Monoxide: Scale out Blockchains with Asynchronous Consensus Zones

Present by: Yi-Chen Liu (Leo Liu), Jia-Wei Liang (Jessie Liang)

Agenda

2019/11/18

- Overview
- Purpose and Goal
- System Structure
- Security Discussion

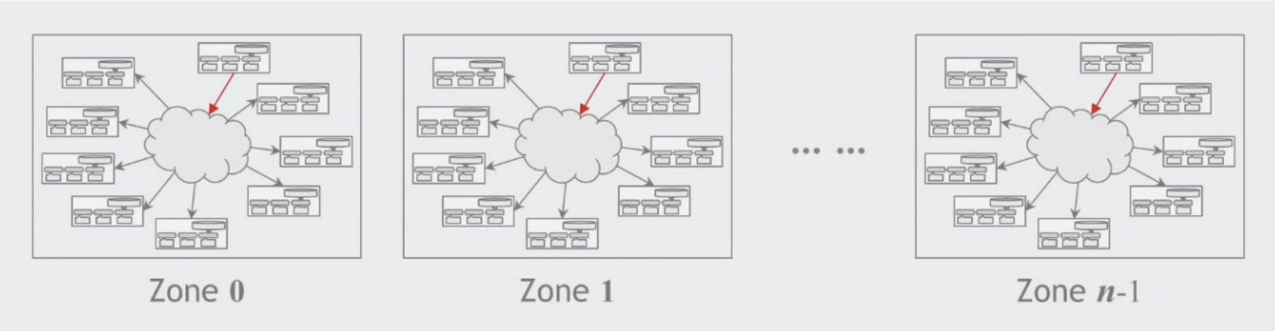
Overview

Overview

Topic 1 : Asynchronous consensus zone => minimize storage and communication

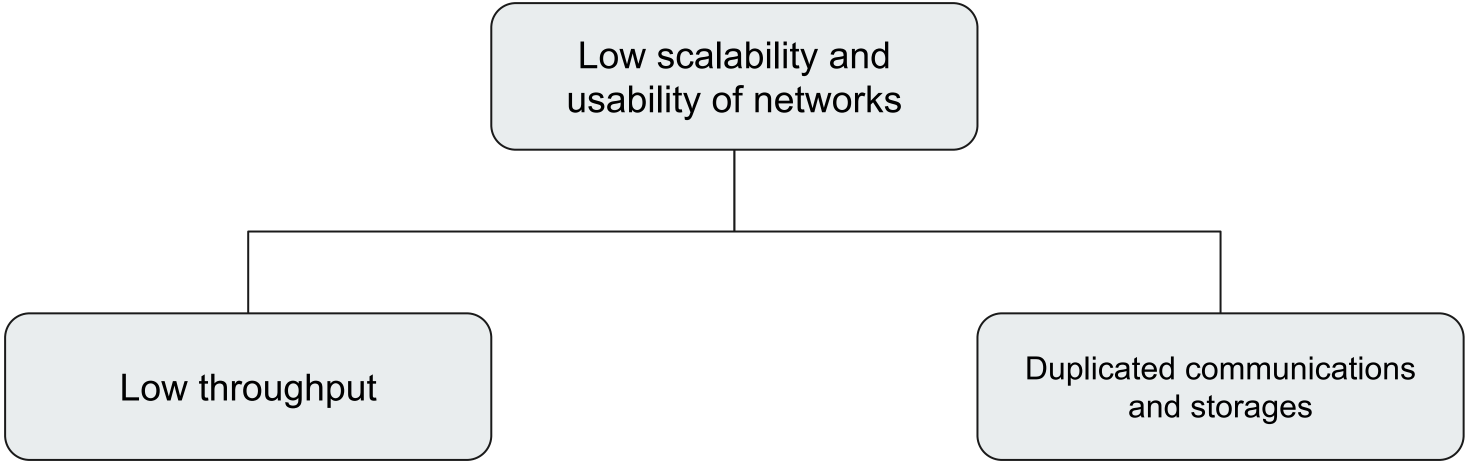
Topic 2: Eventual atomicity => ensure transaction atomicity across zones

Topic 3: Chu-Ko-Nu mining => ensure the effective mining power in each zone to be at the same level of the entire network



Purpose and Goal

Current Flaws of Blockchain


Low scalability and usability of networks

Low throughput

Duplicated communications and storages

Why Scalability Important?

Real-World Applications

- VisaNet: 4K transaction per sec.
- Alipay: 256K transaction per sec.

