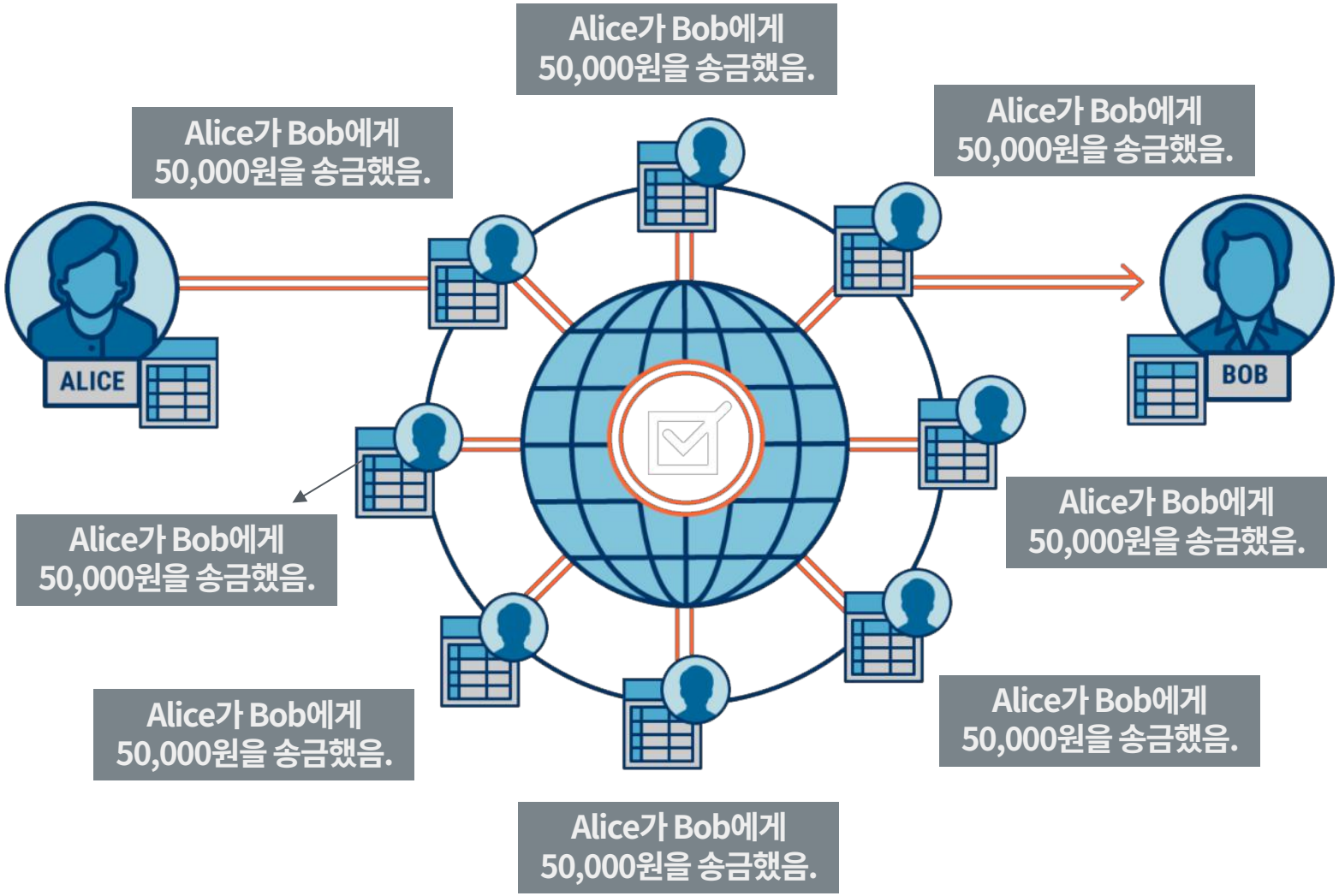
CHAPTER2 합의 알고리즘과 거버넌스

중개자 없는 P2P 거래: **Trustless**

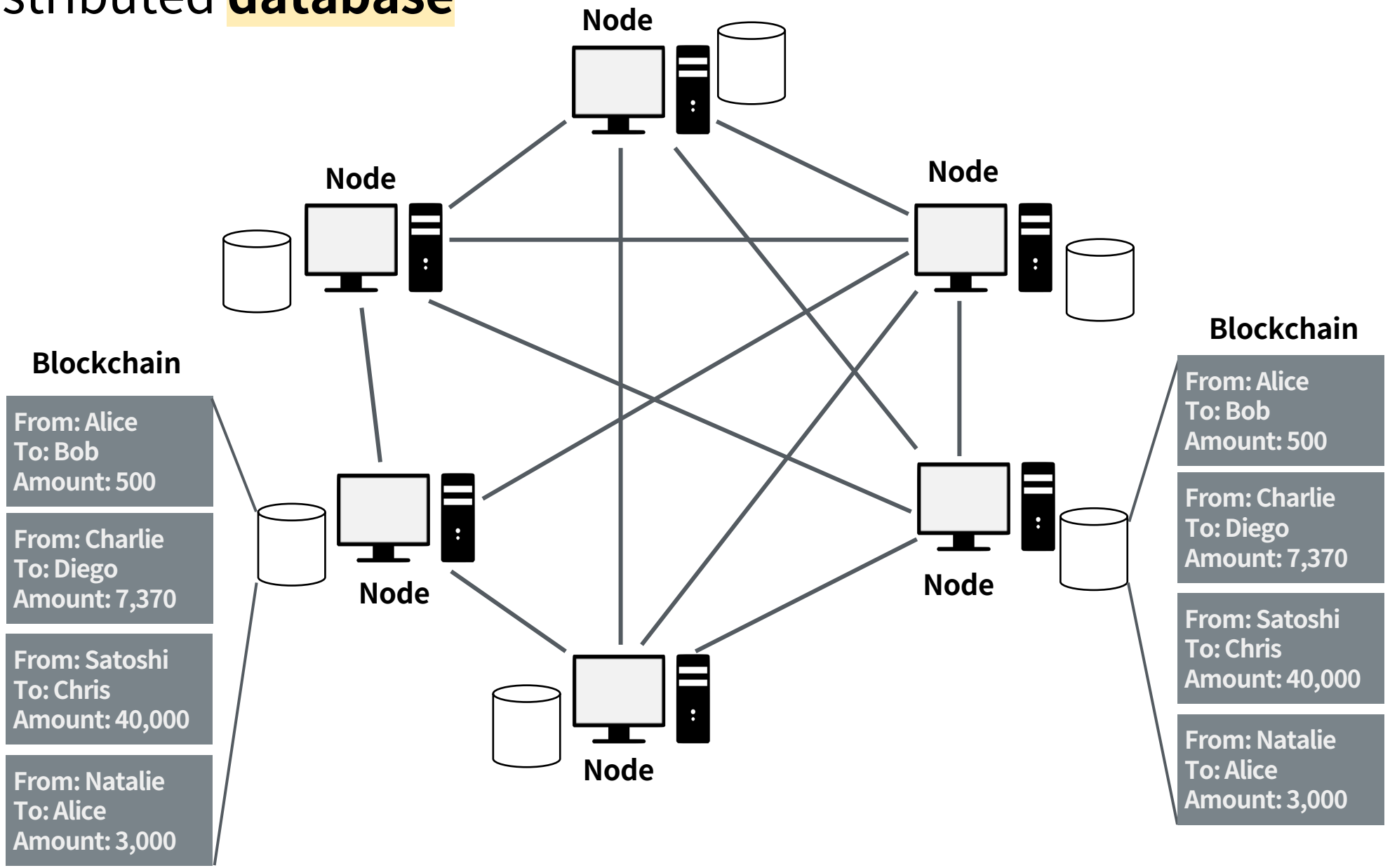
I’ve developed a new open source **P2P** e-cash system called Bitcoin. It's completely **decentralized, with no central server or trusted parties**.

Users hold the crypto keys to their own money and **transact directly with each other**, with the help of the **P2P network to check for double-spending**.

Open distributed **ledgers**



Open distributed **database**



분산 장부의 목표: **동일한 장부**의 유지

언제나 대다수의 노드가 **동일한 장부에 합의**

합의를 위해서는 투표와 같은 **다수결 규칙**이 필요

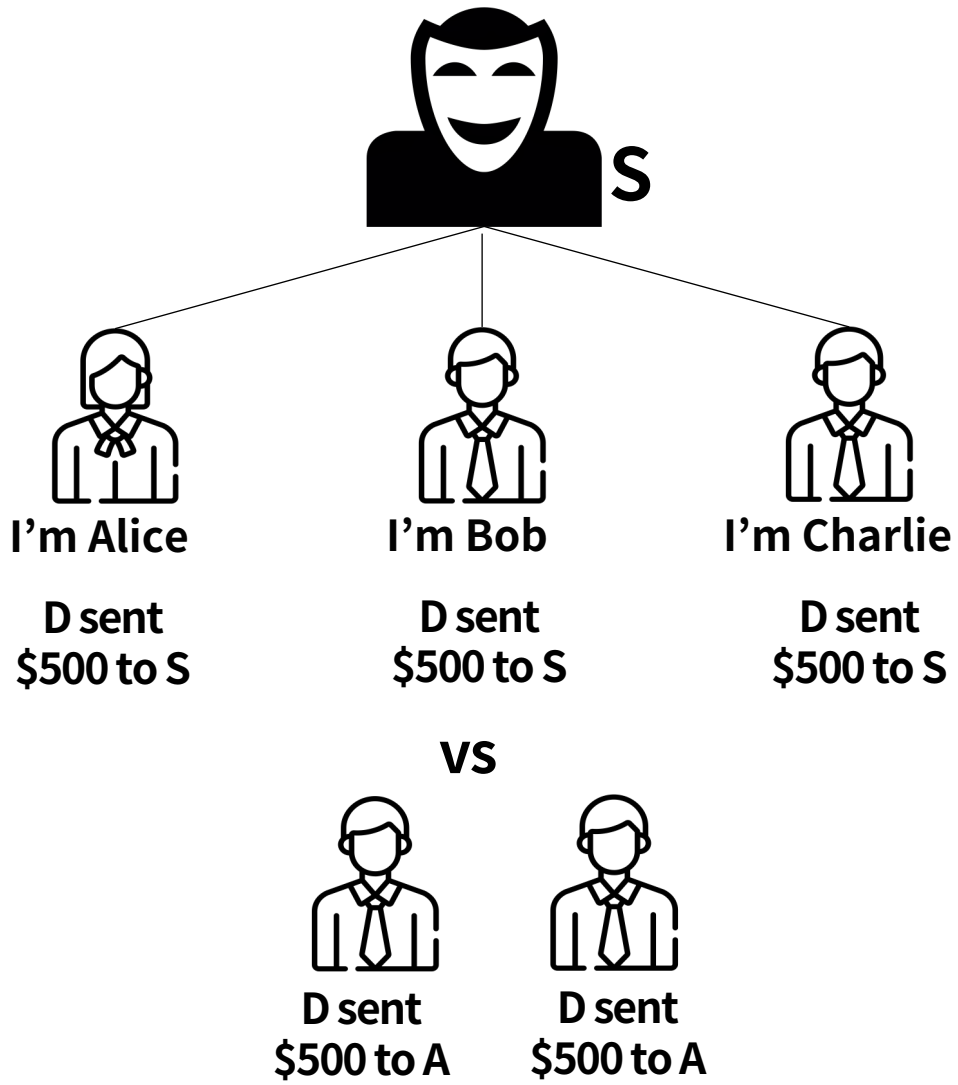
제한된 (폐쇄, 소규모) 네트워크에서 해결

합의 알고리즘

분산 장부의 목표: **동일한 장부**의 유지

Sybil attack

공개 네트워크의 한계



규칙이 필요

Blockchain protocol

누군가가 내 돈을
함부로 쓰지는 않을까

기록하려는 사람이
아무도 없으면?

진짜 동일한 기록일까
아니라고 발뺌하면?

RULES

1.

2.

3.

해결책: 암호학 + 게임이론 + 컴퓨터 과학

누군가가 내 돈을 함부로 쓰지는 않을까	**디지털 서명**
기록하려는 사람이 아무도 없으면?	**기록에 대한 보상**
진짜 동일한 기록일까 아니라고 발뺌하면?	**해시 함수 + PoW**

해결책: 암호학 + 게임이론 + 컴퓨터 과학

누군가가 내 돈을
함부로 쓰지는 않을까

디지털 서명

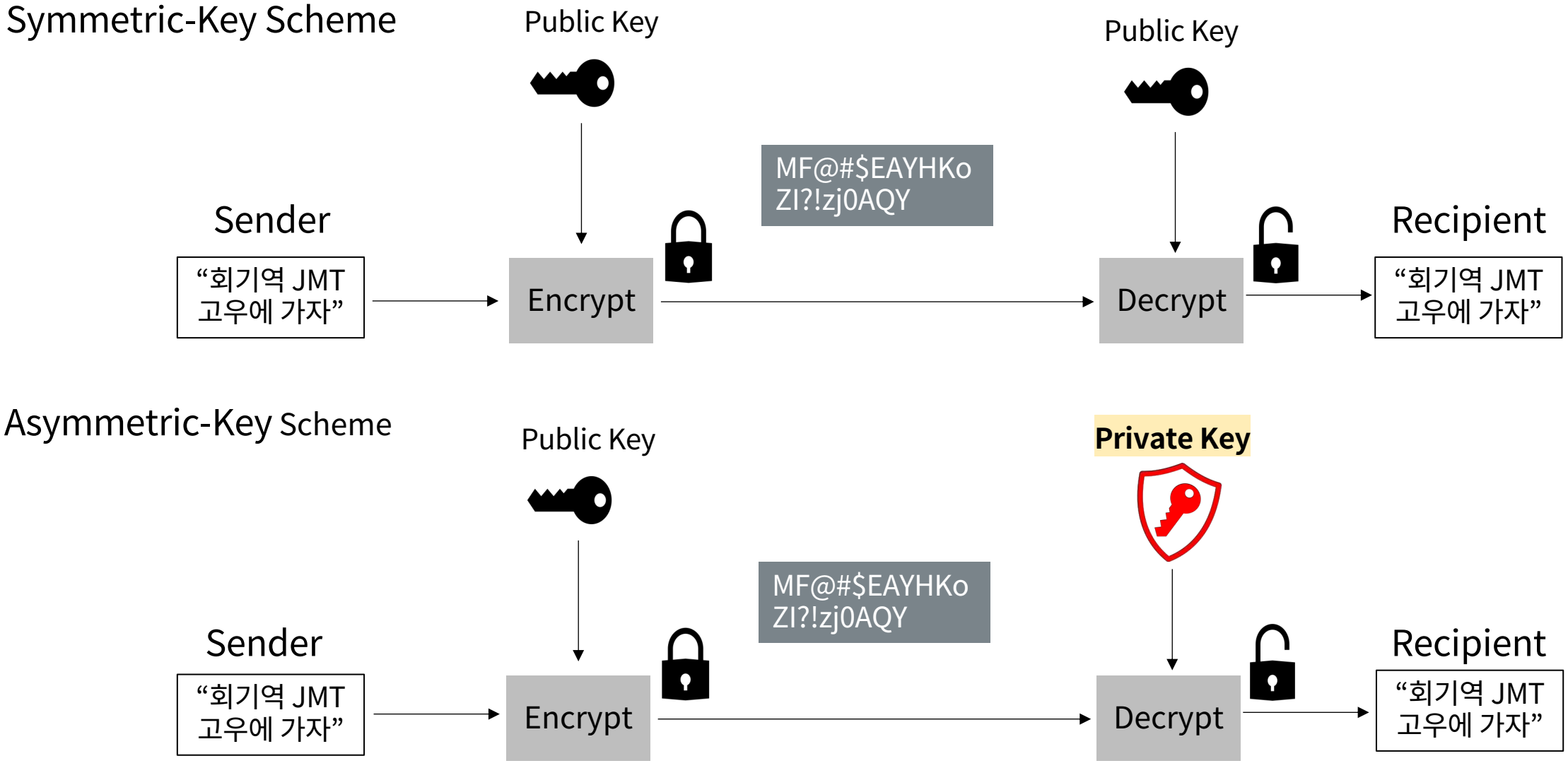
기록하려는 사람이
아무도 없으면?

기록에 대한 보상

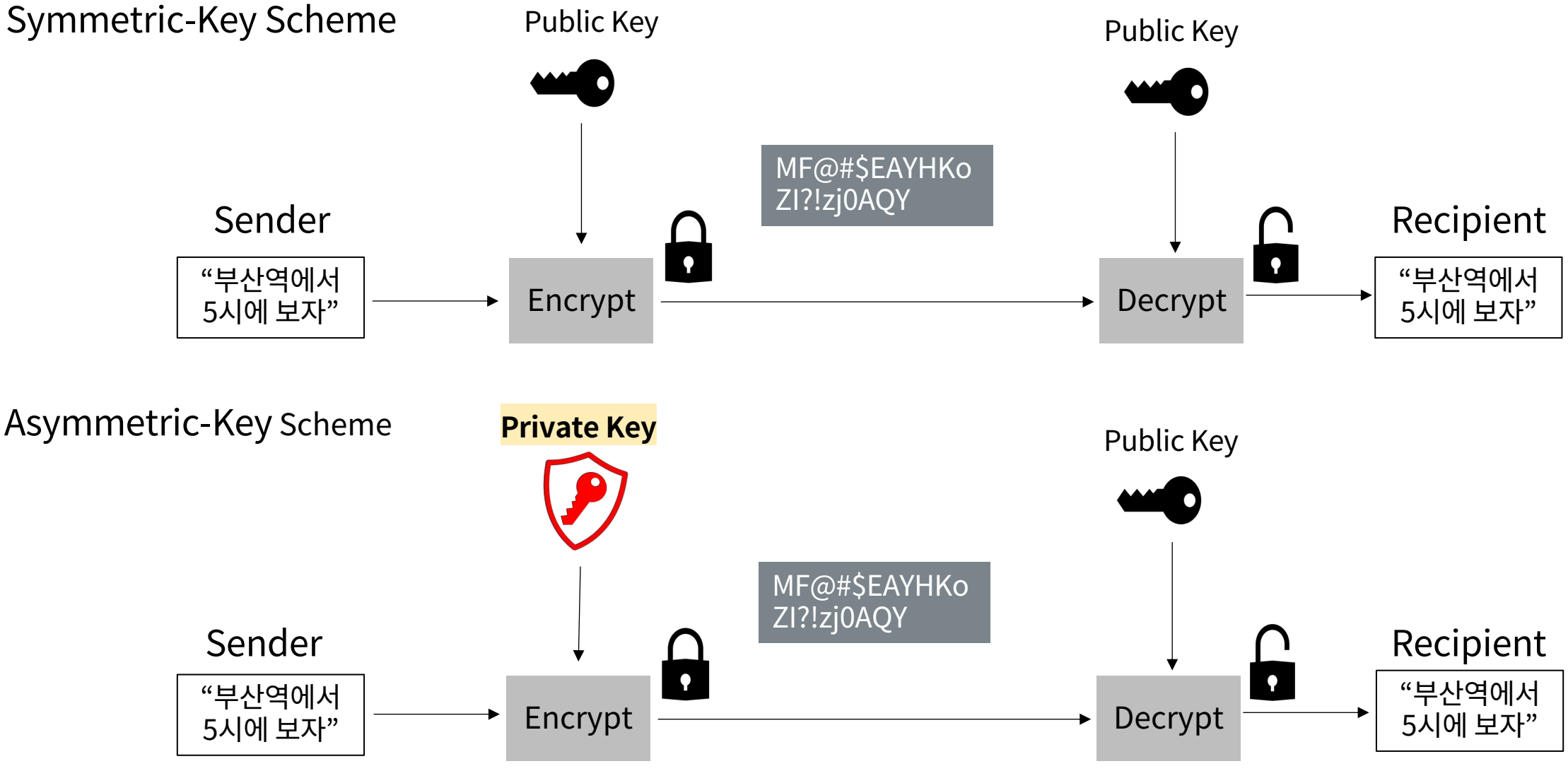
진짜 동일한 기록일까
아니라고 발뺌하면?

