Loopchain: Multi-Channel Smart Contract Support Blockchain

Jay Park
jayhpark@icon.foundation
Contents

I. Blockchain
II. loopchain
III. Multi-Channel
IV. Challenges
I. Blockchain Basic
1.1 Process of blockchain

Life cycle of a transaction: from request to completion

- **Transaction Request**
- **Nodes**
- **Consensus**
  - Transaction validation & approval among several nodes
- **Block Creation**
  - Verified transactions are combined to create a new block
- **Blockchain Expansion**
  - A new block is added to the existing blockchain (permanent, immutable)
- **Blockchain**
- **Expansion**
- **Cryptocurrency, contracts, records**

- **Transaction Broadcast**
A **digitalized business algorithm**: once transformed to a programming, it can be automatically executed without human interference to meet a predefined condition.

## 1.2 Process of Smart contract

### 1. Blockchain

- **More Trust** (than centralized legacy system)
- **Lower Cost** (because of eliminating middlemen)

### Smart Contract

Serverless Application which automatically executes a contract based on the ledger.

<table>
<thead>
<tr>
<th>Agreement</th>
<th>Trigger &amp; Execution</th>
<th>Consensus</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Smart contract" /></td>
<td><img src="image" alt="Trigger &amp; Execution" /></td>
<td><img src="image" alt="Consensus" /></td>
</tr>
</tbody>
</table>

After the specific contract agreed by both parties is met, digitalized business logic will be automatically triggered and executed without human interference.

Finalize the result of execution of contract through consensus between both parties.
DApp (Decentralized Application) requires both on-chain and off-chain service.
II. loopchain
Ledger Database and State Database are separated in one peer for **efficiency** and distributed among several ones for **effectiveness**.

**Counterfeit Proof** of Transaction and Block

**Speed Up** (finalization of a block requires only one confirm)
Self-developed Smart Contract for complex service scenarios (for example, issue certificate in blockchain without CA)

- Separate blockchain core and **container runtime** (smart contract) for stability
- Instead of VM, support **native execution** for performance
2.3 loopchain Overall Architecture

Homogeneous Components ➔ Easy of Administration and Operation
2.4 loopchain Data flow diagram

**Blockchain Nodes**

**Leader(Creation) Node**

**Validation Node**

**Legacy system**

Application

TX

Distribution

Voting

Query

State

WAS

Business logic

DBMS
III. Multi-Channel
### 3.1 Bottleneck of blockchain performance

Too many Txs  Consensus Overhead  Smart Contract Execution overhead

**TPS (Transaction Per Second) comparison**

- **Visa**: 2000~3000
- **Ethereum**: 15~100
- **Bitcoin**: 5~7
- **Google**: 60000~70000 queries

**Note (cf):**
3.2 Solution(1)

III. Multi-Channel

Scale Up: vertical scaling

Fewer, Large server (add more CPUs, RAMs, and HDDs in one server)

Scale Out: horizontal scaling

More, small server (add more servers to the server farm)
3.3 Solution(2)

And Scale Out in the Software

Serialization (execute one by one)  Parallelization (execute group by group)
3.4 Raiden Network

**Off-chain scaling solution** for performing **ERC-20** token transfers on the Ethereum network. (cf) Bitcoin’s Lightning Network (low-fee, scalable, privacy preserving payment)

### Micro Payment channel technology

**Pros:**
- Speed: No consensus, No confirmation
- Low Fee: especially a few dollars or even a cent
- Scalability: linear scaling with # of users
- Privacy: private transfer (off-chain)

**Cons:**
- Token locked up: during the lifetime of the payment channel
- Token only – We need Smart Contract!
3.5 Plasma

Off-chain scaling solution for performing fast smart contract on the Ethereum

cf) Bitcoin’s Segregated Witness (SegWit) (eliminates unnecessary data in smart contracts)

Pros:
- Speed: delegation of complex operations to children
- Low Fee: dependent on only small block producers
- Resource: elimination unnecessary data to save CPU power and storage
- Scalability: distribution of data storage

Cons:
- Too many “exit transaction”: hard pressure on the main blockchain to process enormous final Txs from child blockchains
Horizontal Partitioning of data
cf) scale–out

• Tx data are divided and distributed into multiple servers → performance
• Total number of Tx data per storage is reduced → scalability

• Considerations:
  • Sharding rule
    • Static vs Dynamic
  • Rebalancing
    • expansion or reduction
    • dynamic rule
Virtual Blockchain system: Tx, consensus, and Smart Contract Execution for different business purposes

3.7 expandability of loopchain: Multi-Channel

Virtualization: Single peer, Multi Blockchain → easier deployment

Access control: isolation per channel → more secure
Shard + Multi-Channel: ICX is base coin for ICON - all the SCORE depend on ICX so introduce not Multi-Channel but Shard.

Each SCORE states is independent of each other so introduce Multi-Channel.
3.9 Multi-Channel and Routing

Routing mechanism: address translation using a dynamic routing table to find out the target channel.

Answer: routing table

<table>
<thead>
<tr>
<th>address</th>
<th>Dapp</th>
<th>Channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>hx1234</td>
<td>A</td>
<td>1</td>
</tr>
<tr>
<td>hx2134</td>
<td>B</td>
<td>1</td>
</tr>
<tr>
<td>hx3145</td>
<td>C</td>
<td>2</td>
</tr>
<tr>
<td>hx4234</td>
<td>D</td>
<td>n</td>
</tr>
</tbody>
</table>

Which Dapp?

- Dapp A
- Dapp B
- Dapp C
- Dapp D

SCORE ch.1
SCORE ch.2
SCORE ch.n
3.10 Multi-Channel and Performance

Performance (theoretically): TPS * # of channel

job partitioning to leverage the modern multi-core architecture

I/O bound task
• Network
• Disk

CPU bound task → Concurrency
• Smart Contract Execution
• Hash Calculation

https://insights.sei.cmu.edu/sei_blog/2017/08/multicore-processing.html
IV. Challenges
BTP (Blockchain Transmission Protocol): interchain protocol between Nexus and other Blockchain.

**Comments**

- Notary channel is based on the Multi-Channel service
- Tx issued by transmitter blockchain is transferred to receiver blockchain via Notary channel
- Nexus check the agreed Tx via Light Client connected to the Nexus
4.2 Multi-Channel interchain(1)

No free lunch: the disadvantages of advantages in Multi-Channel

Question: What if Dapp A calls Dapp C?
Answer: Access Denied.

Notice: each channel is isolated from other channels!

Solution:

No free lunch: the disadvantages of advantages in Multi-Channel
4.3 Multi-Channel interchain(2)

Quiz: Difficult problem to solve: Smart Contract calling between blockchains or channels → How about **query function** to request a state?

Hint: query is not Tx.
Thank you

HYPERCONNECT THE WORLD

The ICON project is building one of the largest decentralized networks in the world.