Cryptocurrency

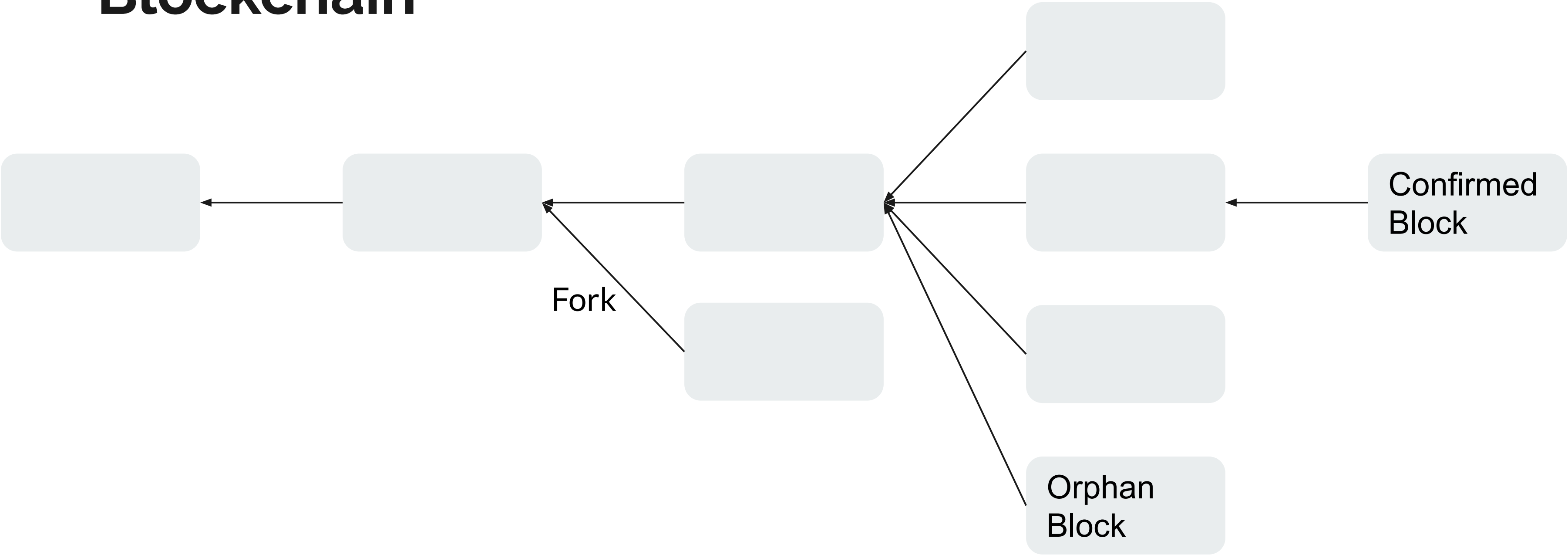
- Bitcoin: 7 transaction per sec.
- Ethereum: 15 transaction per sec.

Goals and Contributions

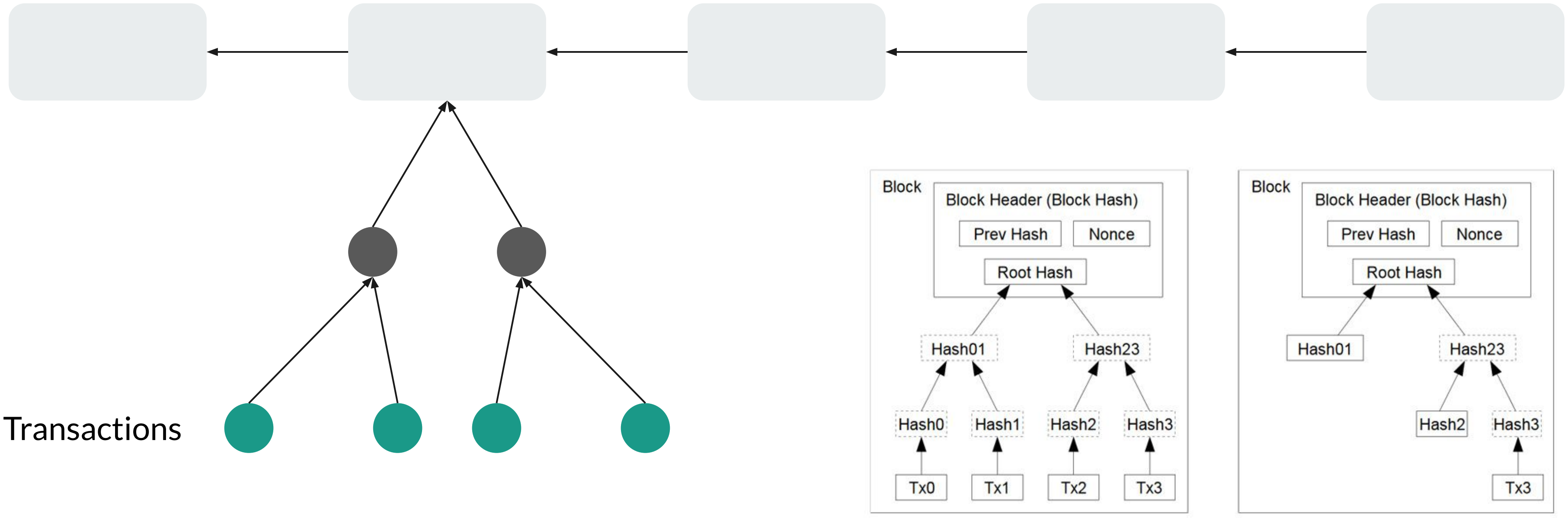
- Lower storage burdens and speed up!
  - Divide the whole network into several sub-network (Zones)
  - Eventual Atomicity principle
- Reinforce system's security
  - Chu-ko-nu mining protocol was introduced