해시 함수 + PoW

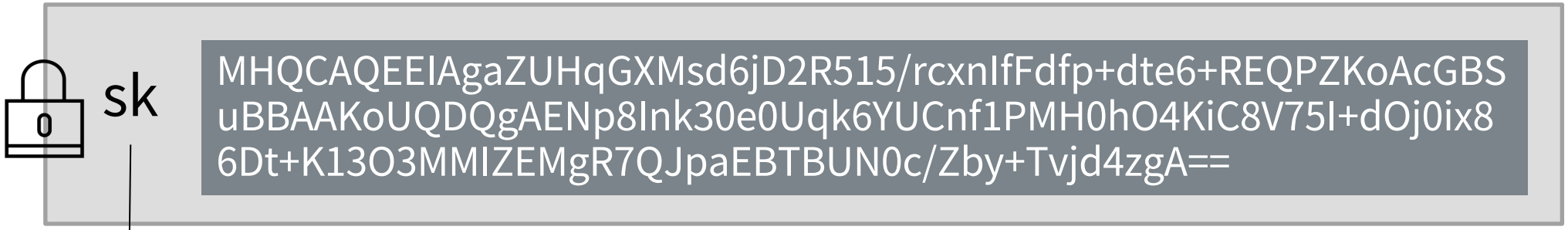
Encryption and Key Scheme (ref #6)



Encryption and Key Scheme



Digital signature: Pk는 Sk로부터 (거의) 유일하게 계산된다 (ref #7)



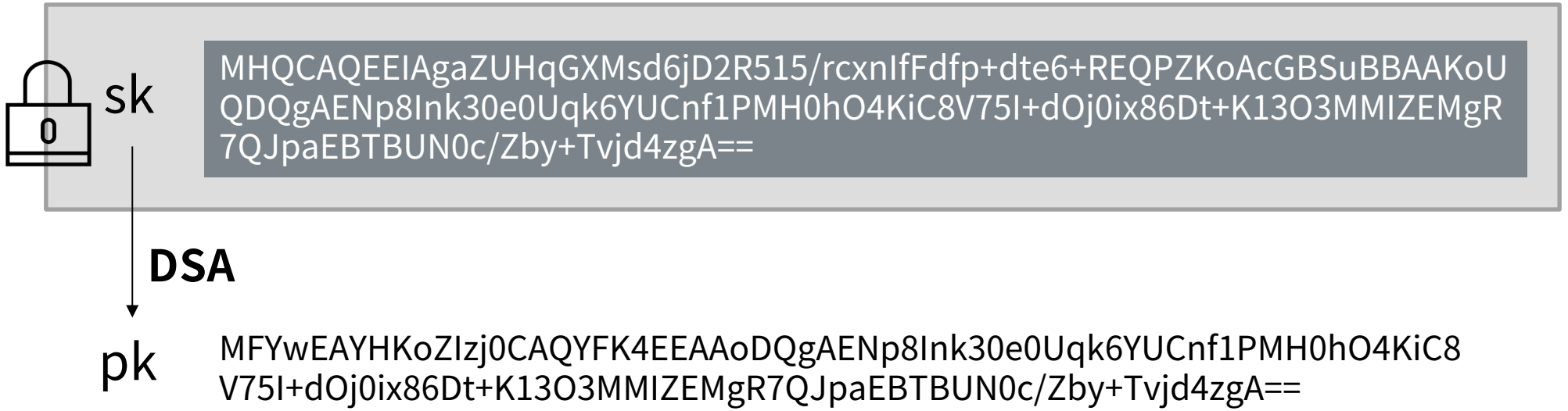
Digital Signature Algorithm

ed25519, secp256k1, secp256r1(P-256)

pk

MFYwEAYHKoZIzj0CAQYFK4EEAAoDQgAENp8Ink30e0Uqk6YUCnf1PM
H0hO4KiC8V75I+dOj0ix86Dt+K13O3MMIZEMgR7QJpaEBTBUN0c/Zby+
Tvjd4zgA==

Digital signature: Sk로 서명하고, 이 서명과 Pk로 검증한다 (ref #7)



Sign(Message, **sk**) = **Signature**

Verify(Message, **Signature**, **pk**) = T/F

Digital signature: 서명은 메시지마다 (거의) 유일하게 결정된다 (ref #7)

Sign(Message, **sk**) = *Signature*

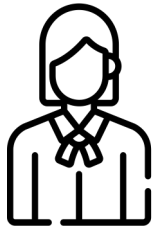
I will give you **500** dollars →  →

MEUCIQD1/HQHlNLUjuUgtcsGU6w/lU9X
uuX7bFPT5BOXOCiA3wIgCfcKeWuwVUOL
u4aIk20C+AfTff/4u2LJLnNquL4NOLs=

I will give you **5000** dollars →  →

MEQCIHGa+a4M4f+BuKmhZqaOFWRzUps
Q3LHti83lSpDHkhQUAiBcZ17eopOlltMuW
ZzFXnxo9AzQT60Y2mQpP39WbDW2Vg==

Digital signature: 동일한 메시지를 다른 Sk로 서명하면 결과가 다르다 (ref #7)



Alice

Pk_A

Sk_A

Bob

Pk_B

Sk_B

Alice sends Bob **500** dollars

Sk_A

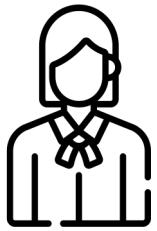
MEUCIQD1/HQHlNLUjuUgtcsGU6w/lU
9XuuX7bFPT5BOXOCiA3wIgCfcKeWuw
VUOLu4aIk20C+AfTff/4u2LJLnNquL4N
OLs=

Alice sends Bob **500** dollars

Sk_B

EMASJAKQa+a4M4f+(^GmhZqaOFWRz
UpsQ3LHti890a7sdkGhQUqwwZ17eop
OlltM56hb!@ADAzQT60Yo98ads21DAD

Digital signature: 서명에 사용된 Sk에 대응하는 Pk만이 True를 리턴한다 (ref #7)



Alice Pk_A Sk_A

Bob Pk_B Sk_B

Sign(Message, Sk_A) = *Signature*

Verify(Message, *Signature*, Pk_A) = **True**

Verify(Message, *Signature*, Pk_B) = **False**

Digital signature

서명은 메시지마다 (거의) 유일하게 결정된다.

동일한 메시지를 다른 Sk로 서명하면 결과가 (거의) 다르다.

서명에 사용된 Sk에 대응하는 Pk만이 True를 리턴한다.

나만이 내 행동(돈)에 대해 서명할 수 있다.
이미 서명한 이상 반박할 수는 없다.

거의 유일하게 = 그렇지 않은 경우가 현실적으로 불가능 (**Infeasible**)

$$2^{256} = (2^{32})^8$$

1600경

≒ (40억) (40억) (40억) (40억) (40억) (40억) **(40억) (40억)**

최고급 GPU ≒ 초당 약 10억 번

GPU 4대(**40억 번**) 장착한 PC **40억 개**

40억 명

40억 개의 지구

40억 초 ≒ 126.8년

126.8년 * **40억** ≒ 5,070억 년

거의 유일하게 = 그렇지 않은 경우가 현실적으로 불가능 (**Infeasible**)

$$2^{256} = (2^{23})^{11.xx} = (2^{18})^{14.xx}$$

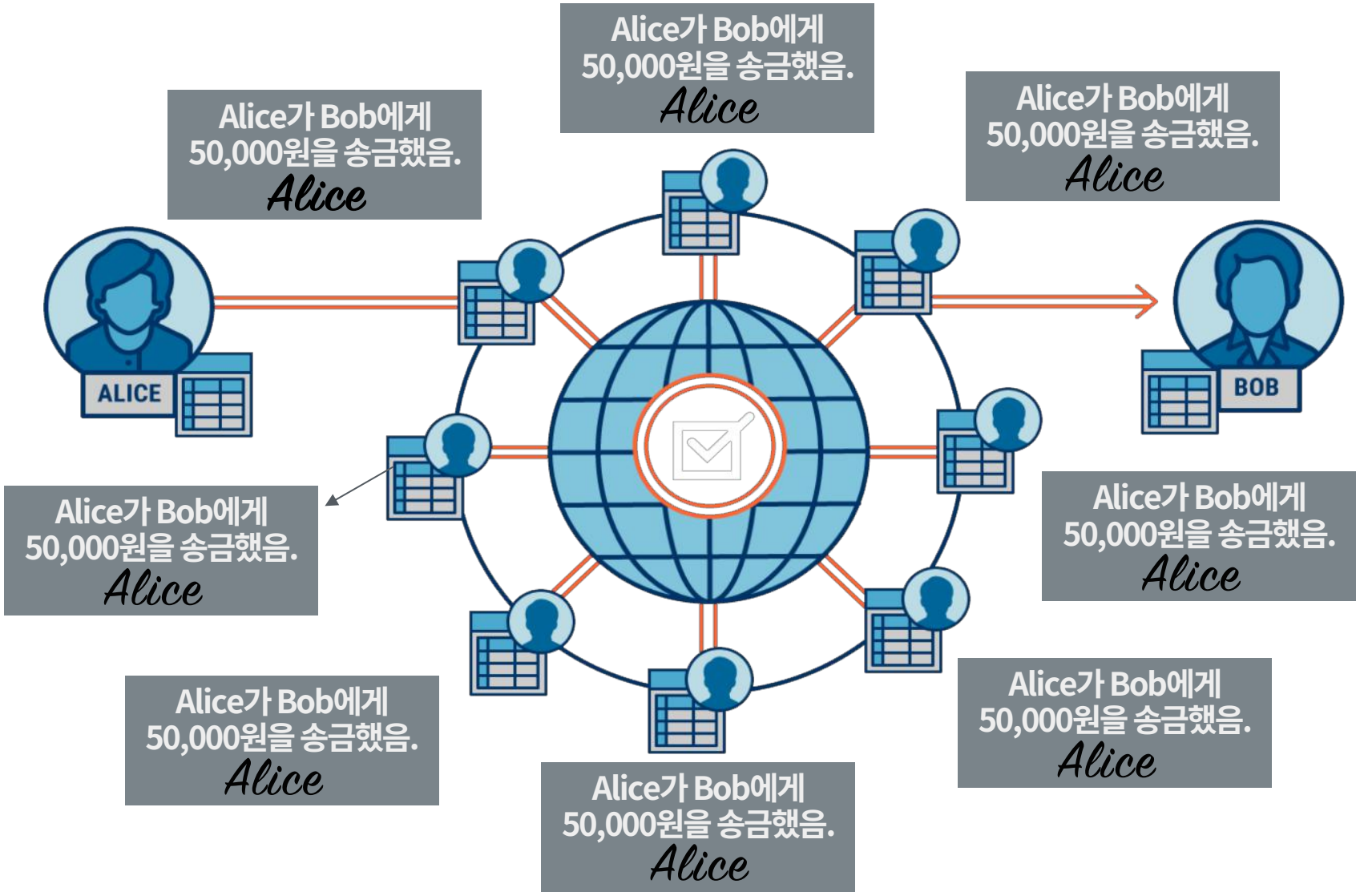
로또 1등 맞을 확률 = $1 / 8{,}145{,}060 \fallingdotseq 1/ 2^{23}$

로또 1등을 연속으로 **11번** 당첨될 확률

벼락 맞을 확률 $\fallingdotseq 1/ 280{,}000 \fallingdotseq 1 / 2^{18}$

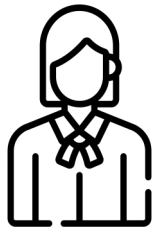
벼락을 연속으로 14번 맞을 확률

Distributed ledgers with digital signature



RULES

1. 비트코인을 송금하는 사람은 거래에 자신의 디지털 서명을 포함시켜야 한다.



Pk_A
Sk_A

Sign('Alice sends 10 btc to Bob', Sk_A) = *Alice's Signature*

해결책: 암호학 + 게임이론 + 컴퓨터 과학

누군가가 내 돈을
함부로 쓰지는 않을까

디지털 서명

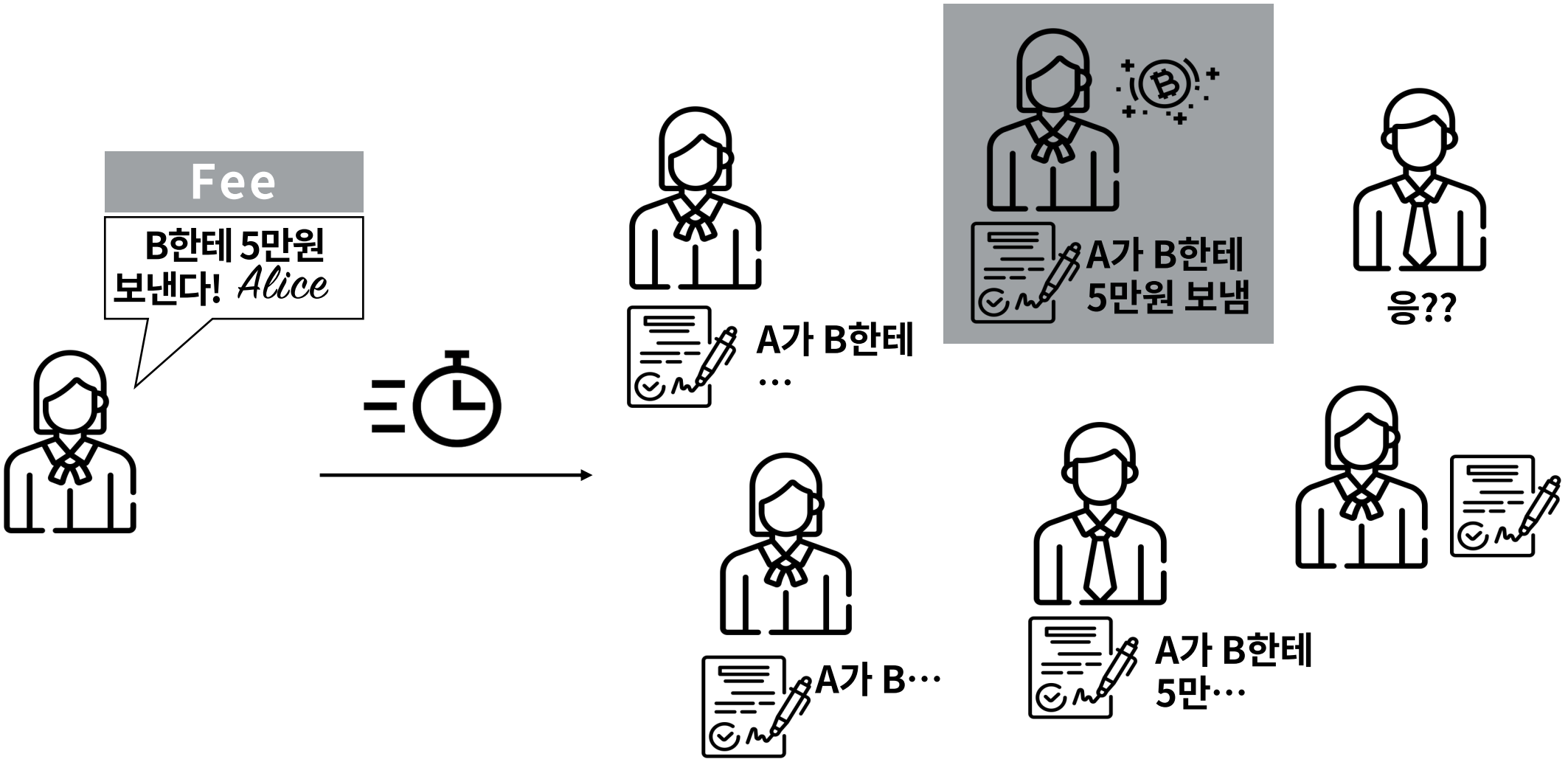
기록하려는 사람이
아무도 없으면?

기록에 대한 보상

진짜 동일한 기록일까
아니라고 발뺌하면?

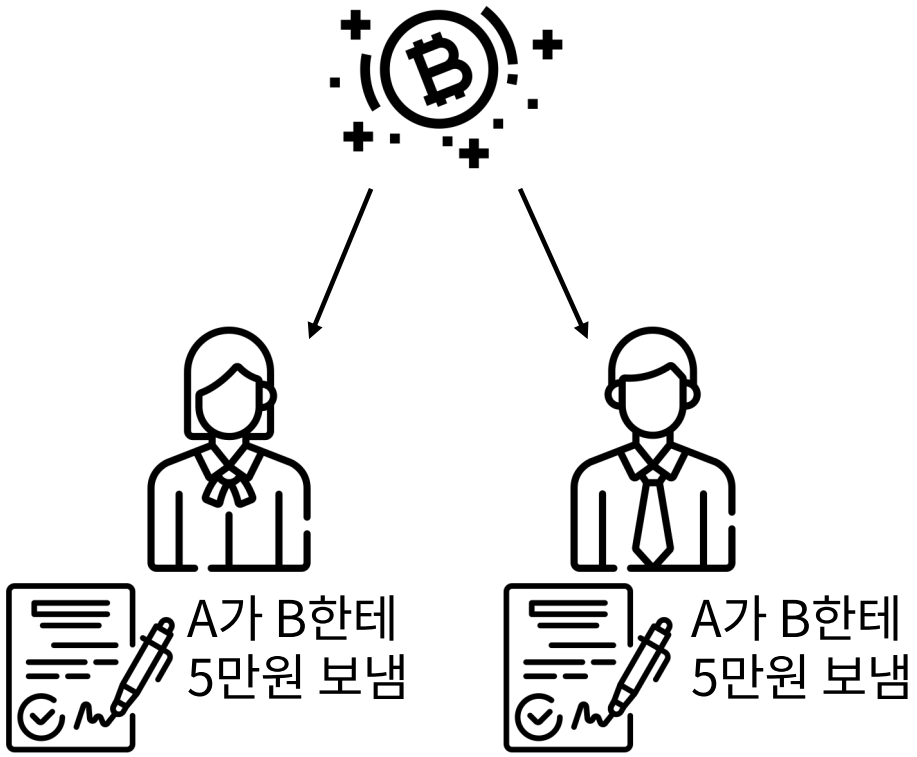
해시 함수 + PoW

제일 먼저 기록한 사람이 보상을 가져간다



제일 먼저 기록한 사람이 보상을 가져간다

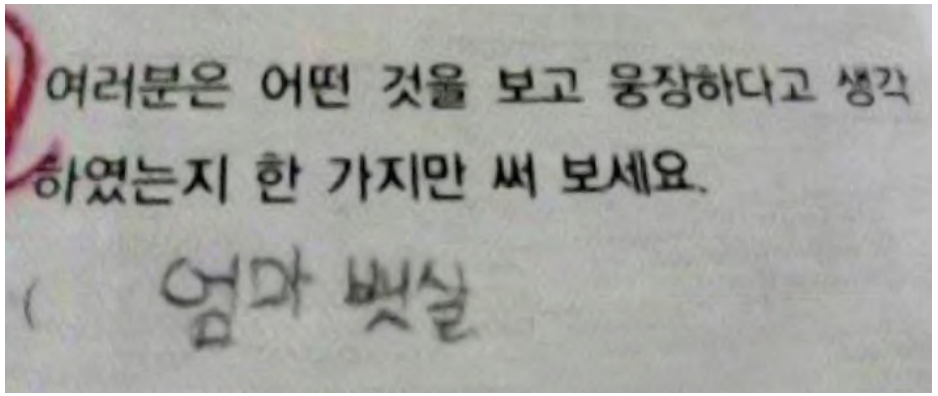
제일 먼저???



