether they possess similar functions or are completely heterogenous ones, can achieve cross-chain transactions in a scalable way.

The problem has been the topic of several works, such as [2], [3] and [1]. However, [2] and [3] fail to describe how cross-chain transactions are conducted between existing blockchains, and [1] suffers from scalability issues, *i.e.*, it only supports one-to-one communications. We propose a new framework, the *InterChain*, which supports scalable communications between any blockchains.

2 OVERVIEW OF INTERCHAIN

InterChain is inspired from the Internet architecture. Each local network may run its own protocols. The Internet protocol suite was

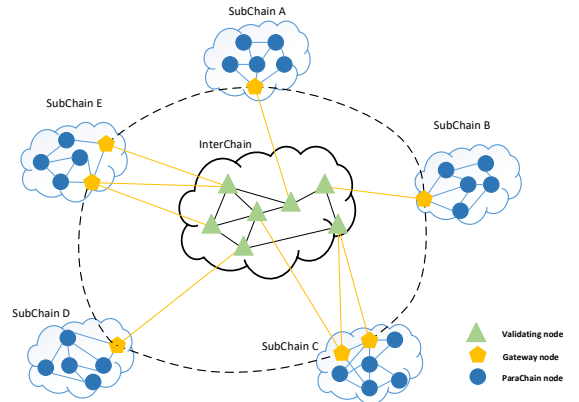


Figure 1: Architecture of InterChain.

then proposed, so these heterogeneous networks are interconnected. We follow the same principle in the design of InterChain. This architecture interconnects separated blockchains in a scalable way. We show in Figure 1 the proposed architecture for InterChain, and give basic definitions before presenting the details of InterChain.

- **SubChain** or a sub blockchain is the blockchain that needs to interact with others by cross-chain transactions.
- **InterChain** is a blockchain that interconnects all SubChains.
- **InterChain nodes** are nodes participating in InterChain. These nodes are categorized into gateway nodes and validating nodes.
- **Validating nodes** participate in the consensus of InterChain. They also collect block headers of SubChains, and validate cross-chain transactions from SubChains.
- **Gateway nodes** belong to both InterChain and a SubChain. If a cross-chain transaction, *e.g.*, t_c , is generated in the SubChain, gateway nodes will send another transaction that relays t_c and the proof for t_c to InterChain. Therefore, t_c will be validated by validating nodes.

When a SubChain connects to InterChain, a part of the nodes in the SubChain network are required to install the clients of InterChain and act as gateway nodes. Gateway nodes then send the registering request to InterChain by these clients. The SPV(Simplified Payment Verification) method to testify transactions from the SubChain is registered to InterChain. Therefore, Given block headers and Merkle proofs for the cross-chain transactions, validating nodes are able to validate the transactions.

Each SubChain holds following smart contracts.

- **Sending Contract:** When a user transfers assets to an account on another SubChain, the user starts with a transaction that deposits the assets in Sending Contract. These assets cannot be withdrawn until a predefined clock times out.
- **Exchange Contract:** It provides a mechanism in which users exchange tokens. If a user Carol holds accounts X_c and Y_c on SubChain X and Y, and wants to exchange 5 tokens issued by Y for 10 tokens issued by X, she is required to deposit 5 tokens to Exchange Contract on Y. An order is also submitted by Carol, which specifies the account addresses Carol holds on X and Y, the number of tokens she deposits, and the number of tokens issued by X that she expects to receive. This contract establishes an order book that collects orders from users on Y.

3 WORKFLOW

A three-handshaking method is utilized to conduct a cross-chain transaction. *i.e.*, a cross-chain transaction used to transfer assets between blockchains is completed through three rounds. In InterChain, a cross-chain transaction is defined by a six-tuple.

< Src Chain, Src Account, Src Tokens, Dest Chain, Dest Account, Dest Tokens >

The six-tuple means Src Account on Src Chain initializes a cross-chain transaction to Dest Account on Dest Chain, and assets in the form of Src Tokens will be removed from Src Account, and subsequently assets with the equal value will be added into Dest Account in the form of Dest Tokens.