System Structure

Blockchain



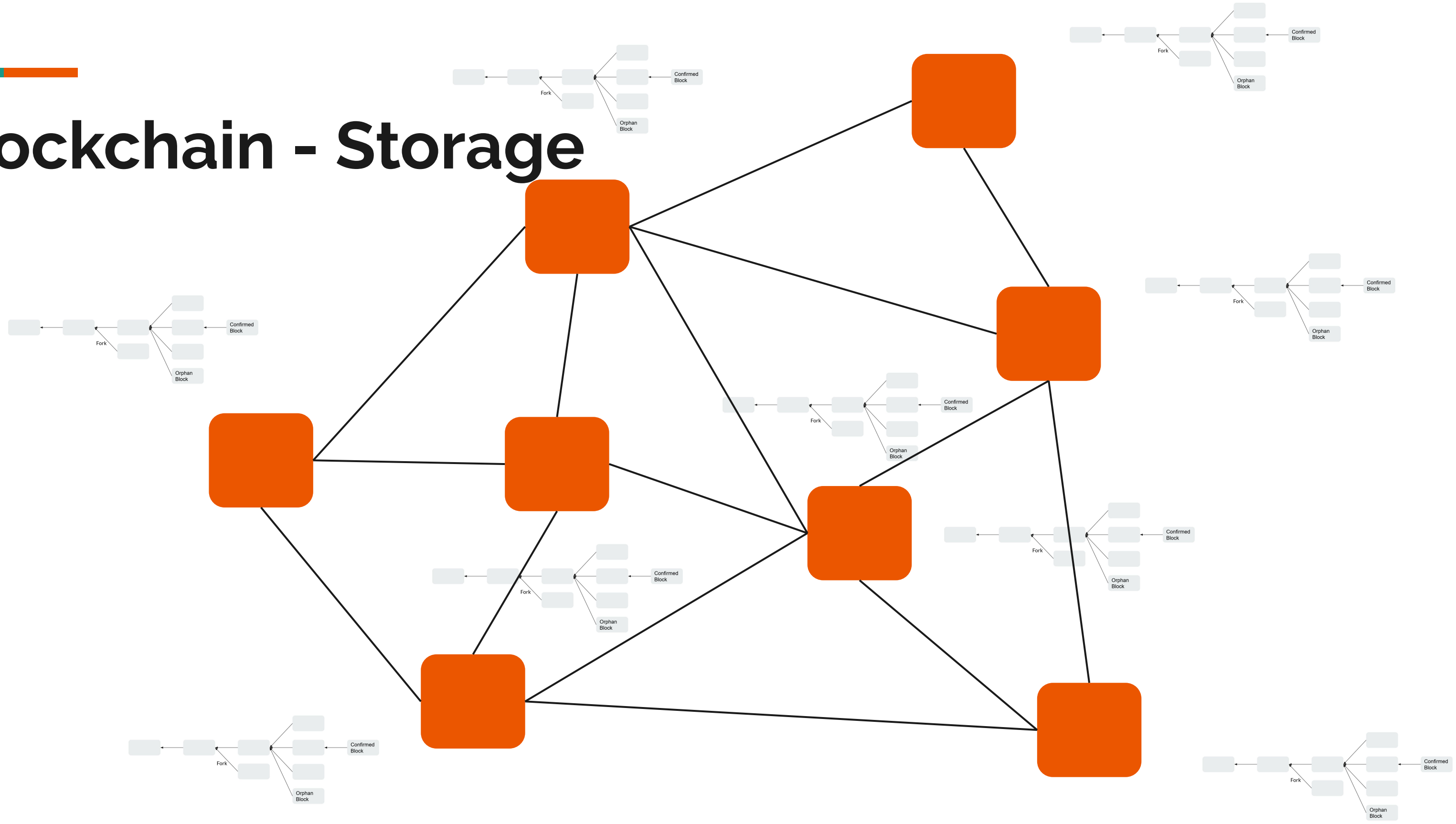
Blockchain - Merkle Tree



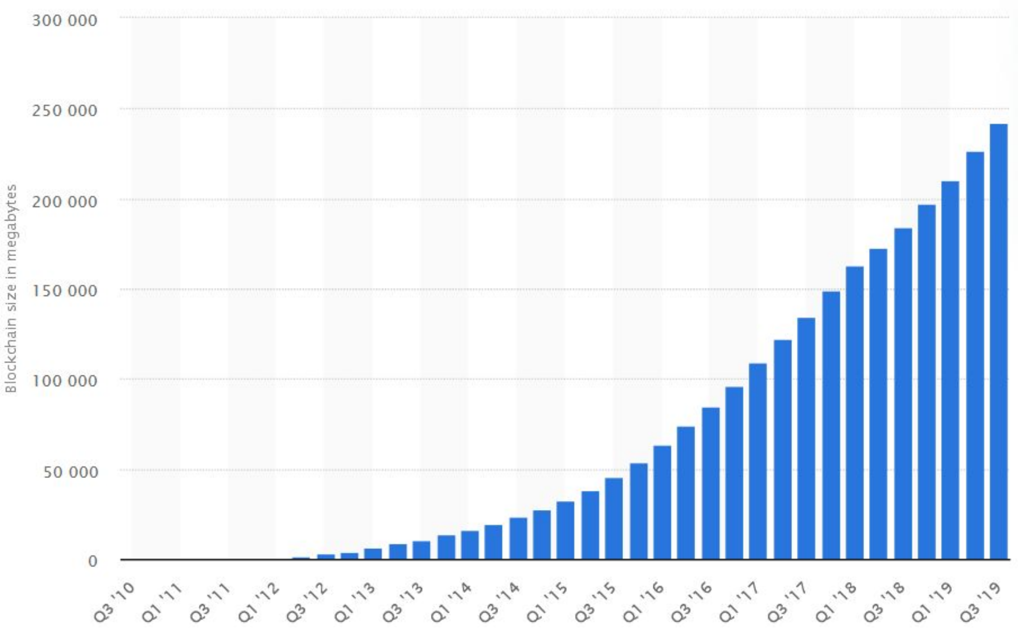
Transactions Hashed in a Merkle Tree

After Pruning Tx0-2 from the Block

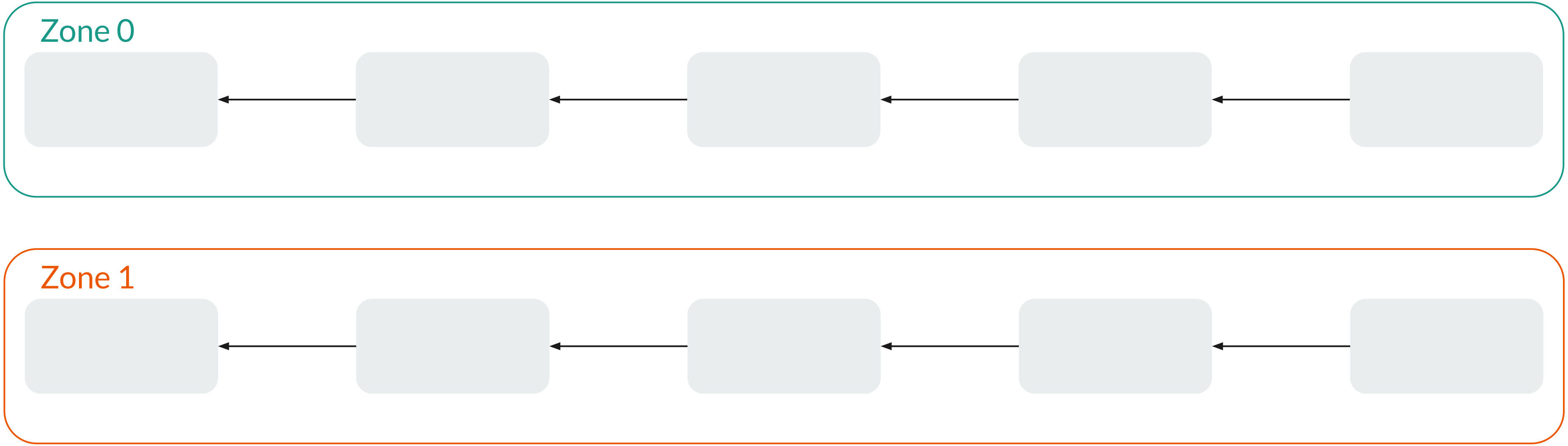
Blockchain - Storage