경쟁

문제를 **가장 빨리**

맞춘 사람이 승리



RULES

1. 비트코인을 송금하는 사람은 거래 기록에 자신의 디지털 서명을 포함시켜야 한다.
2. 문제를 가장 빨리 맞춘 사람이 기록이 공식 기록으로 인정되고,
기록의 대가로 보상을 받는다.

RULES

1. 비트코인을 송금하는 사람은 거래에 자신의 디지털 서명을 포함시켜야 한다.
2. 문제를 가장 빨리 맞춘 사람이 기록이 공식으로 인정되고, 그 대가로 보상을 받는다.

해결책: 암호학 + 게임이론 + 컴퓨터 과학

누군가가 내 돈을
함부로 쓰지는 않을까

디지털 서명

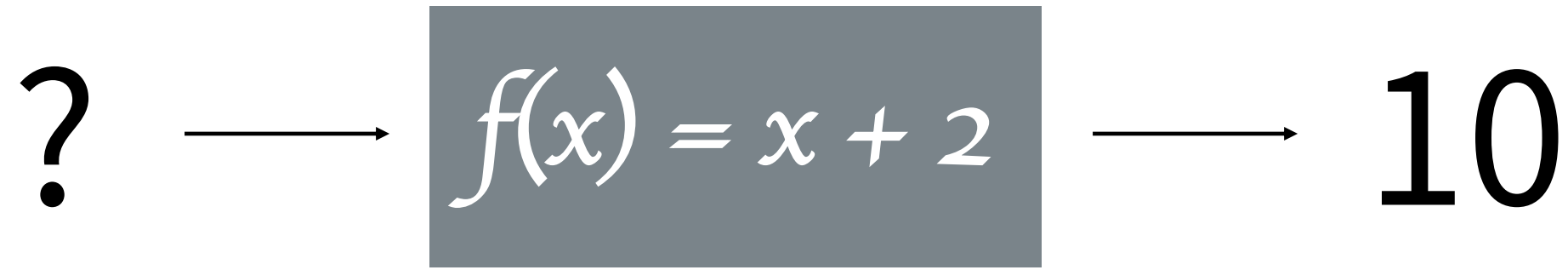
기록하려는 사람이
아무도 없으면?

기록에 대한 보상

진짜 동일한 기록일까
아니라고 발뺌하면?

해시 함수 + PoW

함수의 **결과값**을 보고 **입력값** 맞추기



함수의 **결과값**을 보고 **입력값** 맞추기 (**Hash Puzzle**)

(Cryptographic) Hash Function (ref #8)

임의의 길이의 데이터를 **고정된 길이**의 데이터로 매핑하는 함수

Deterministic

$x = y \Rightarrow h(x) = h(y)$

Cryptographic Hash Function

Collision resistance

같은 결과값을 갖는 서로 다른 입력값을 찾기가 **실질적으로 불가능**

$x \neq y \Rightarrow h(x) \neq h(y)$

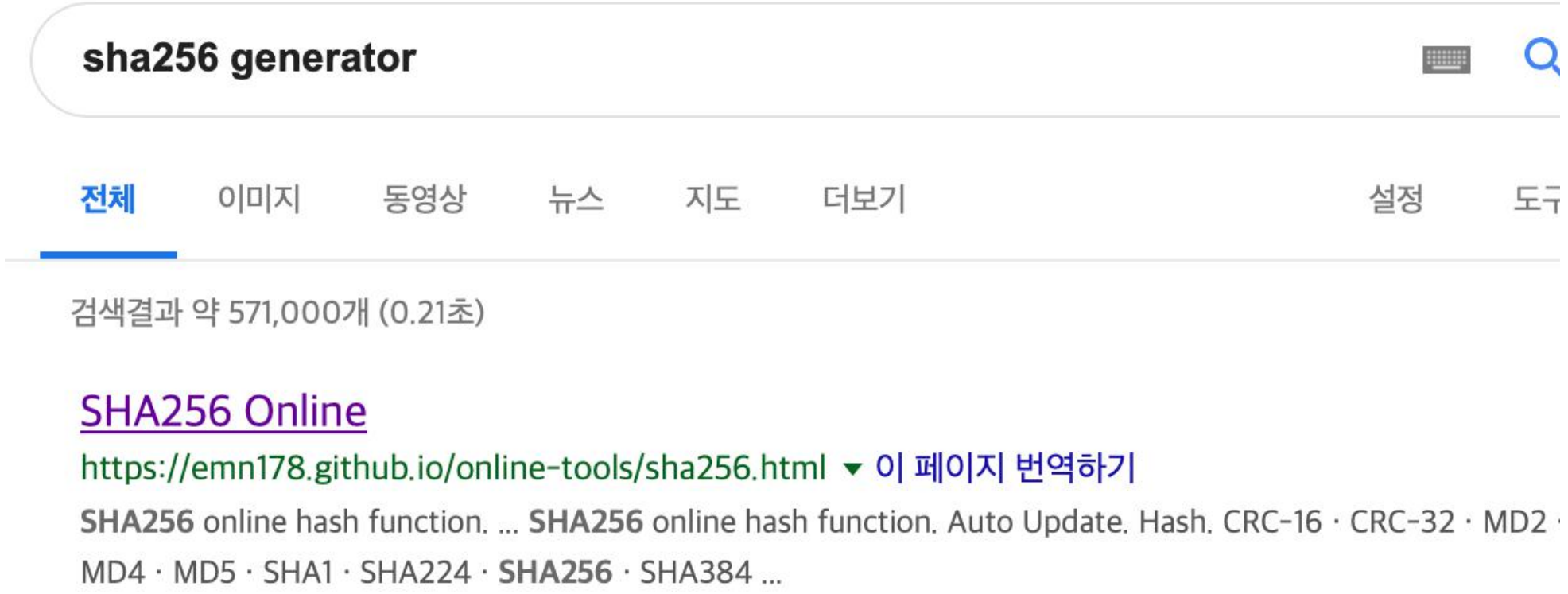
Hiding

결과값을 가지고 입력값을 찾기가 **실질적으로 불가능**

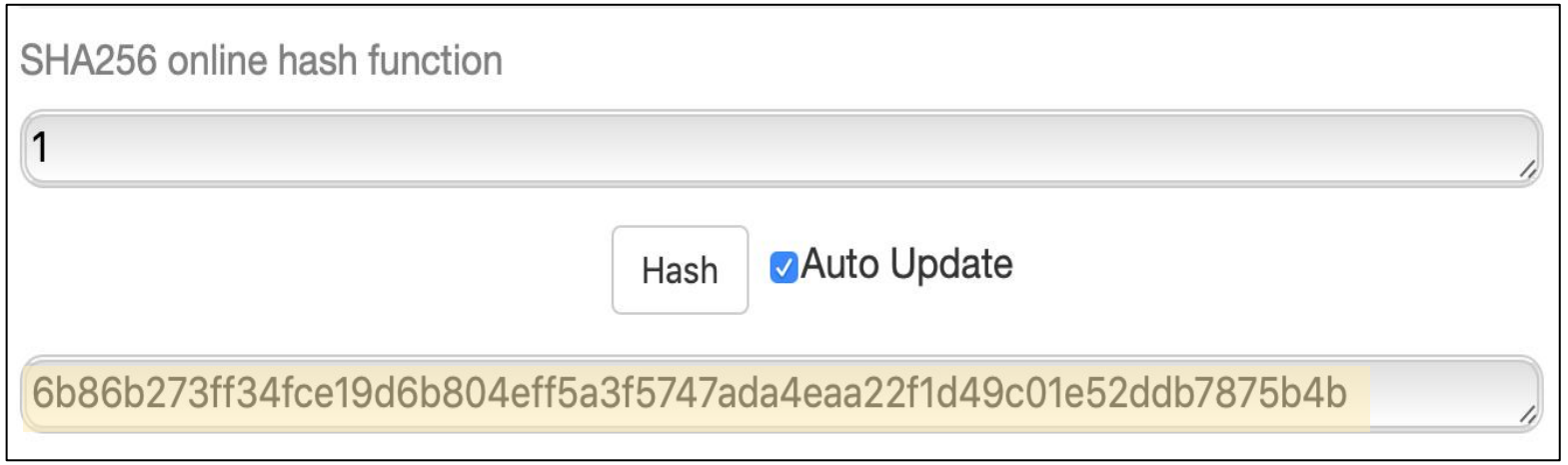
Puzzle friendliness

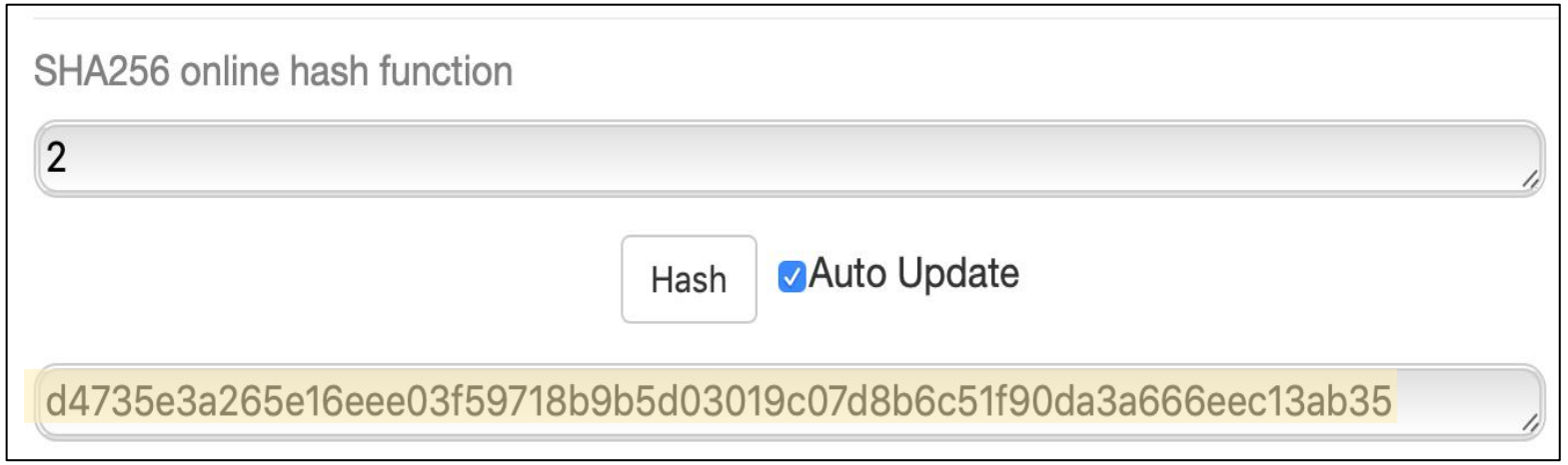
무작위로 찾는게 **현재로선 최선**

Secure Hash Algorithm with 256 bits (ref #8)



Secure Hash Algorithm with 256 bits (ref #8)





Applications of SHA: Digital fingerprint, Message digest (ref #8)

내 전 재산 중 **5%**를 ㅇㅇㅇ에 기부한다.

→

6BC1820A9BC0EB207F86741A22DCE5460
C959655A73FB3BDF2581B62FEF631CF

내 전 재산 중 **95%**를 ㅇㅇㅇ에 기부한다.

→

7A32C64C3AE19CEC3809E6DAC5B10A59F
58AEFF127907446F59A8375830D573A

고대 근처 맛집 리스트
대성집: 해장국 핵존맛, 수육 캬
고른햇살: 김혜자 쓰앵님 이상의 가성비 분식
형제집: 아재감성 술집쓰. 오돌뼈, 닭도리탕...
용초수: 꿔바로우 하... 토마토볶음 오...
유자유: 김치떡볶이 + 비빔밥
오샬: 인도커리 냠냠
춘자: 핵 저렴한 술집
회기역 근처 이자카야 고우 꼭 가세요. JMT!
회기역 근처 이자카야 고우 두 번 가세요!!!
...

→

9DC44C08F26189C2DB2ECB5B0711348D
CDE76DAC6227F39ACB33CB0882BD3A6E

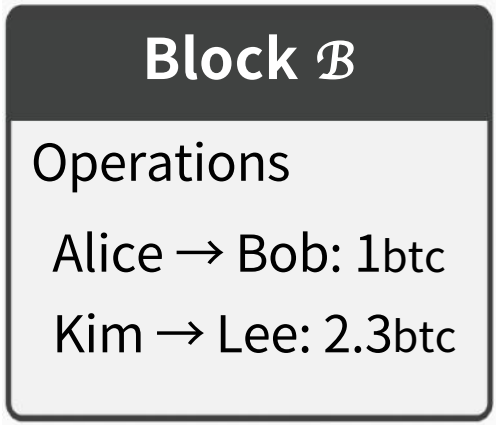
Hash puzzle in Bitcoin

블록의 해시 값이 **특정 조건**을 만족해야 적법한(**valid**) 블록으로 인정

특정 조건을 만족하도록 블록을 만들자!

Hash puzzle in Bitcoin

블록의 해시 값이 **특정 조건**을 만족해야 적법한(**valid**) 블록으로 인정



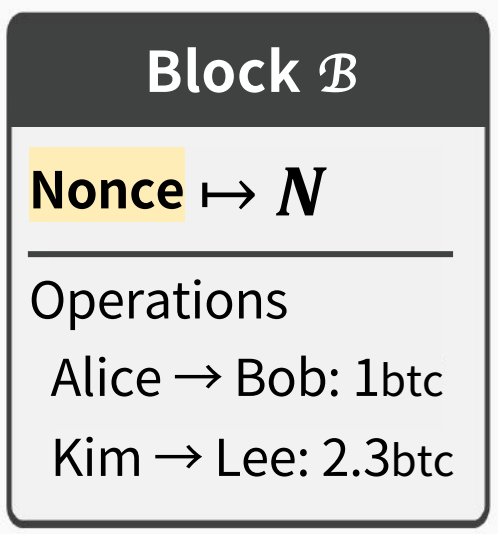
$\mathcal{Hash}(\mathcal{B})$ = 0347C308D64E2DF...D0C23B625031980089 1C3E6D2

$\mathcal{Hash}(\mathcal{B})$ = 0000 0011 0100 0111 ...

$\mathcal{Hash}(\mathcal{B})$가 홀수인가?
$\mathcal{Hash}(\mathcal{B})$를 7로 나눈 나머지가 2인가?
...

Hash puzzle in Bitcoin

블록의 해시 값이 **특정 조건**을 만족해야 적법한(**valid**) 블록으로 인정



$\mathcal{B}_i$ = Block $\mathcal{B}$ with i as a nonce:

$\mathcal{H}ash(\mathcal{B}_0)$ = 0110 1011 0110 1101 ...

$\mathcal{H}ash(\mathcal{B}_1)$ = 0011 0011 0100 0111 ...

...

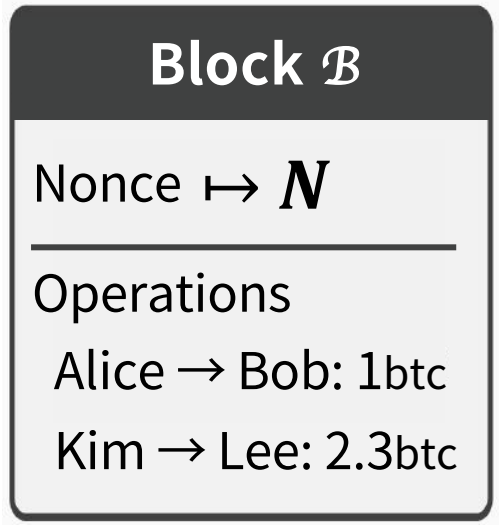
$\mathcal{H}ash(\mathcal{B}_{98})$ = 0000 0110 0110 0101 ...

...

$\mathcal{H}ash(\mathcal{B}_i)$이 5개의 0으로 시작하는 i를 찾아라

Hash puzzle in Bitcoin

$\mathcal{H}ash(\mathcal{B}_i)$이 n개의 0으로 시작하는 i를 찾아라



$\mathcal{B}_i$ = Block $\mathcal{B}$ with i as a nonce

256 0s and 1s

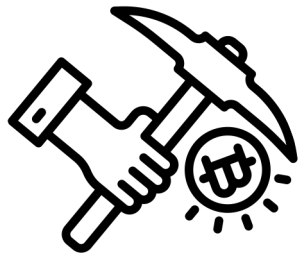
$\mathcal{H}ash(\mathcal{B}_i)$ = 0110 1011 0110 1101 ... 0110 1101 1111

n이 충분히 커지면, 확률적으로 i 역시 커진다

RULES

1. 비트코인을 송금하는 사람은 거래에 자신의 디지털 서명을 포함시켜야 한다.
2. 블록의 해시가 n개의 0으로 시작하도록 만드는 논스를 가장 먼저 찾은 사람이 기록이 공식으로 인정되고, 그 대가로 보상을 받는다.
3. 각 블록은 오퍼레이션과 논스로 구성된다.