3.1 Workflow for Asset Transfer

We use an instance to illustrate how assets are transferred. As is shown in Figure 2, X and Y are SubChains in the framework. Different tokens are issued by X and Y, *i.e.*, XCoins and YCoins. Alice with an account X_a on X sends a cross-chain transaction to Bob with an account Y_b on Y, and 10 XCoins will be transferred from Alice to Bob. As Y_b only accepts YCoins, assets from X_a should be exchanged into YCoins before received by Y_b . Assume Alice expects Bob to receive 5 YCoins. The six-tuple is defined as:

< X, X_a , 10 XCoins, Y, Y_b , 5 YCoins >

The workflow is achieved in the three-handshaking method.

3.1.1 SubChain X to SubChain Y. Alice initializes a transaction, which transfers 10 XCoin to the Sending Contract on SubChain X. This transaction carries the six-tuple, so the transaction is recognized as a cross-chain transaction. Gateway nodes on X monitor the Sending Contract, and relays the this transaction to the InterChain.

As gateway nodes on Y also reside on the InterChain network, they will detect all cross-chain transactions. Once finding the transaction aimed at SubChain Y, gateway nodes on Y will call the Exchange Contract on Y to match the six-tuple against the order book.

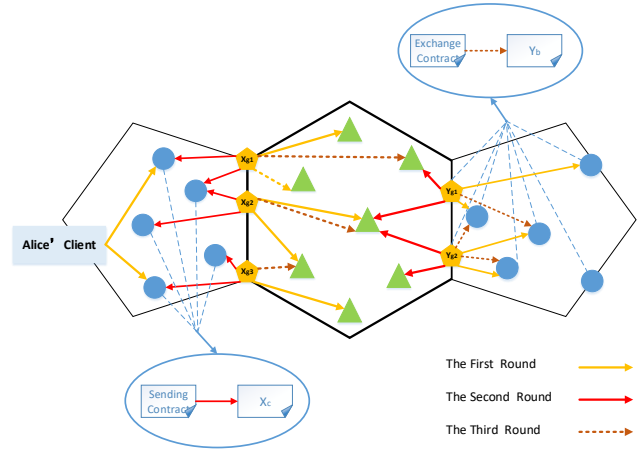


Figure 2: Workflow for Asset Transfer InterChain.

We assume the order mentioned above, *i.e.*, the order from Carol, is matched and selected. This order will be frozen temporarily.

3.1.2 SubChain Y to SubChain X. Gateway nodes on Y publish a transaction in InterChain, by which the selected order is relayed to InterChain. Once detecting this transaction, gateway nodes on X will call the Sending Contract on X. The Sending Contract transfers 10 XCoins that Alice deposited to Carol’s account, *i.e.*, X_c .

3.1.3 SubChain X to SubChain Y. Gateway nodes on X send an acknowledgement transaction to InterChain after Carol receives the coins. Once detecting the this transaction, gateway nodes on Y call the Exchange Contract on Y. The Exchange Contract then transfers 5 YCoins that Carol deposited to Bob’s account, *i.e.*, Y_b . The order that Carol submitted is subsequently removed.

4 CONCLUSION

Mutual interactions between blockchains have been increasingly demanded. Although preliminary solutions to the chain interoperability have been proposed, these solutions either suffer from scalability issues, or do not describe how to concretely realize cross-chain transactions.

We propose InterChain, which supports inter-connections between blockchains. We give the architecture of InterChain, and use a three-handshaking method to realize asset transfers. However, several issues remain to be resolved. For example, an efficient consensus algorithm should be designed to support InterChain. Further more, details for cross-chain transactions should be hidden during the relay so the transactions will not be modified by malicious nodes and the participants’ privacy will not be revealed. We will explore these issues and implement a prototype in the future work.

REFERENCES

[1] A Back, M Corallo, and L Dashjr. 2014. Enabling blockchain innovations with pegged sidechains. (2014).
 [2] Ethan Buchman. 2016. Internet of Blockchains - Cosmos Network. <https://cosmos.network/>.
 [3] Gavin Wood. 2016. POLKADOT: VISION FOR A HETEROGENEOUS MULTI-CHAIN FRAMEWORK. <https://icowhitepapers.co/wp-content/uploads/PolkaDot-Whitepaper.pdf>.