Blockchain - Size

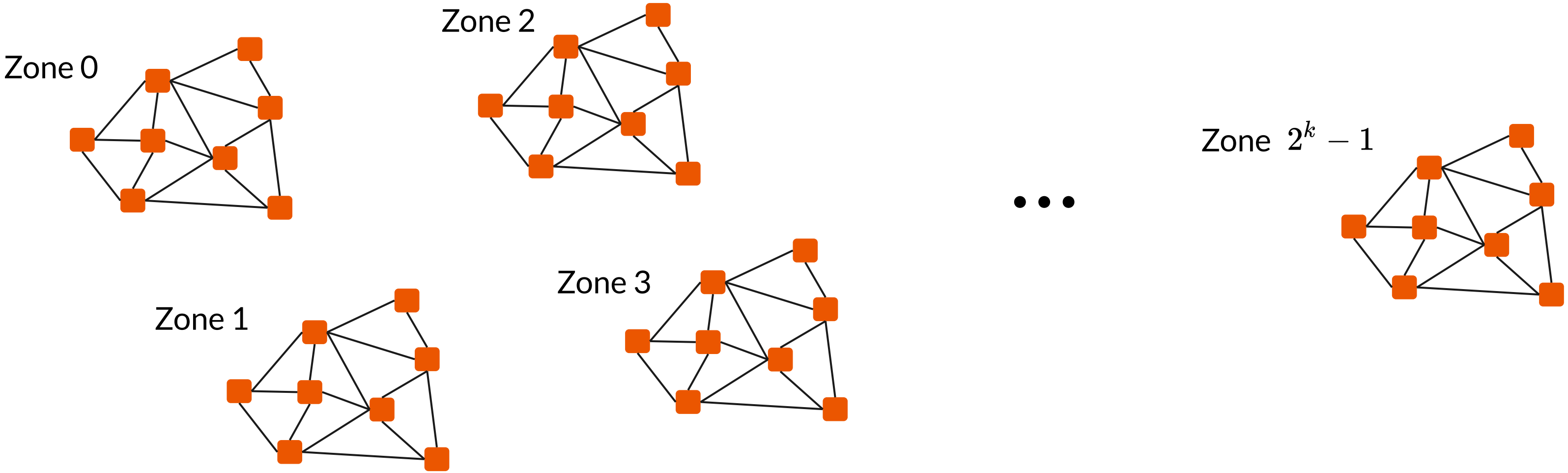


New Approach - Concept



New Approach - Partitioning and Naming

- Nodes' Address = public key
- The first k bit of public key indicate the zone that the node belongs to



Zone 0

Zone 1

Zone 2

Zone 3

...