Mining competition & Coinbase Tx



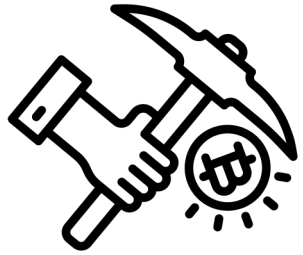
Miner A

Block $\mathcal{B}$

Nonce
2345678999
Operations
Miner A gets 12.5btc
Alice → Bob: 1btc
Kim → Lee: 2.3btc

SHA256

0010101001111100010011101101010010100100011
01001101010000101011000110101110110000111 10
10111000011100111000001100010000011100111 01
11001011011011101000010001011011000111111 10
10101100110001101111001001001100111110101 00
00101011110100000001001111111001011101101



Miner B

Block $\mathcal{B}$

Nonce
6876678999
Operations
Miner A gets 12.5btc
Alice → Bob: 1btc
Kim → Lee: 2.3btc

SHA256

0000000011100101001111100010011101101010 01
01001000110100110101000010101100011010111 01
10000000011100111000001100010000011100111 01
11001011011011101000010001011011000111111 10
10101100110001101111001001001100111110101 00
00101011110100000001001111111001011101101

Mining difficulty

n개의 bit로 나타낼 수 있는 경우의 수

$$\overbrace{\boxed{\mathbf{11111\ldots11111}}}^{n} = 2^n$$

Mining difficulty

n개의 bit 중 앞 i개가 0일 때,
나타낼 수 있는 경우의 수

$$\underbrace{0000}_{i}\underbrace{1\ldots11111}_{n-i} \quad (n) = 2^{n-i}$$

Mining difficulty

256개의 bit 중 앞 96개가 0일 때

Output space of SHA256

000000000011111111...11111111

target space

2^{96}

probability

2^{160}

2의 160승도 여전히 매우 큰 수

2^{160} 비트코인 지갑의 개수 (RIPE-MD160)

= 1,460,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000

2,045년, 지구 인구 **90억**명

모든 사람이 비트코인 주소 **천만개** 씩 소유

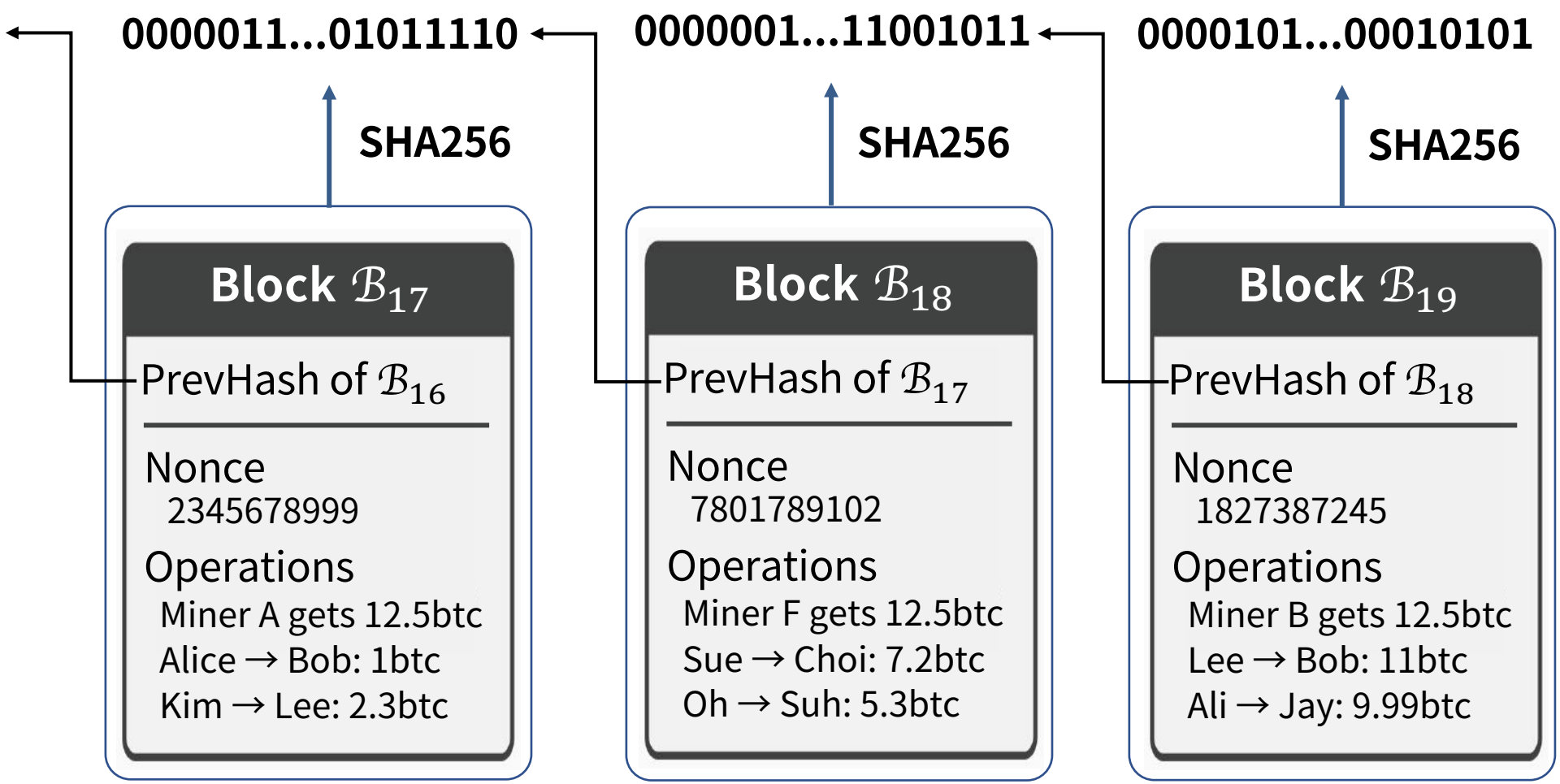
총 **90,000,000,000,000,000개**의 지갑

90,000,000,000,000,000 / 2^{160}

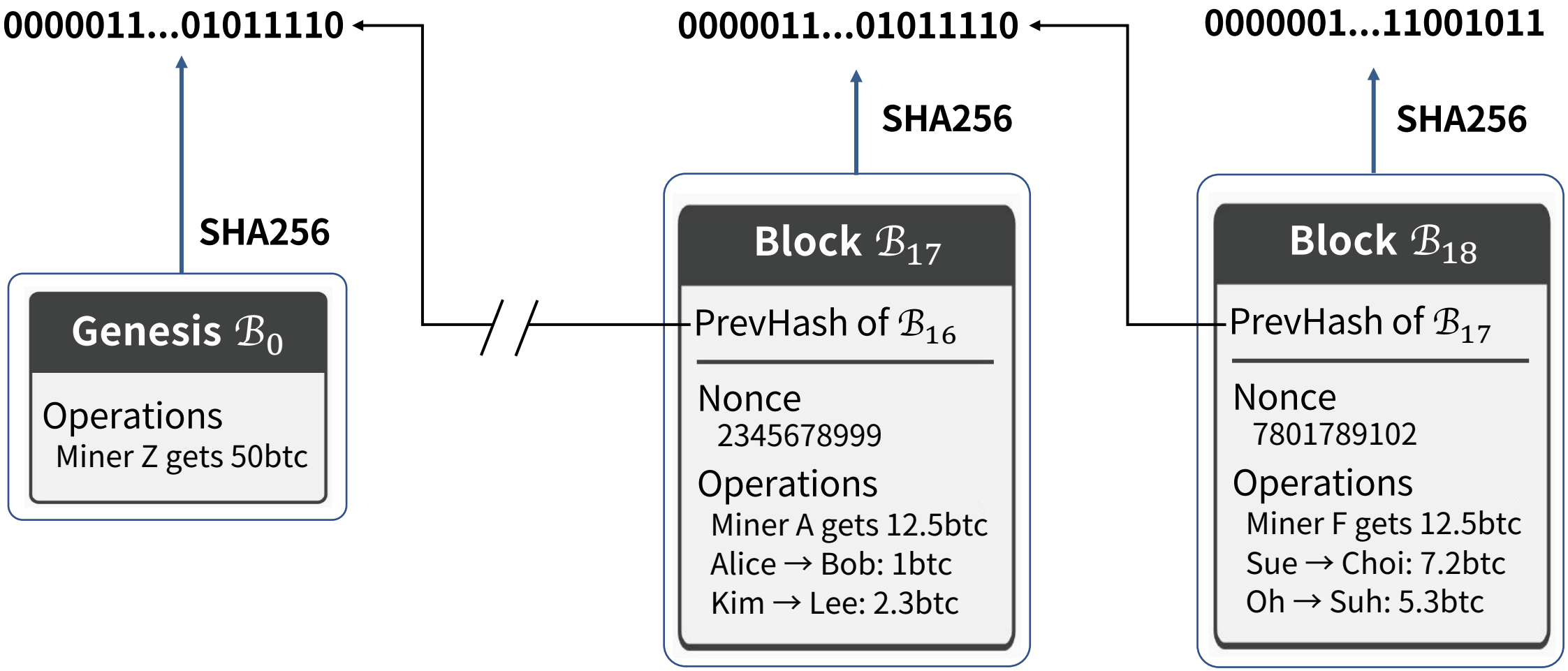
= **0.000000000000000000000000000000615%**

= 전기세, 실제 비트코인이 들어있는지, 얼마나 들어있는지 등은 고려 안 됨

Blockchain = A chain of blocks



Blockchain = A chain of blocks



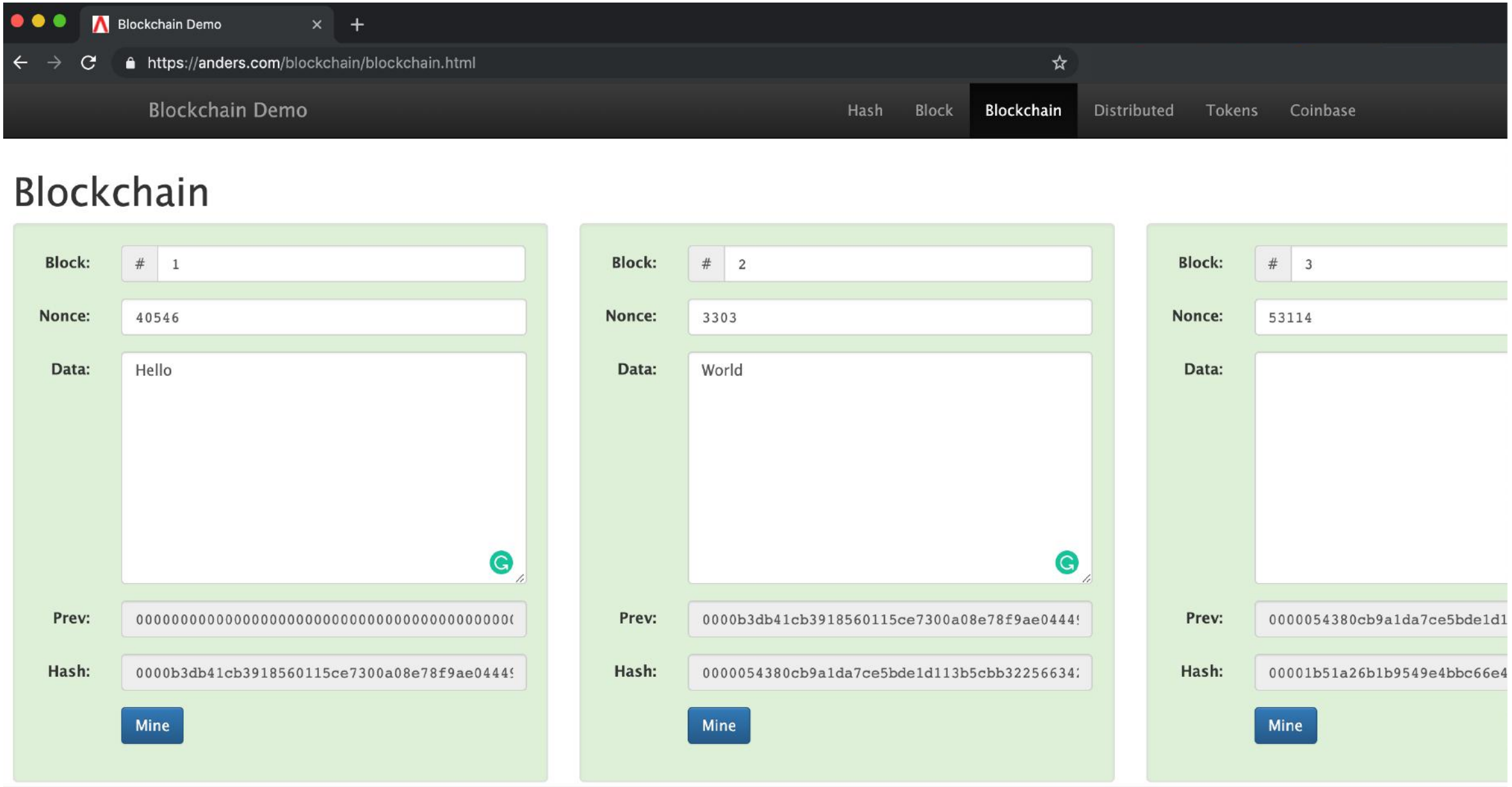
bitcoin/src/chainparams.cpp

```
42   * Build the genesis block. Note that the output of its generation
43   * transaction cannot be spent since it did not originally exist in the
44   * database.
45   *
46   * CBlock(hash=000000000019d6, ver=1, hashPrevBlock=00000000000000, hashMerkleRoot=4a5e1e, nTime=1231006505, nBits=1d00ffff, nNonc
47   *   CTransaction(hash=4a5e1e, ver=1, vin.size=1, vout.size=1, nLockTime=0)
48   *     CTxIn(COutPoint(000000, -1), coinbase 04ffff001d0104455468652054696d65732030332f4a616e2f3230303920436861696e63656c6c6f72206f6
49   *     CTxOut(nValue=50.00000000, scriptPubKey=0x5F1DF16B2B704C8A578D0B)
50   *   vMerkleTree: 4a5e1e
51   */
52  static CBlock CreateGenesisBlock(uint32_t nTime, uint32_t nNonce, uint32_t nBits, int32_t nVersion, const CAmount& genesisReward)
53  {
54      const char* pszTimestamp = "The Times 03/Jan/2009 Chancellor on brink of second bailout for banks";
55      const CScript genesisOutputScript = CScript() << ParseHex("04678afdb0fe5548271967f1a67130b7105cd6a828e03909a67962e0ea1f61deb64
56      return CreateGenesisBlock(pszTimestamp, genesisOutputScript, nTime, nNonce, nBits, nVersion, genesisReward);
57  }
```

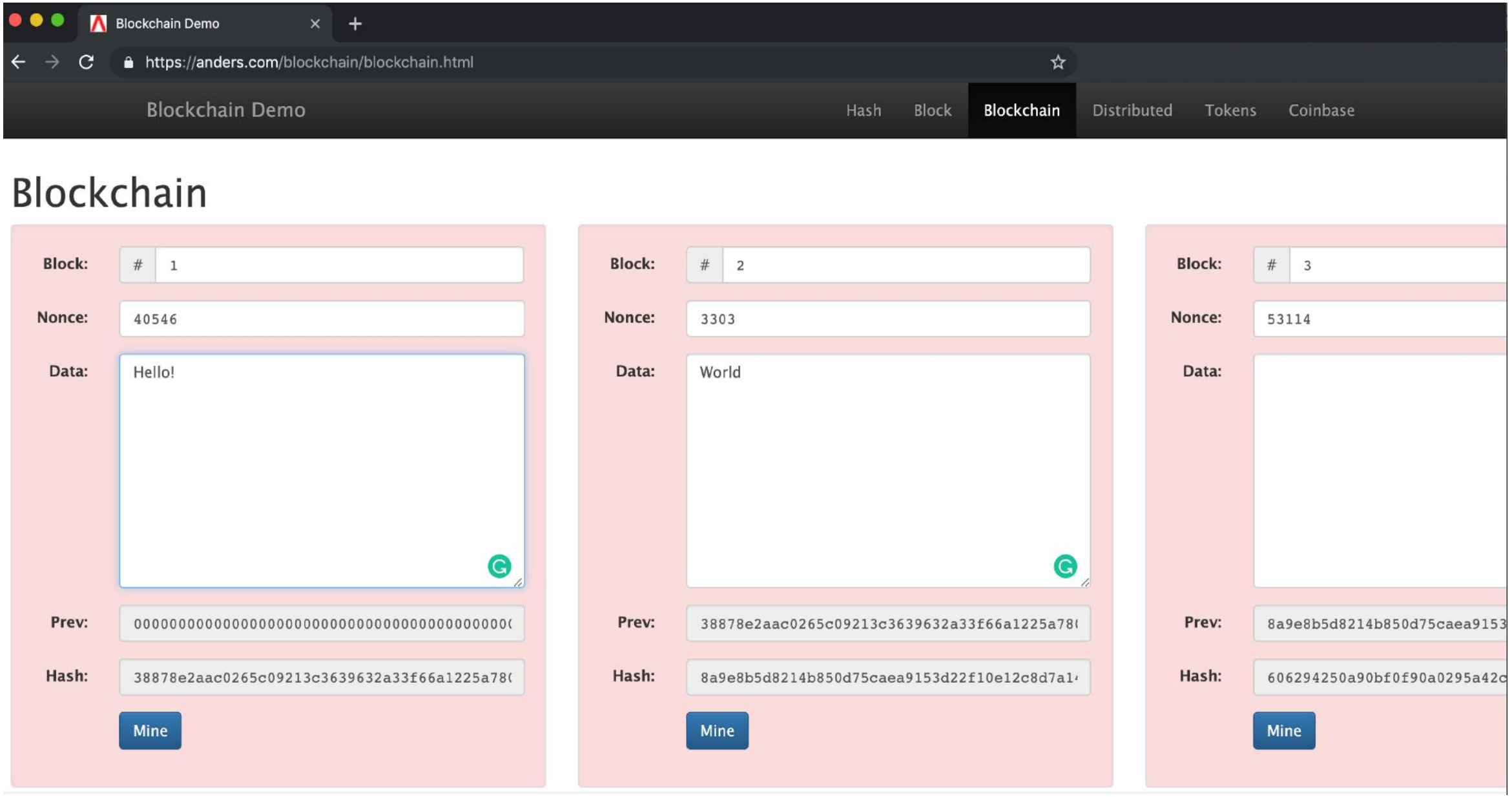
bitcoin/src/chainparams.cpp

```
 *
 * CBlock(hash=000000000019d6, ver=1, hashPrevBlock=00000000000000, hashMerkleRoot=4a5e1e, nTime=1231006505, nBits
 *   CTransaction(hash=4a5e1e, ver=1, vin.size=1, vout.size=1, nLockTime=0)
 *     CTxIn(COutPoint(000000, -1), coinbase 04ffff001d0104455468652054696d65732030332f4a616e2f3230303920436861696e6
 *     CTxOut(nValue=50.00000000, scriptPubKey=0x5F1DF16B2B704C8A578D0B)
 *   vMerkleTree: 4a5e1e
 */
static CBlock CreateGenesisBlock(
    uint32_t nTime,
    uint32_t nNonce,
    uint32_t nBits,
    int32_t nVersion,
    const CAmount& genesisReward) {
      const char* pszTimestamp = "The Times 03/Jan/2009 Chancellor on brink of second bailout for banks";
      const CScript genesisOutputScript = CScript() \
        << ParseHex(
            "04678afdb0fe5548271967f1a67130b7105cd6a828e03909a67962e0ea1f61deb649f6bc3f4cef38c4f35504e51ec112de5c384
        << OP_CHECKSIG;
      return CreateGenesisBlock(pszTimestamp, genesisOutputScript, nTime, nNonce, nBits, nVersion, genesisReward);
}
```

Blockchain Demo: Finding nonce (ref #9)



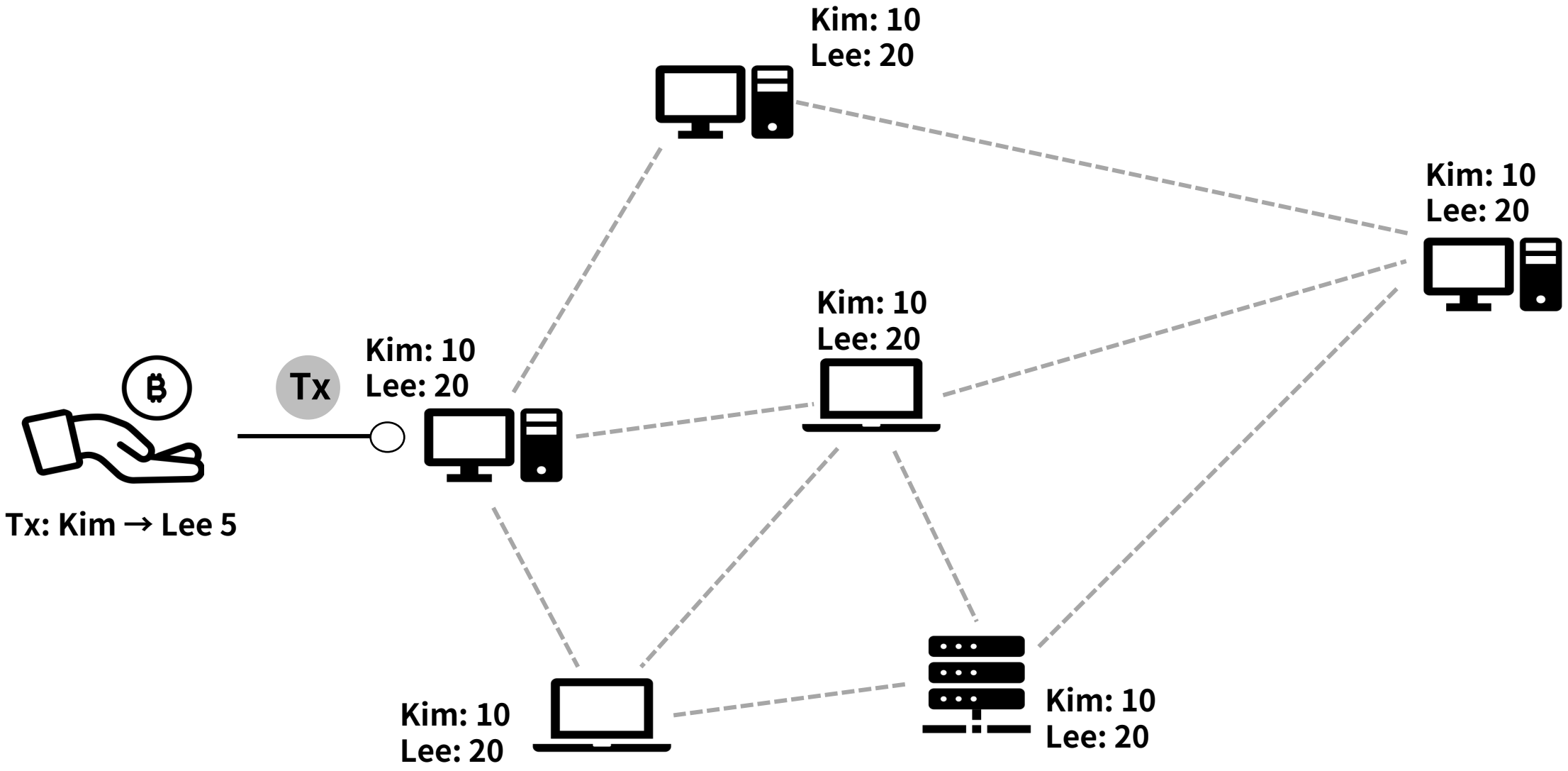
Blockchain Demo: Finding nonce (ref #9)



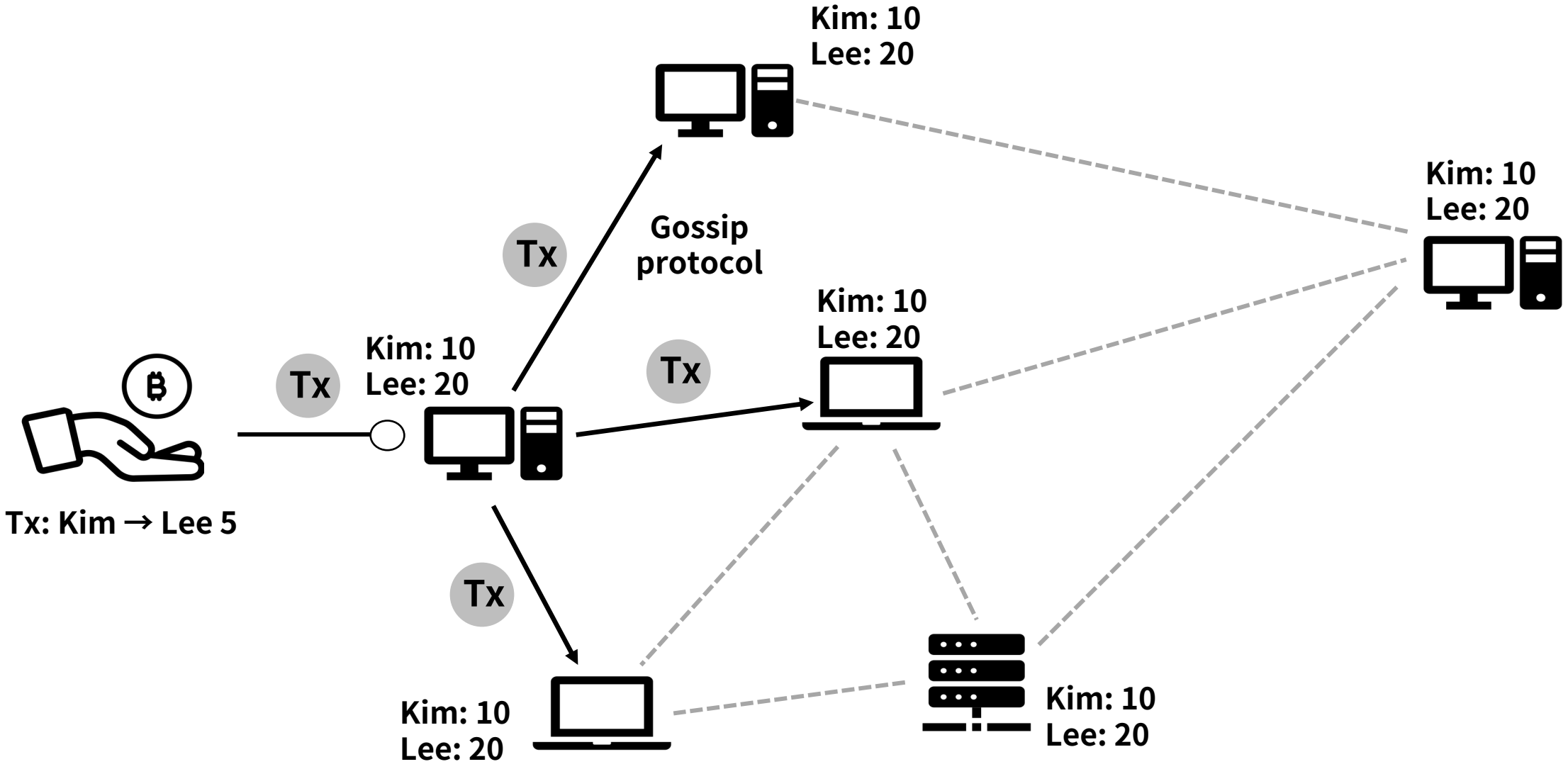
RULES

1. 비트코인을 송금하는 사람은 거래에 자신의 디지털 서명을 포함시켜야 한다.
2. 각 블록은 이전 블록의 해시, 오퍼레이션, 이전 블록의 해시, 논스로 구성된다.
3. 제일 먼저 블록의 해시를 난이도보다 낮게 만든 채굴자가 보상을 받는다.
4. 채굴자는 블록에 보상(코인베이스 트랜잭션)을 포함시킨다.
5. 새 블록의 해시는 $\mathcal{H}ash$(이전 블록의 해시 + 오퍼레이션 + 논스)이며, 논스를 바꿔가며 목표 해시를 찾는다.