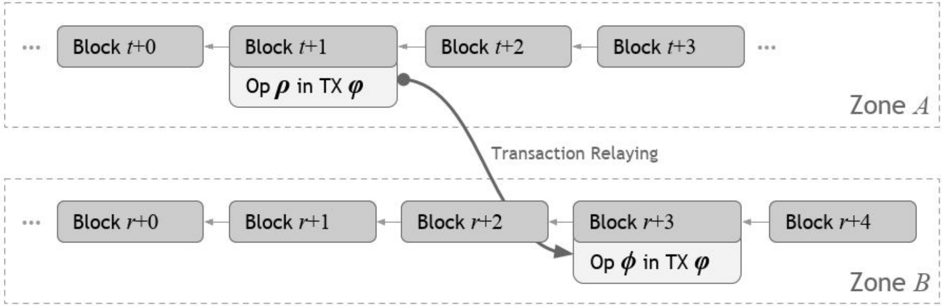
Zone $2^k - 1$

New Approach - Miner's Rule

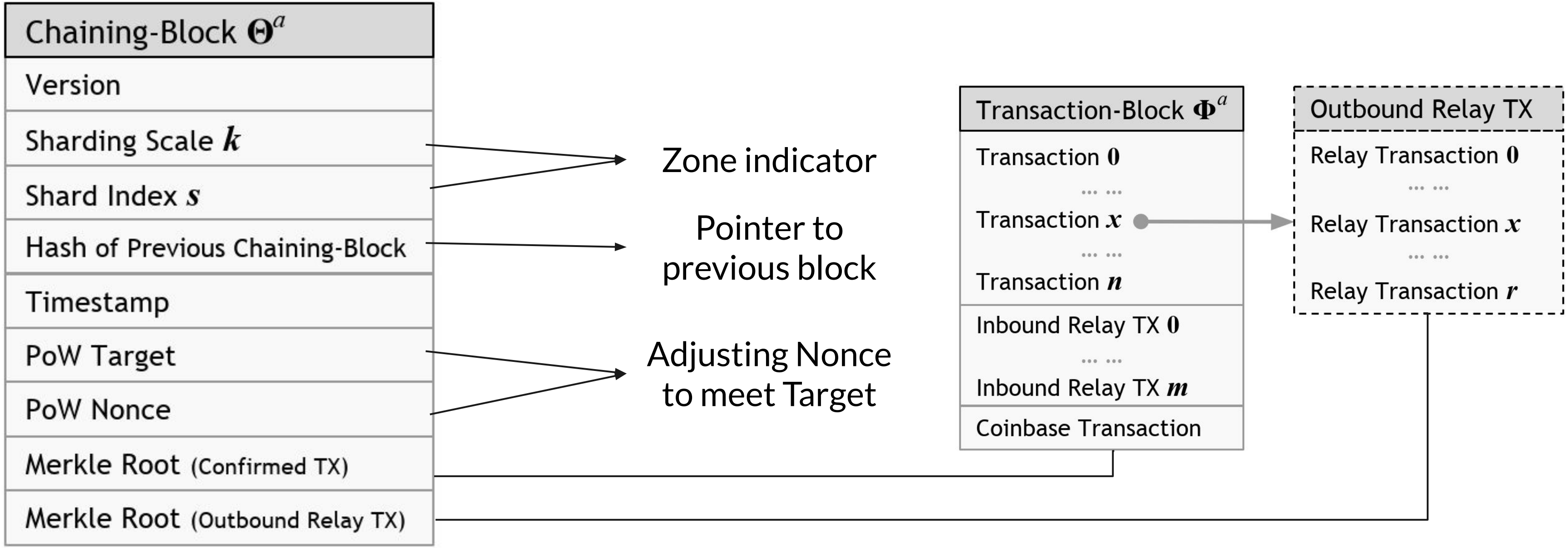
- Only responsible for mining transactions that happen within the zone
- Any full node only records the chain for balances of users in its own zone

New Approach - Simple Transaction Example

- Inner-Zone Transaction: follow the original blockchain approach
- Cross-Zone Transaction: (Zone A) X send $ to (Zone B) Y
    - Miner in Zone A check X’s balance
    - Miner in Zone A create confirm block in Zone A
    - Miner in Zone A create relay block, then send to Zone B
    - Miner in Zone B receive the relay block, then create confirm block in Zone B



New Approach - Detail Block Structure



Chaining-Block $\boldsymbol{\Theta}^a$

Version

Sharding Scale $\boldsymbol{k}$

Shard Index $\boldsymbol{s}$

Hash of Previous Chaining-Block

Timestamp

PoW Target

PoW Nonce

Merkle Root (Confirmed TX)

Merkle Root (Outbound Relay TX)