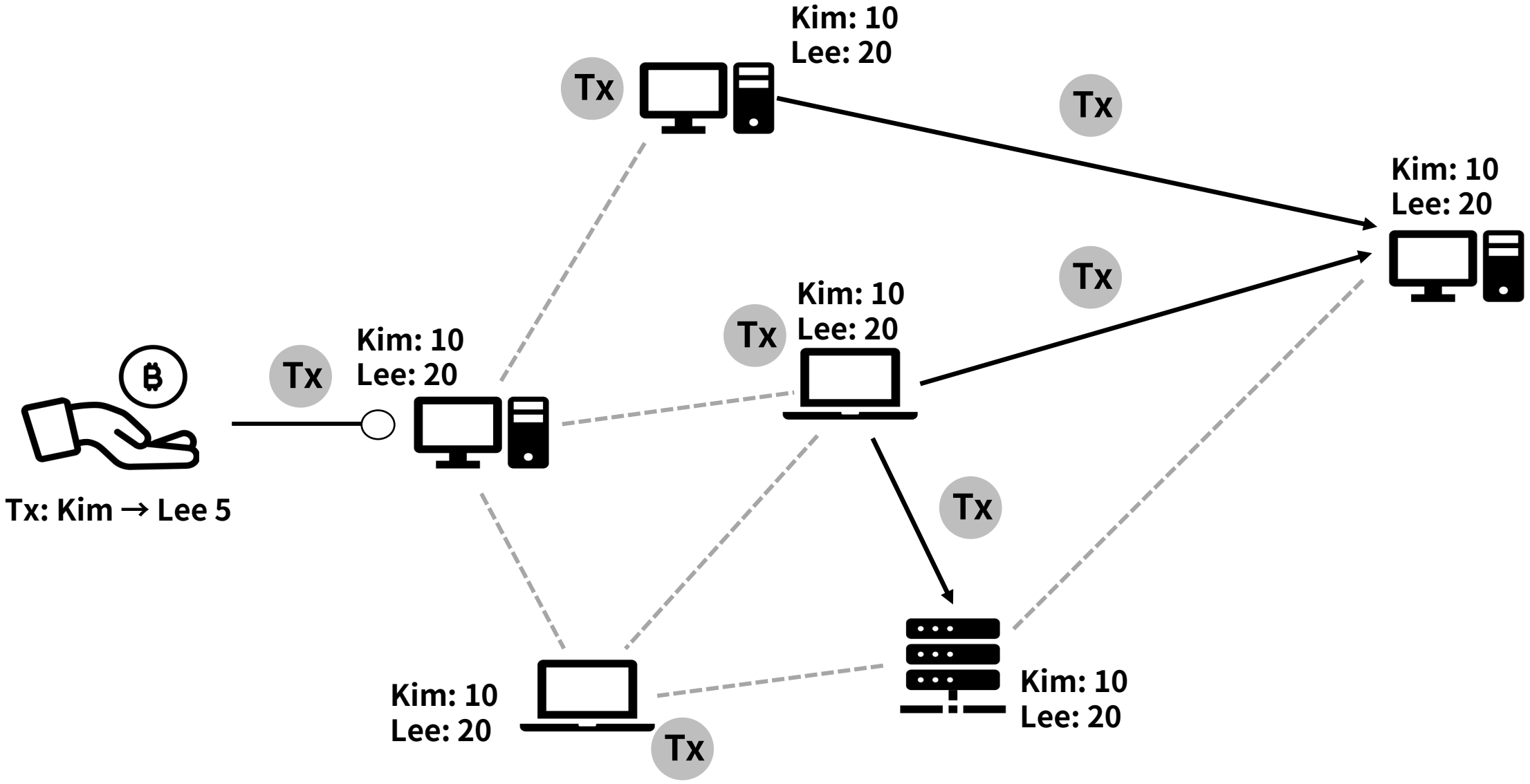
Life cycle of transaction



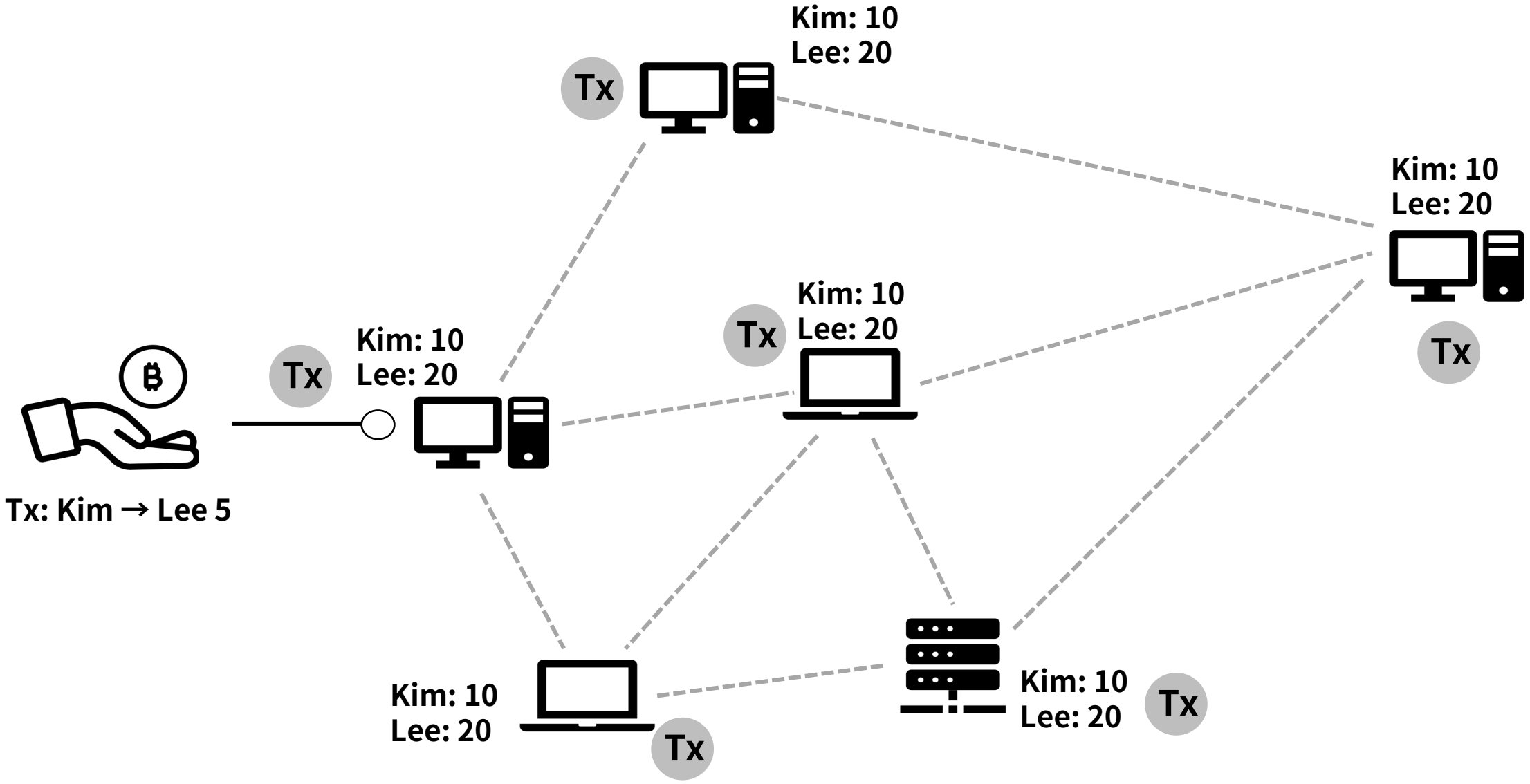
Life cycle of transaction



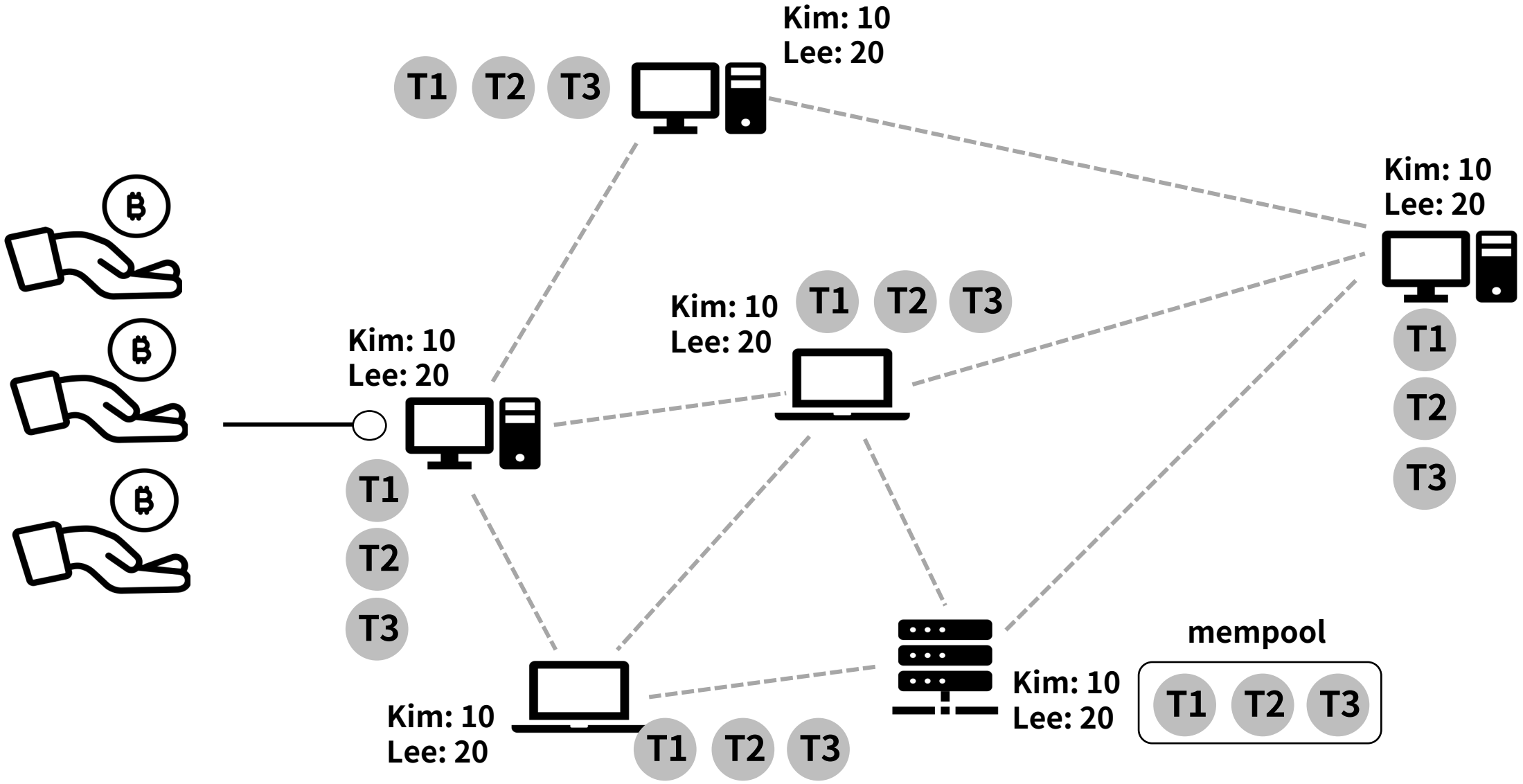
Life cycle of transaction



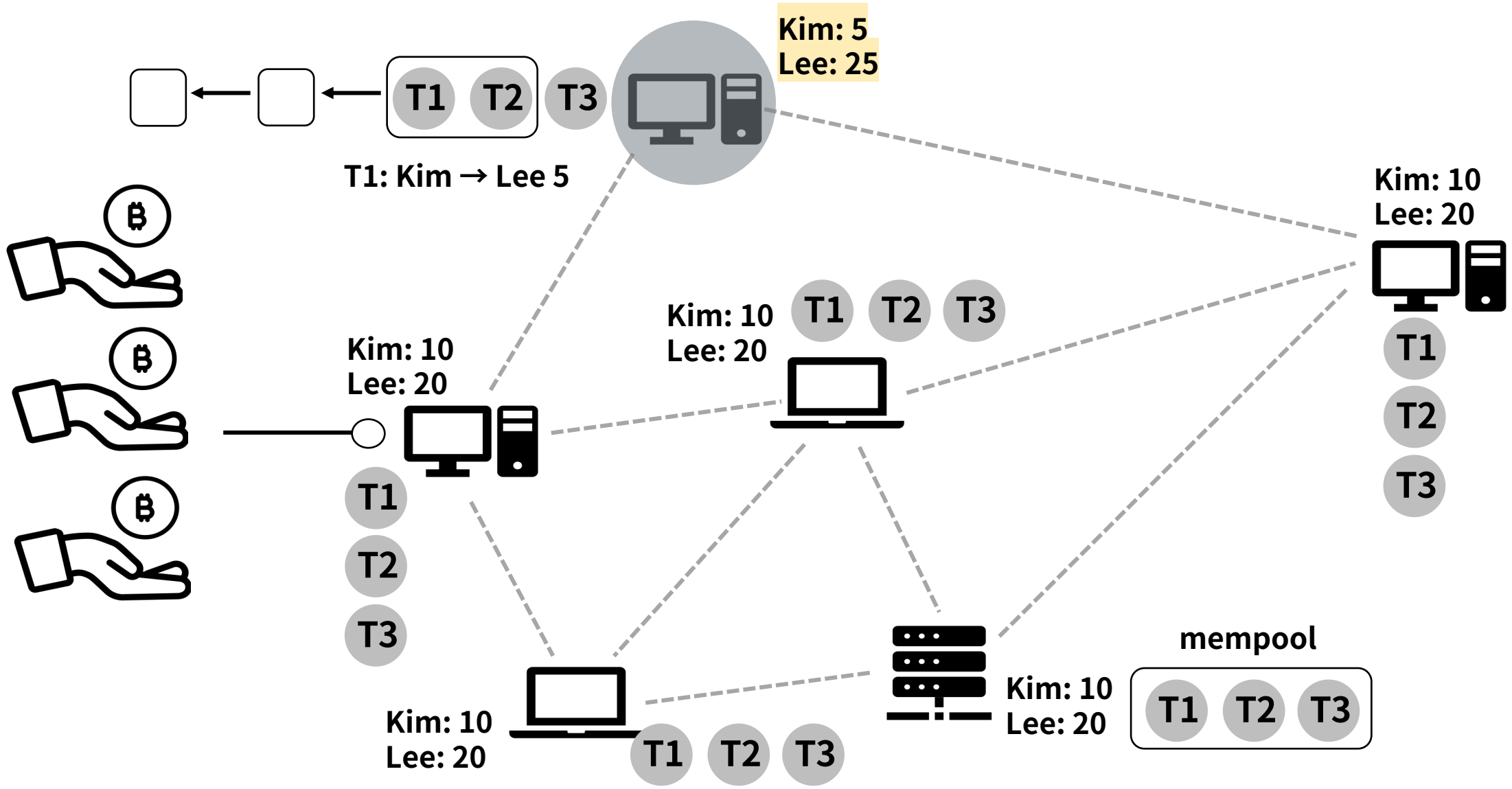
Life cycle of transaction



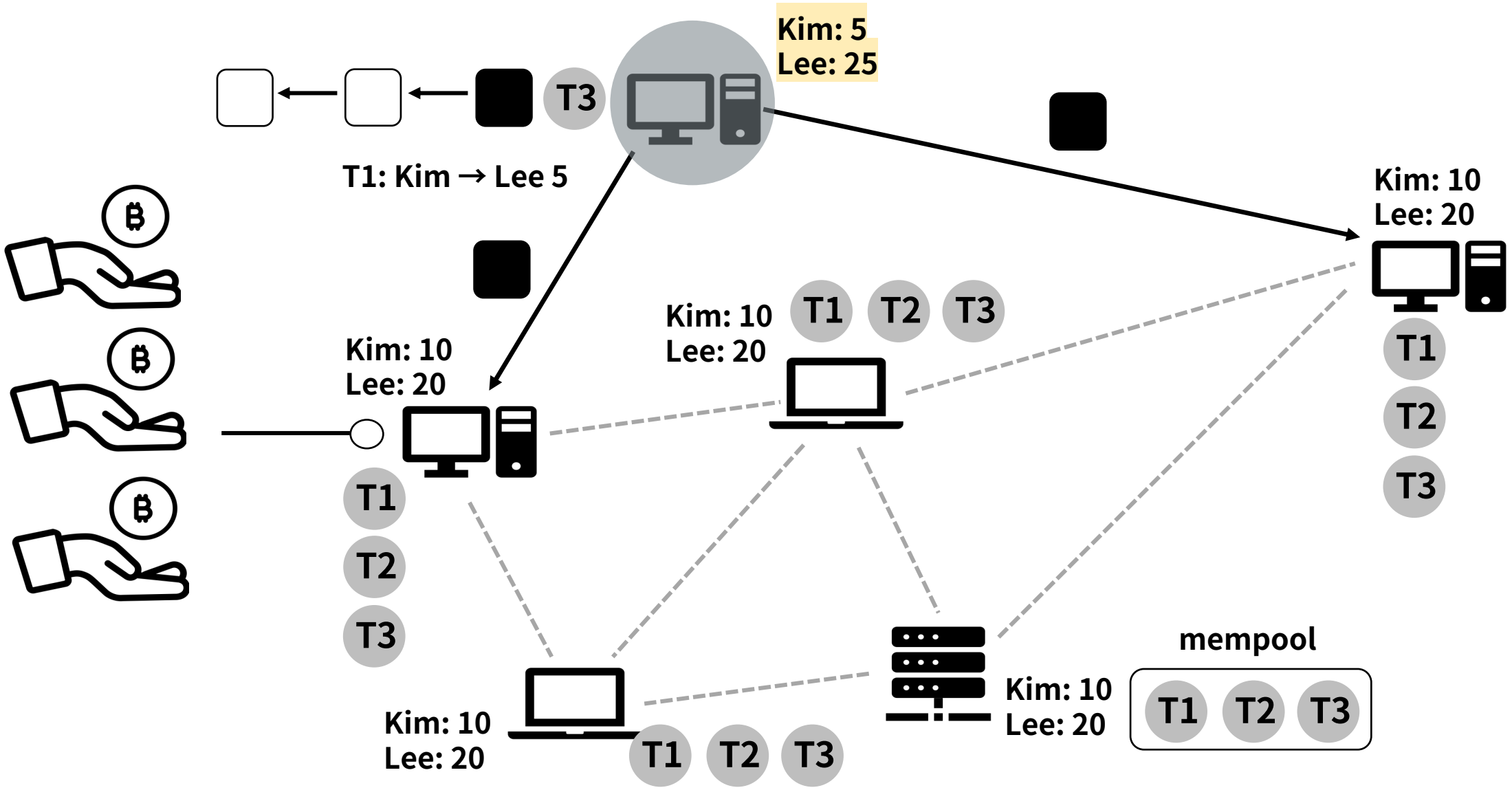
Life cycle of transaction



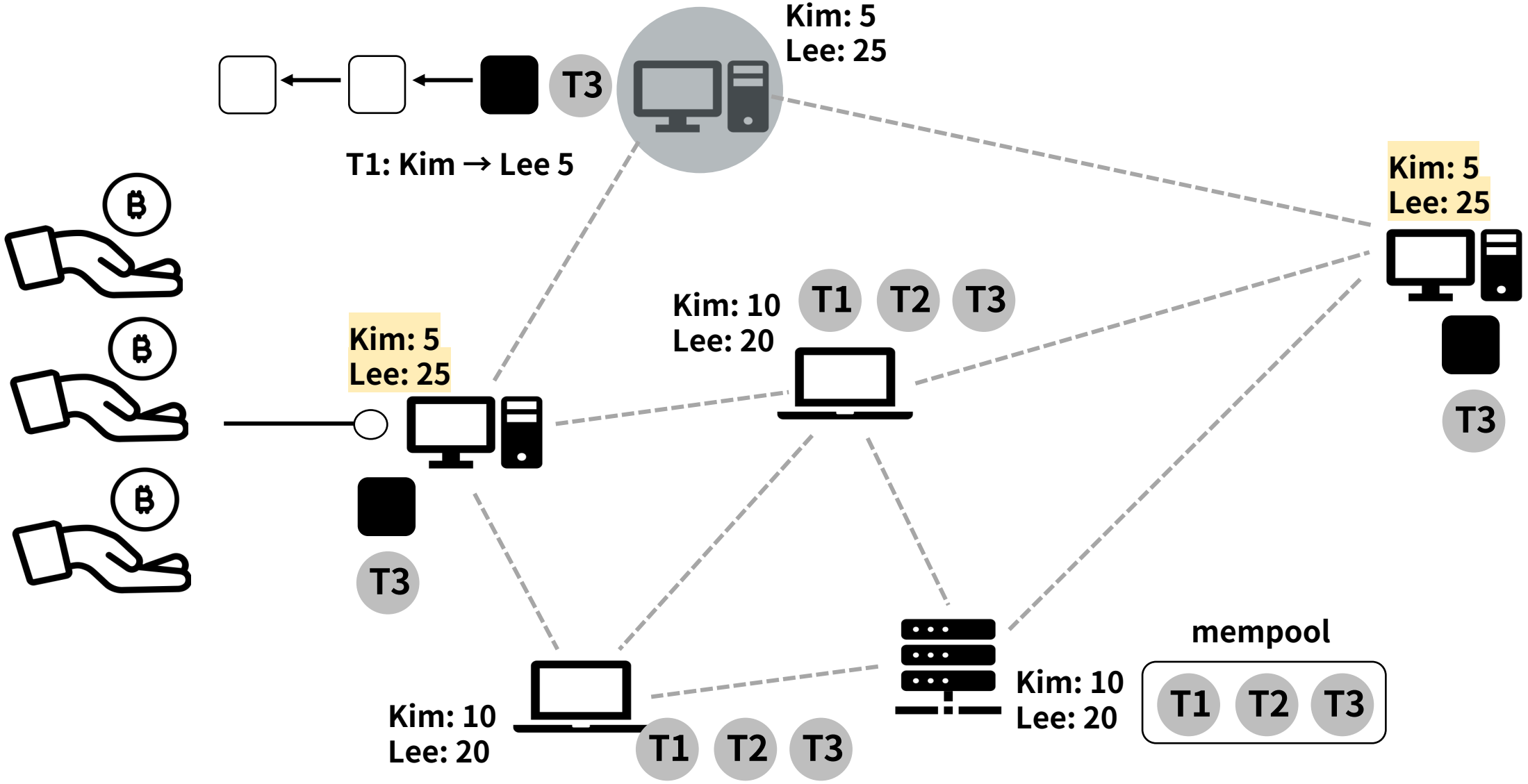
Life cycle of transaction



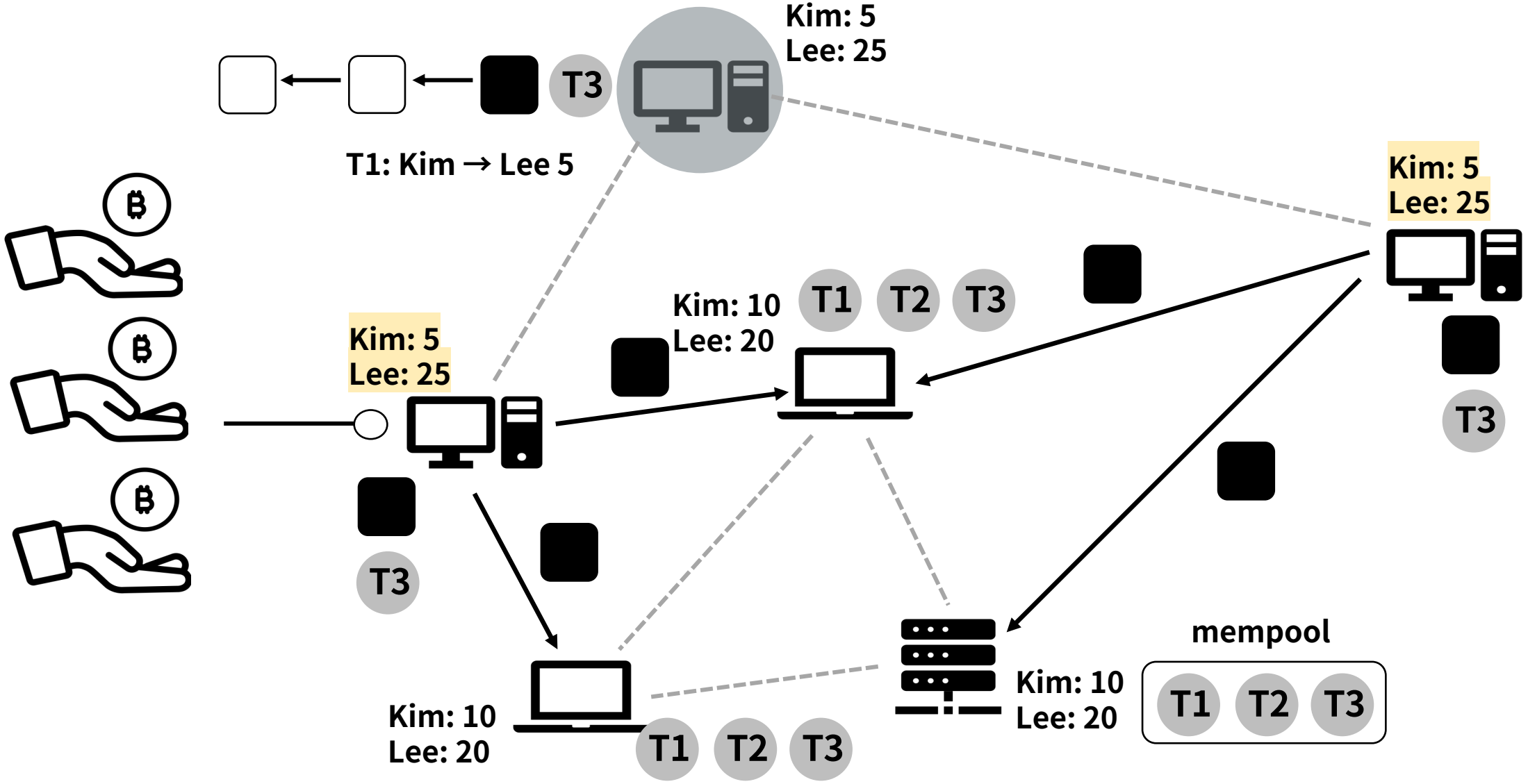
Life cycle of transaction



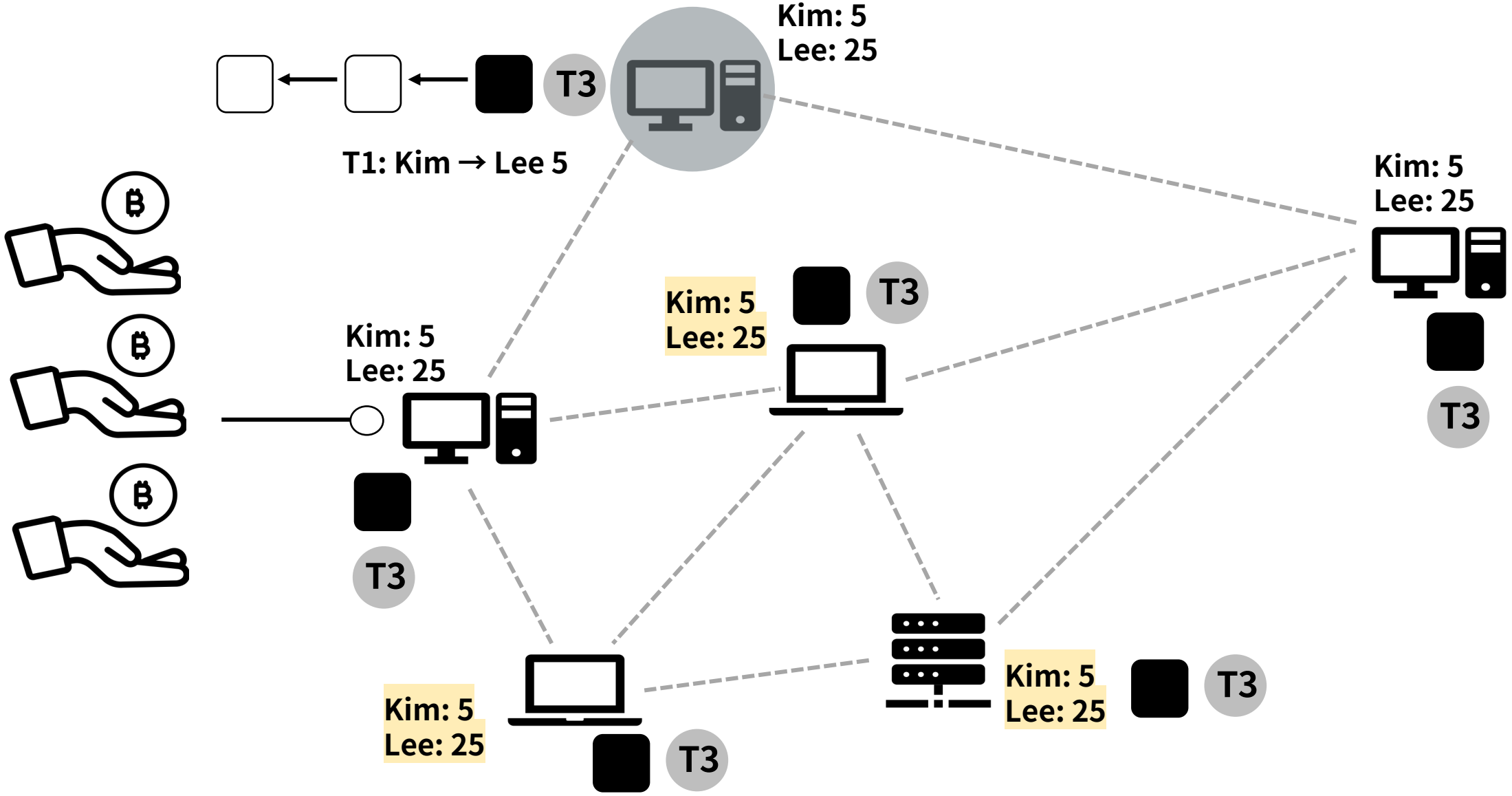
Life cycle of transaction



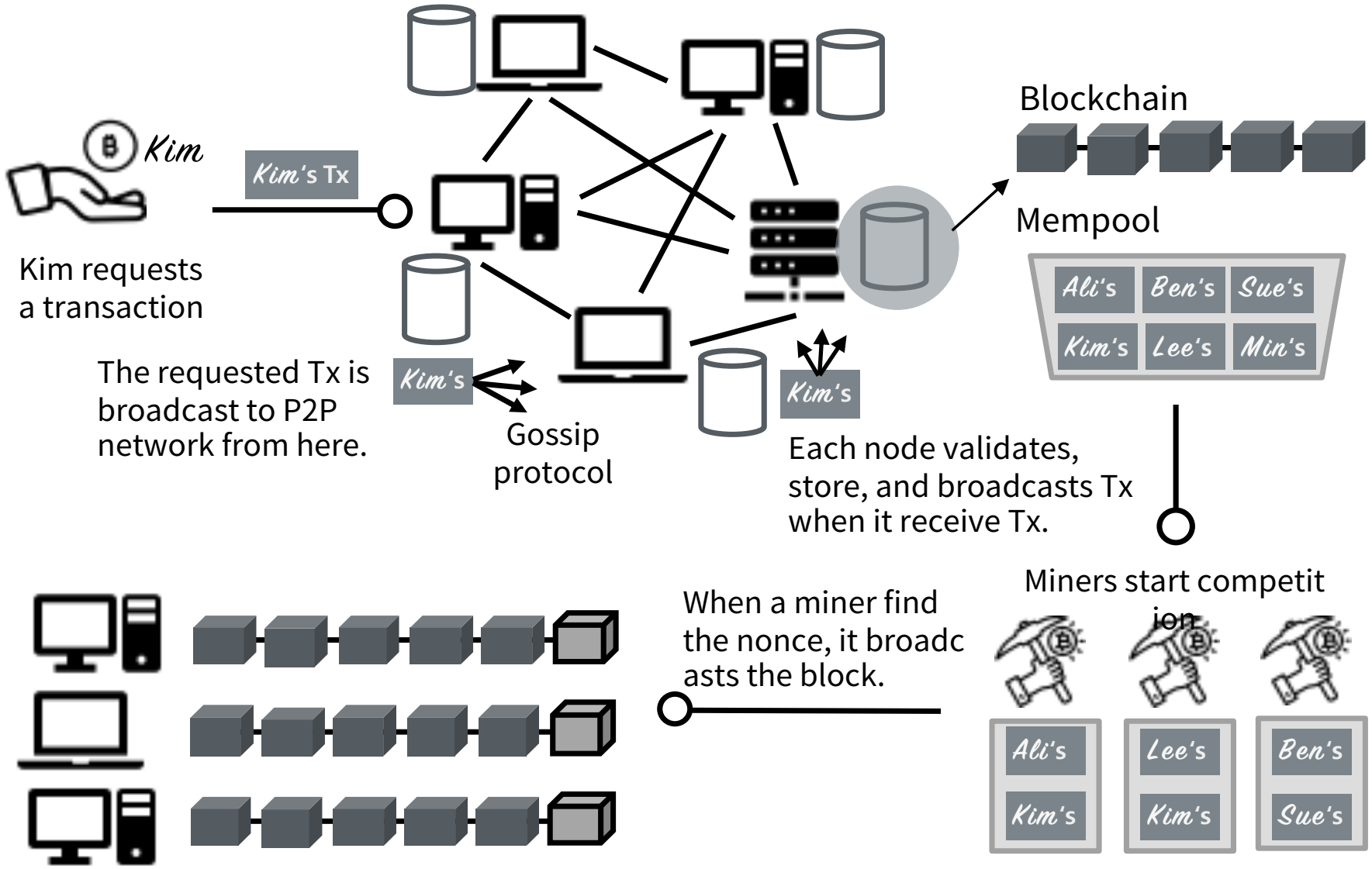
Life cycle of transaction



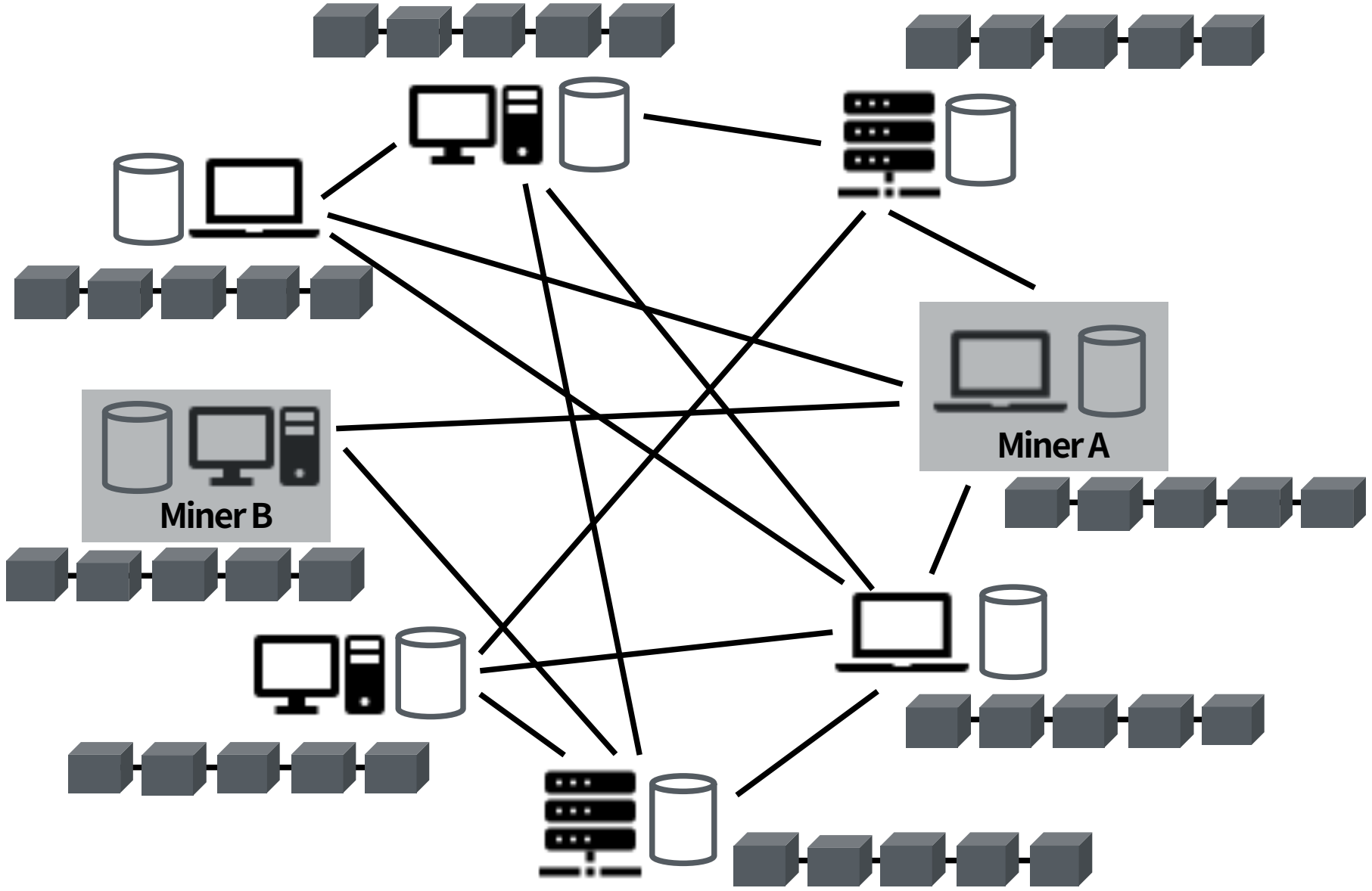
Life cycle of transaction



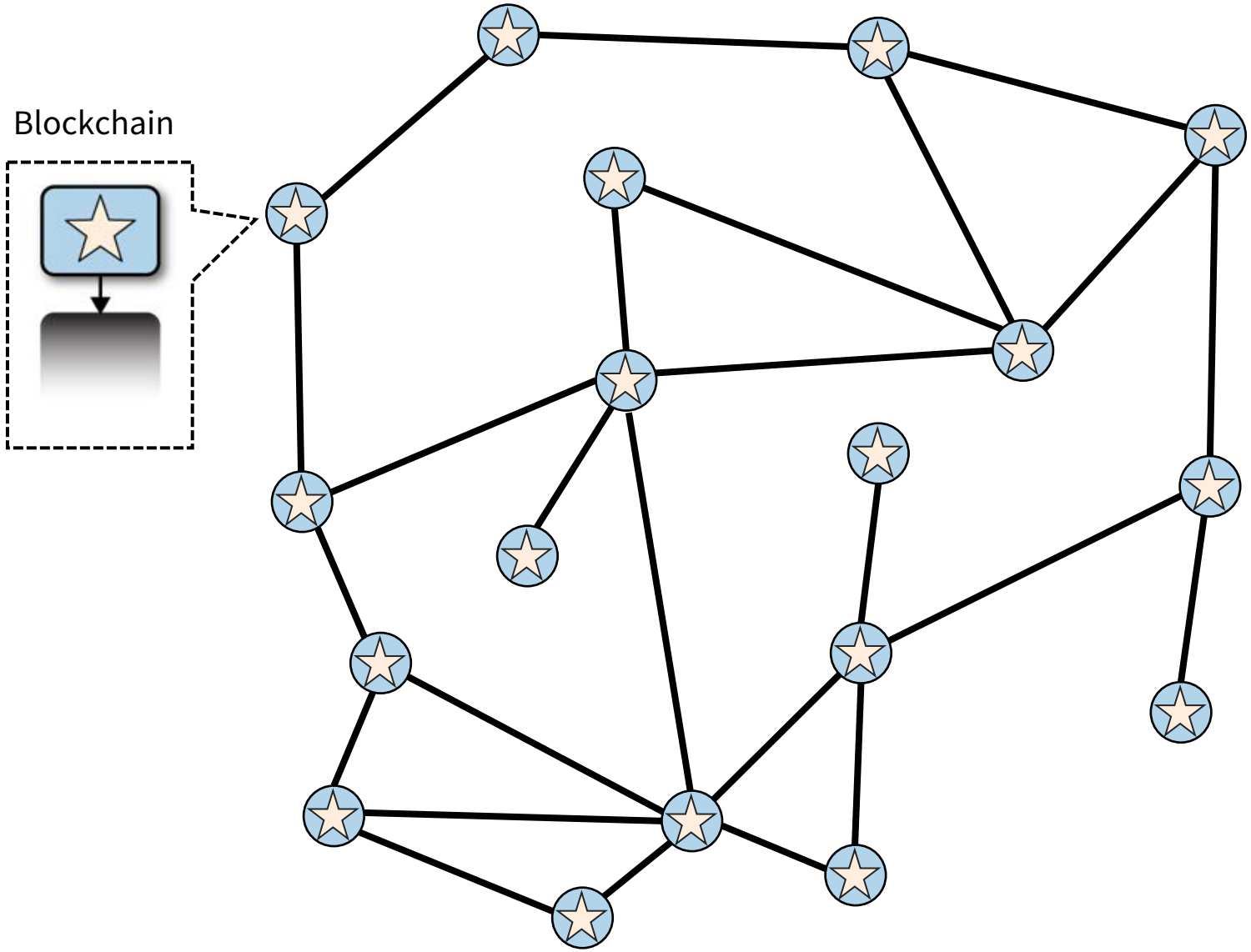
블록체인 충돌 해소 과정


Kim
Kim's Tx
Kim requests
a transaction
The requested Tx is
broadcast to P2P
network from here.
Kim's
Gossip
protocol
Kim's
Each node validates,
store, and broadcasts Tx
when it receive Tx.
Blockchain
Mempool
Ali's
Ben's
Sue's
Kim's
Lee's
Min's
Miners start competit
ion
When a miner find
the nonce, it broadc
asts the block.
Ali's
Kim's
Lee's
Kim's
Ben's
Sue's

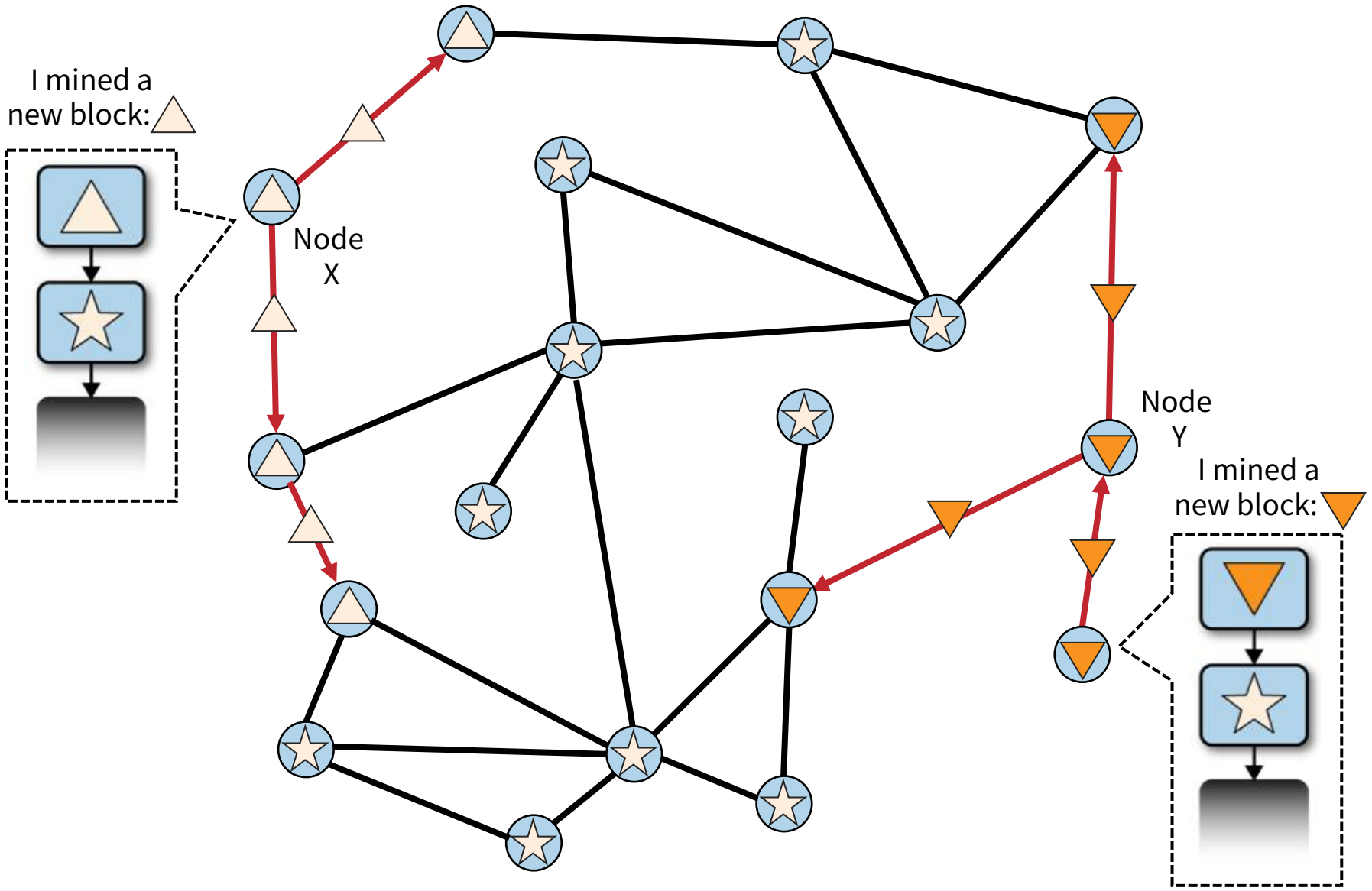
블록체인 충돌 해소 과정



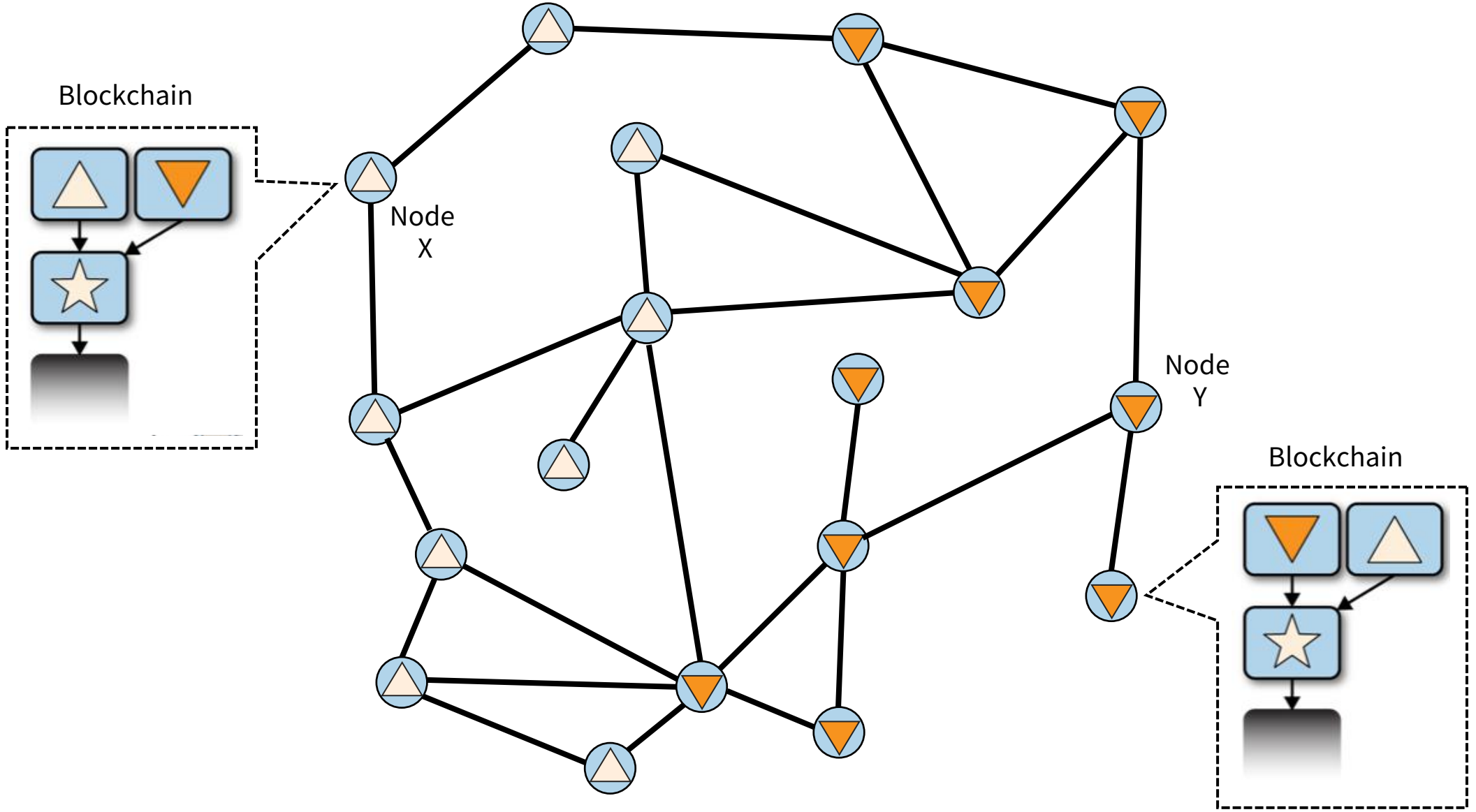
블록체인 충돌 해소 과정 (Mastering Bitcoin)