Zone indicator

Pointer to previous block

Adjusting Nonce to meet Target

Transaction-Block $\boldsymbol{\Phi}^a$

Transaction $\mathbf{0}$

… …

Transaction $\boldsymbol{x}$

… …

Transaction $\boldsymbol{n}$

Inbound Relay TX $\mathbf{0}$

… …

Inbound Relay TX $\boldsymbol{m}$

Coinbase Transaction

Outbound Relay TX

Relay Transaction $\mathbf{0}$

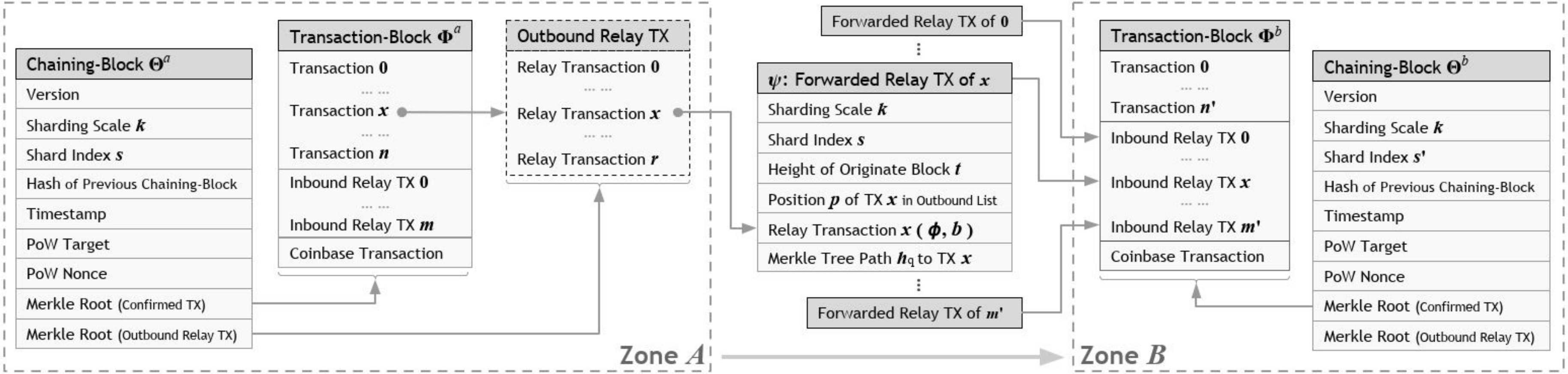
… …

Relay Transaction $\boldsymbol{x}$

… …

Relay Transaction $\boldsymbol{r}$

New Approach - Detail Block Structure (Cont.)



Chaining-Block Θ^a
Version
Sharding Scale k
Shard Index s
Hash of Previous Chaining-Block
Timestamp
PoW Target
PoW Nonce
Merkle Root (Confirmed TX)
Merkle Root (Outbound Relay TX)

Transaction-Block Φ^a
Transaction 0
... ...
Transaction x
... ...
Transaction n
Inbound Relay TX 0
... ...
Inbound Relay TX m
Coinbase Transaction

Outbound Relay TX
Relay Transaction 0
... ...
Relay Transaction x
... ...
Relay Transaction r

Zone A

Forwarded Relay TX of 0
⋮
ψ: Forwarded Relay TX of x
Sharding Scale k
Shard Index s
Height of Originate Block t
Position p of TX x in Outbound List
Relay Transaction x (ϕ, b)
Merkle Tree Path h_q to TX x
⋮
Forwarded Relay TX of m'

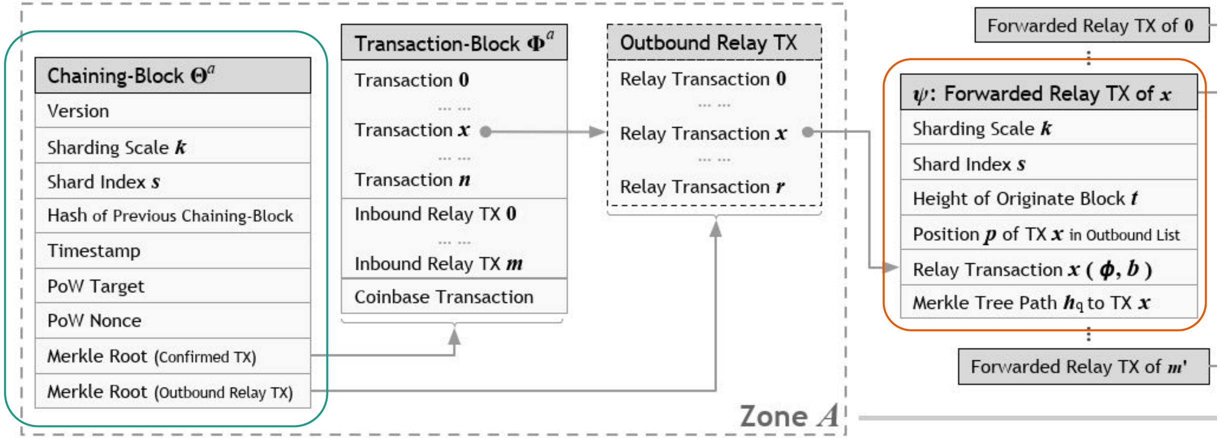
Transaction-Block Φ^b
Transaction 0
... ...
Transaction n'
Inbound Relay TX 0
... ...
Inbound Relay TX x
... ...
Inbound Relay TX m'
Coinbase Transaction

Chaining-Block Θ^b
Version
Sharding Scale k
Shard Index s'
Hash of Previous Chaining-Block
Timestamp
PoW Target
PoW Nonce
Merkle Root (Confirmed TX)
Merkle Root (Outbound Relay TX)

Zone B

New Approach - Transaction Verification

- Attribute set γ should be confirmed and matched with the originate block
    - γ := <s, k, t , p, {hq}>



Chaining-Block Θ^a
- Version
- Sharding Scale k
- Shard Index s
- Hash of Previous Chaining-Block
- Timestamp
- PoW Target
- PoW Nonce
- Merkle Root (Confirmed TX)
- Merkle Root (Outbound Relay TX)

Transaction-Block Φ^a
- Transaction 0
- … …
- Transaction x
- … …
- Transaction n
- Inbound Relay TX 0
- … …
- Inbound Relay TX m
- Coinbase Transaction

Outbound Relay TX
- Relay Transaction 0
- … …
- Relay Transaction x
- … …
- Relay Transaction r

Forwarded Relay TX of 0

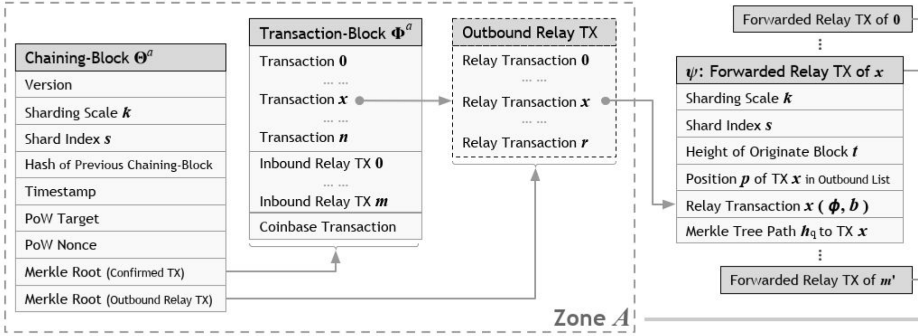
ψ: Forwarded Relay TX of x
- Sharding Scale k
- Shard Index s
- Height of Originate Block t
- Position p of TX x in Outbound List
- Relay Transaction x (ϕ, b)
- Merkle Tree Path h_q to TX x

Forwarded Relay TX of m'

Zone A

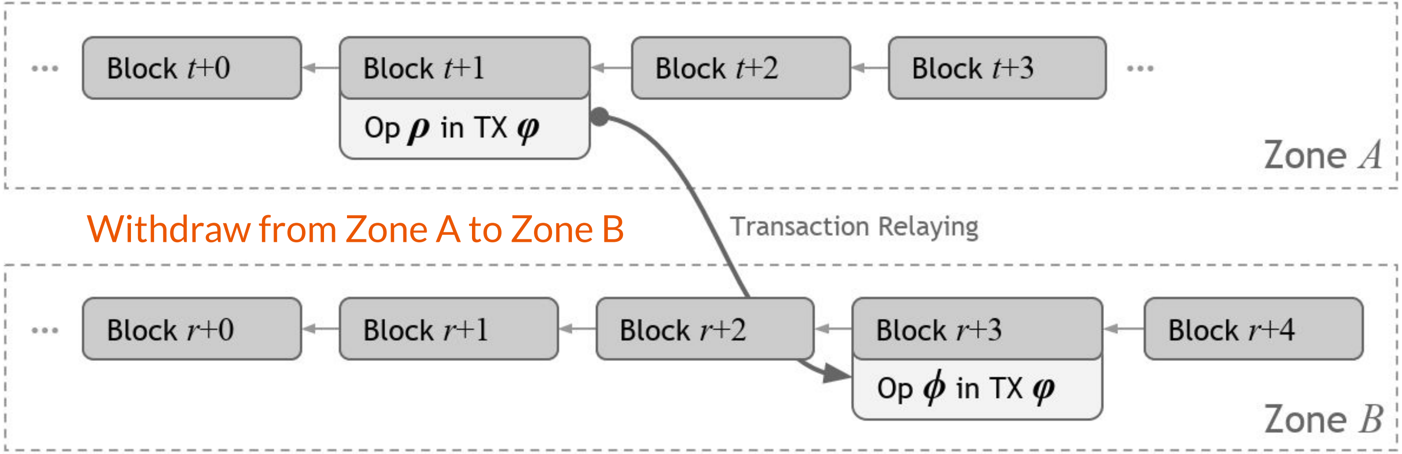
New Approach - Block Verification

- Check 3 types of transactions:
    - Confirmed initiative transactions in its own zone.
    - Inbound relay transactions previously forwarded from other zones.
    - Outbound relay transactions forwarded to other zones.
- Any block containing illegal transactions or mismatched pairs of initiative/relay transactions will be rejected


Chaining-Block Θ^a
Version
Sharding Scale k
Shard Index s
Hash of Previous Chaining-Block
Timestamp
PoW Target
PoW Nonce
Merkle Root (Confirmed TX)
Merkle Root (Outbound Relay TX)
Transaction-Block Φ^a
Transaction 0
Transaction x
Transaction n
Inbound Relay TX 0
Inbound Relay TX m
Coinbase Transaction
Outbound Relay TX
Relay Transaction 0
Relay Transaction x
Relay Transaction r
Forwarded Relay TX of 0
ψ: Forwarded Relay TX of x
Sharding Scale k
Shard Index s
Height of Originate Block t
Position p of TX x in Outbound List
Relay Transaction x (ϕ, b)
Merkle Tree Path h_q to TX x
Forwarded Relay TX of m'
Zone A

New Approach - Eventual Atomicity

- Withdraw first, Deposit Later
  - Assumption 1: once the withdraw operation is confirmed, the deposit operation will be executed.
  - Assumption 2: withdraw operations will be picked as long as there are well-behaved miners



New Approach - Eventual Atomicity (Cont.)

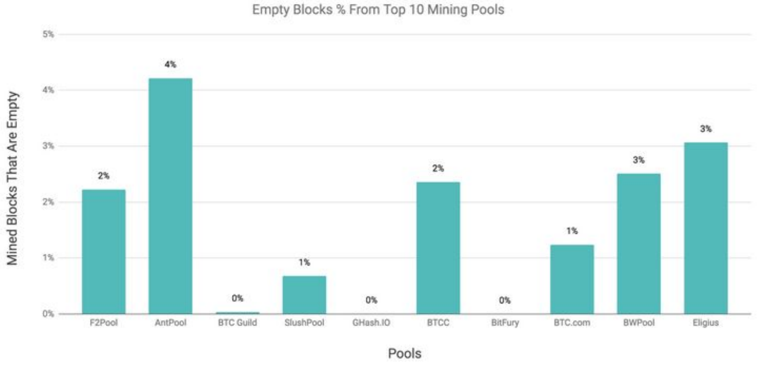
- What if no one picks up the relay block?
  - The relay block will exist eternally unless the originate block has been dropped
- What if the relay block has been dropped accidentally?
  - A new relay block will be generated automatically from the original zone
- Creating multiple blocks for a single transaction means inevitable latency might occur?
  - Mining works between zones are independent

New Approach - Eventual Atomicity (Cont.)

- Malicious Miners. What can we do?
  - Creating empty blocks without confirming any transaction, neither for normal transactions and relay ones.
  - Solution: there will be someone honest to create valid block. Don't worry. And, **the chance is rare!**

True?

~19% !



(Data Source: BTC.com)

Of the current total of 546,237 mined Bitcoin blocks, 101,215 of them were empty blocks.

Security Discussion

Per-Zone Security

- H: The mining power of the entire network
- N: The total number of Zones
- Per-Zone mining power = H/N

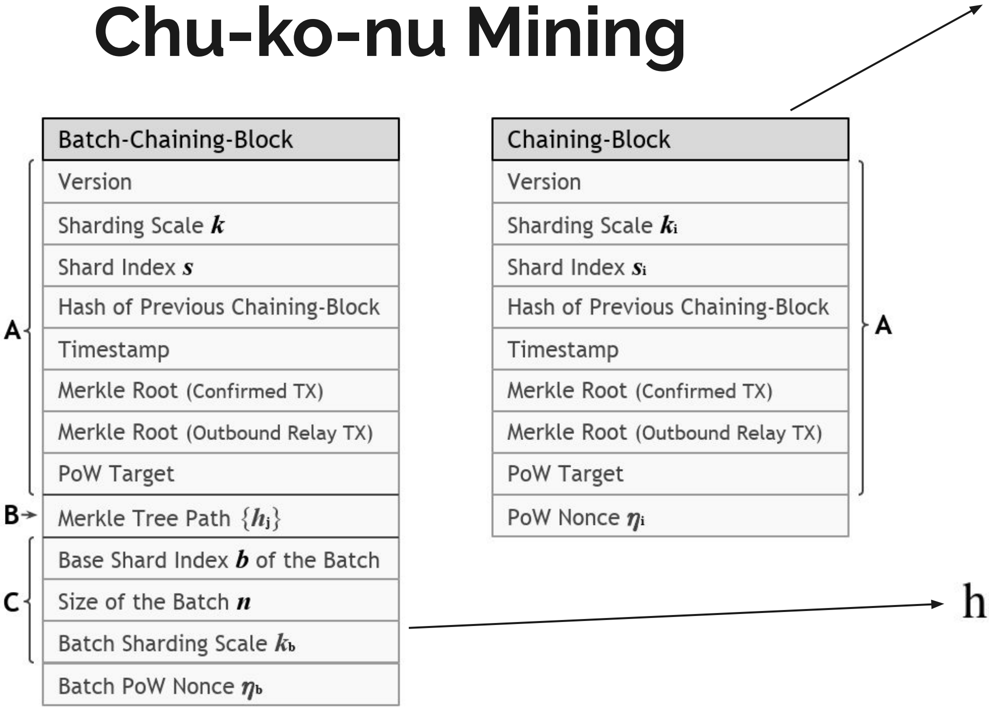
If a malicious participate has T mining power, which T > H/N*50%, the participant can control the zone.

Chu-ko-nu Mining

- Goal: raise the attacking bar in each Zone from H/N*50% to H*50%
- Allow miners create multiple blocks and broadcast to n Zones, where n < N
- To increase the efficiency, miners only need to calculate PoW once
- In each zone, full nodes, as well as miners, treat batch-chaining-blocks and chaining-blocks equally when accepting a new block

51% Attack

Chu-ko-nu Mining



Batch-Chaining-Block
Version
Sharding Scale k
Shard Index s
Hash of Previous Chaining-Block
Timestamp
Merkle Root (Confirmed TX)
Merkle Root (Outbound Relay TX)
PoW Target
Merkle Tree Path $\{h_j\}$
Base Shard Index b of the Batch
Size of the Batch n
Batch Sharding Scale k_b
Batch PoW Nonce η_b

Chaining-Block
Version
Sharding Scale k_i
Shard Index s_i
Hash of Previous Chaining-Block
Timestamp
Merkle Root (Confirmed TX)
Merkle Root (Outbound Relay TX)
PoW Target
PoW Nonce η_i

$$\text{hash}\left(\langle \mathrm{A}_i, \eta_i \rangle\right) < \tau,$$

- τ: PoW target
- b: Zone Index
- η: Nonce

$$\text{hash}\left(\langle h_0, \mathrm{C}, \eta_b \rangle\right) < \tau,$$

Thank You

Present by: Yi-Chen Liu, Jia-Wei Liang