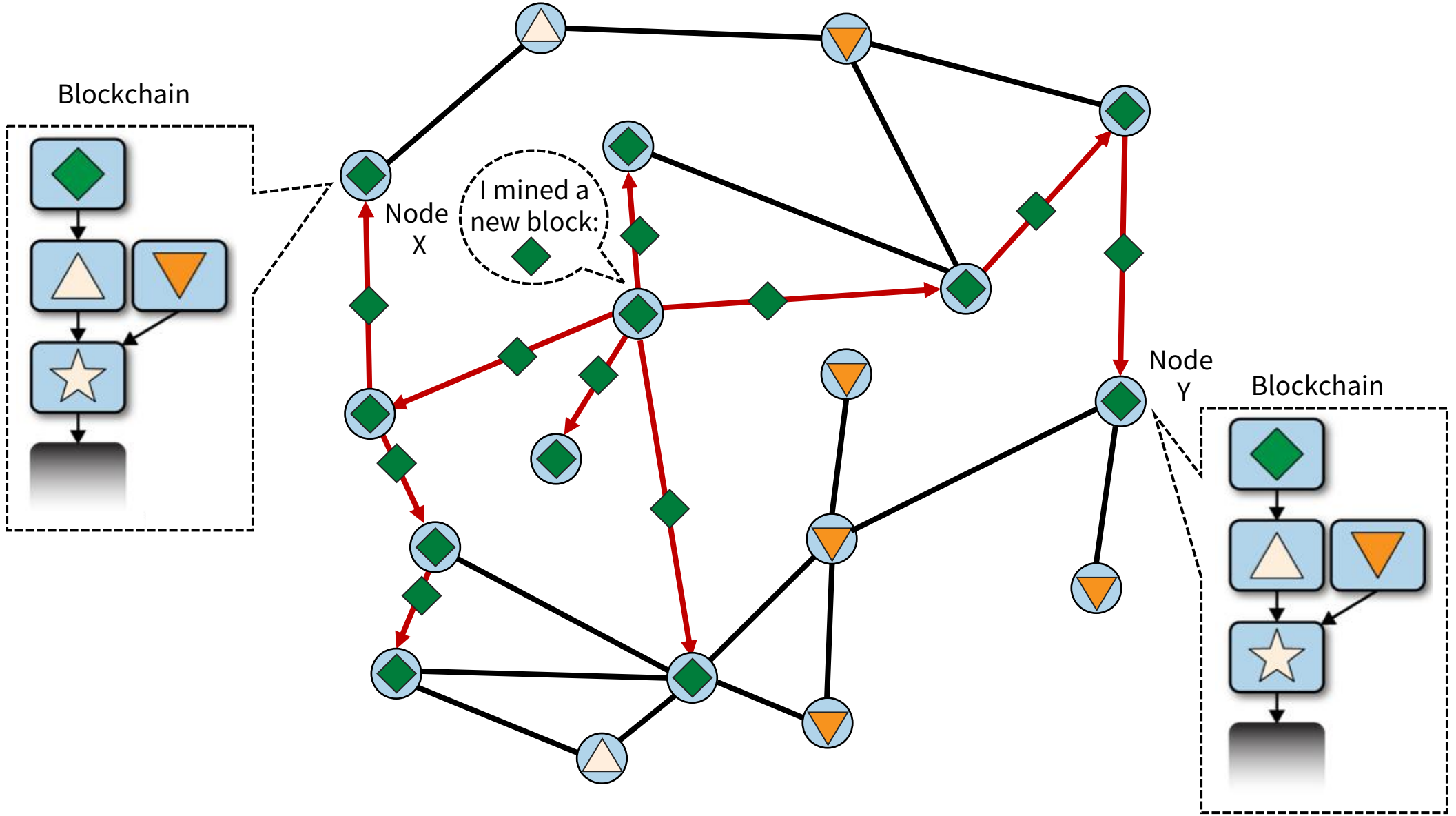
블록체인 충돌 해소 과정 (Mastering Bitcoin)



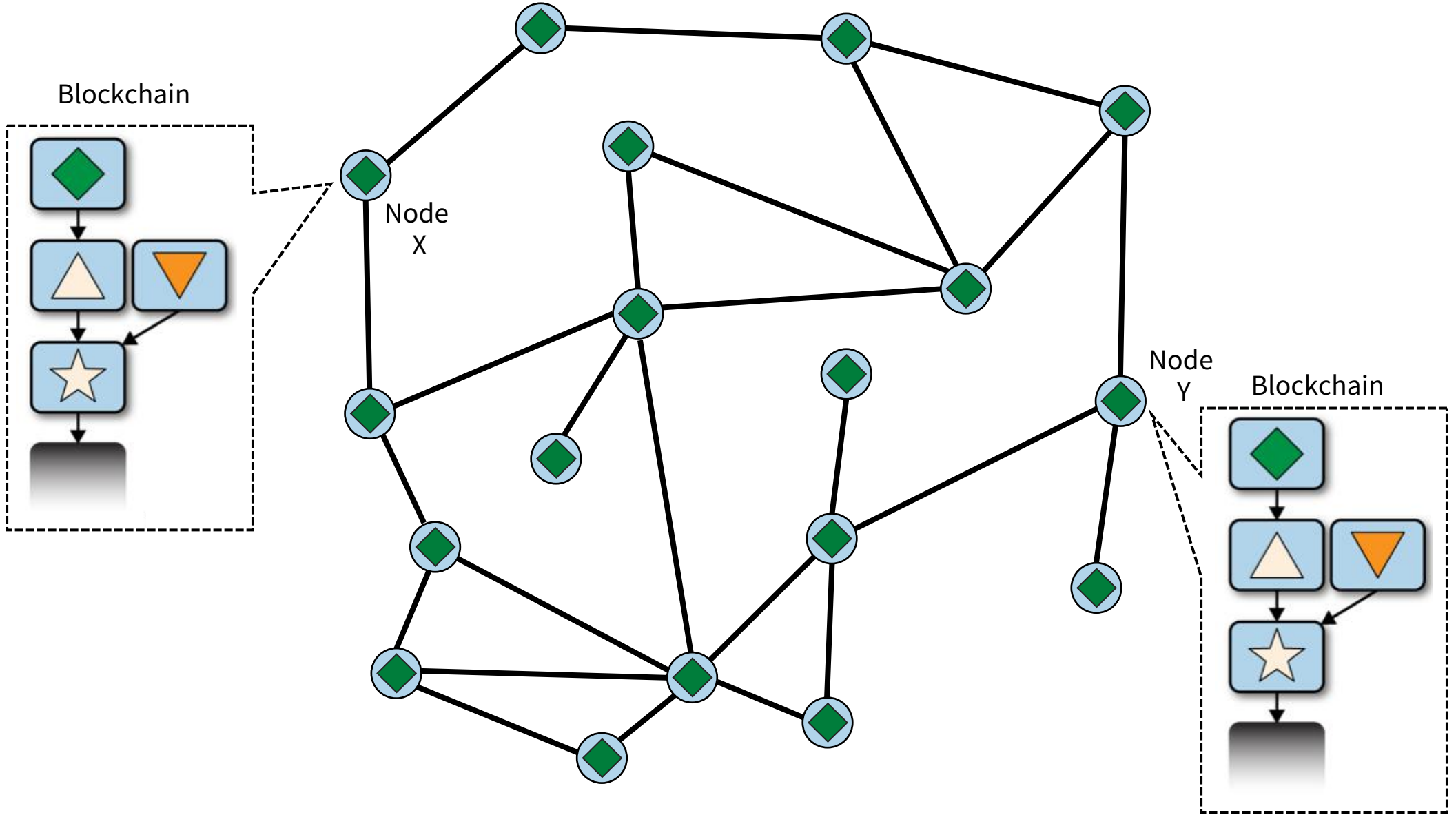
블록체인 충돌 해소 과정 (Mastering Bitcoin)



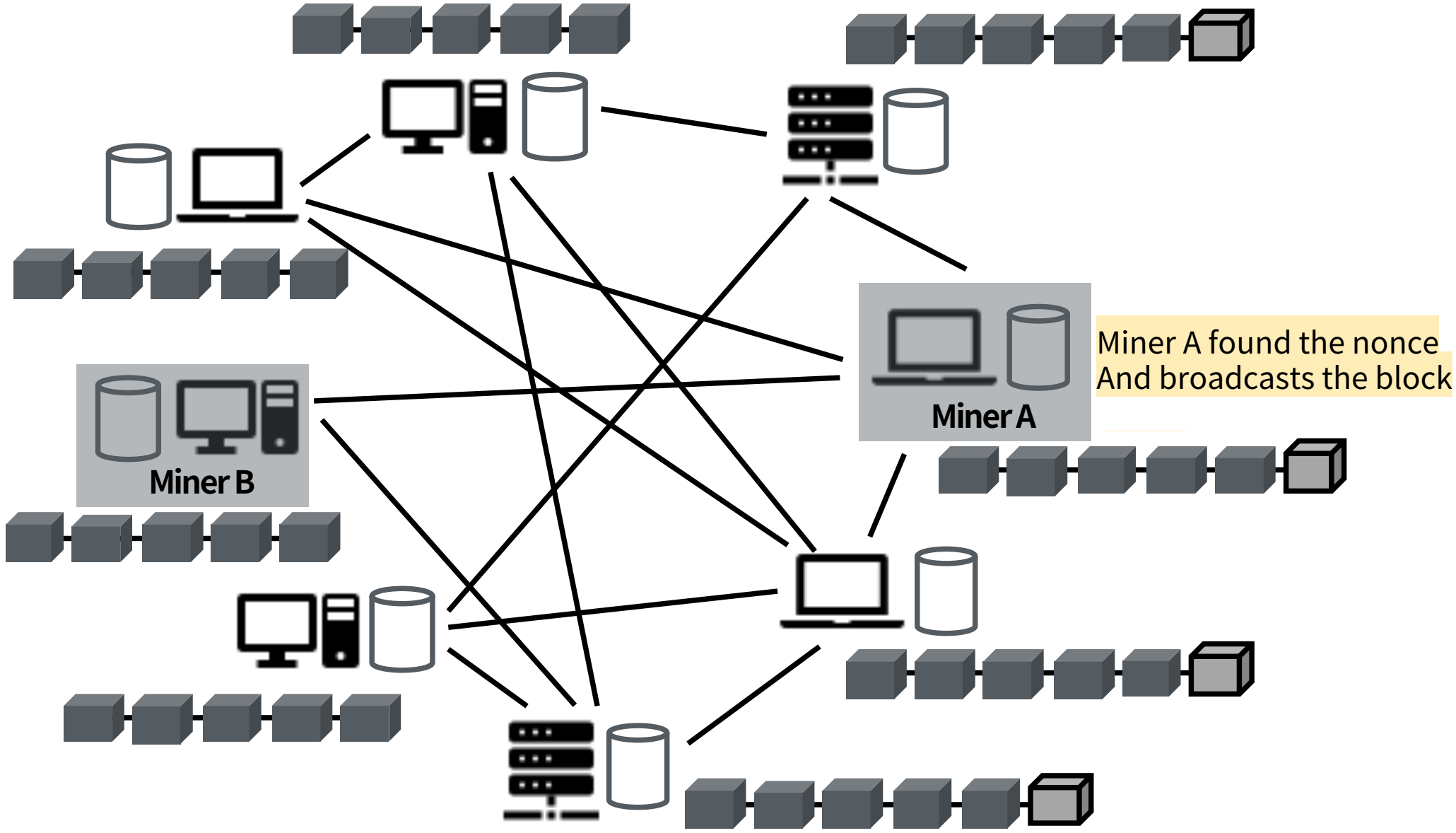
블록체인 충돌 해소 과정 (Mastering Bitcoin)



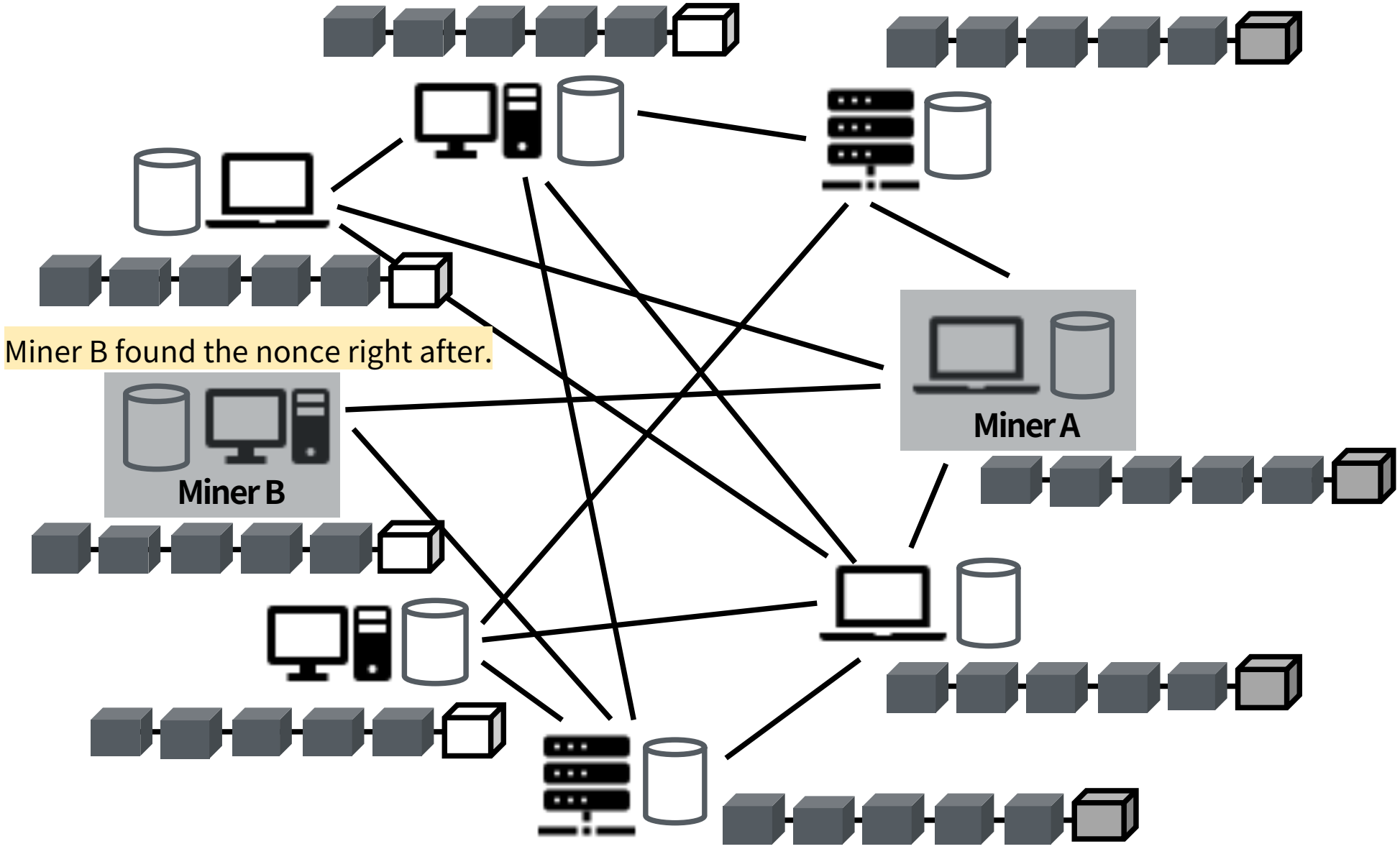
블록체인 충돌 해소 과정 (Mastering Bitcoin)



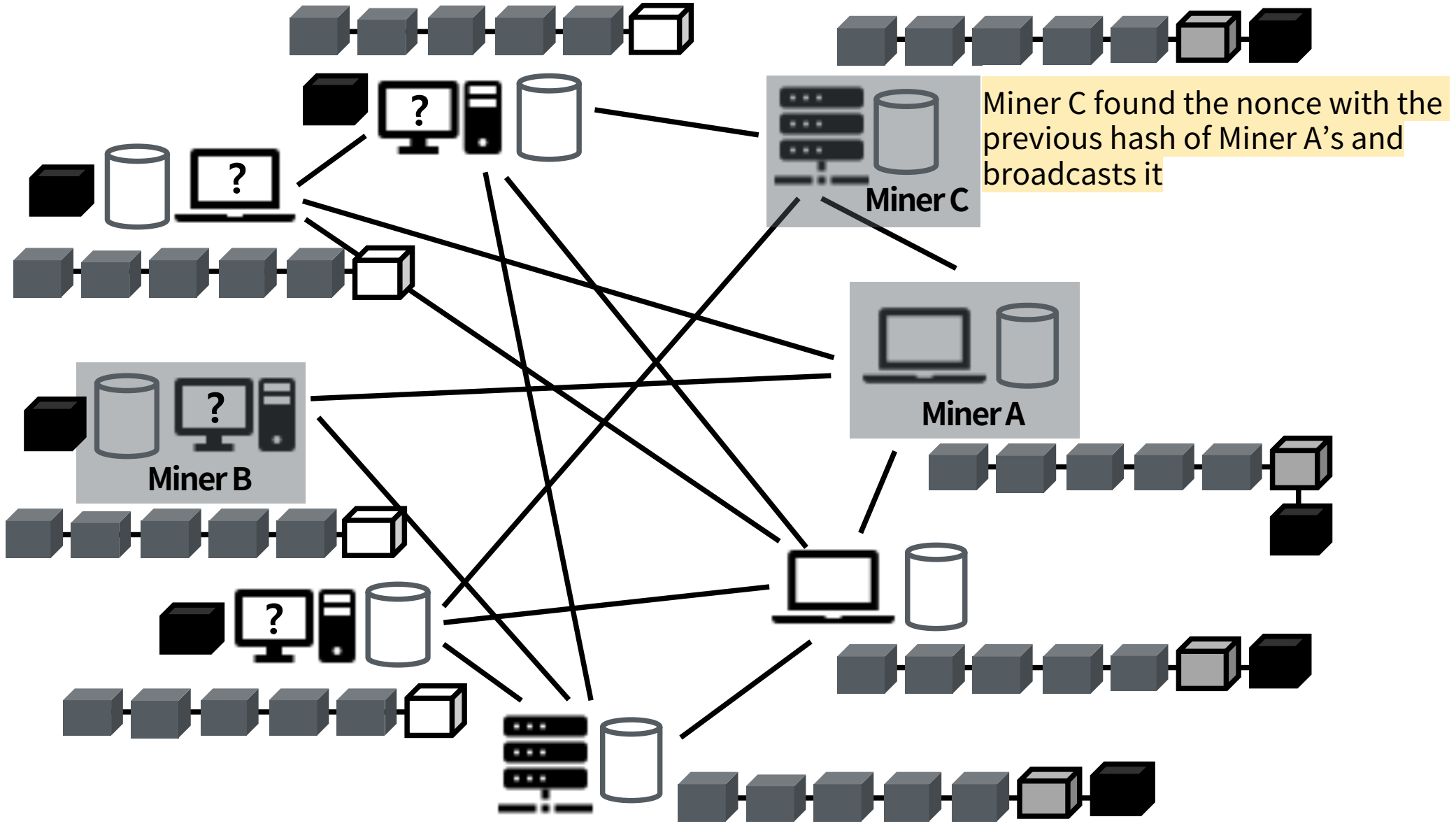
블록체인 충돌 해소 과정


Miner A found the nonce
And broadcasts the block
Miner A
Miner B

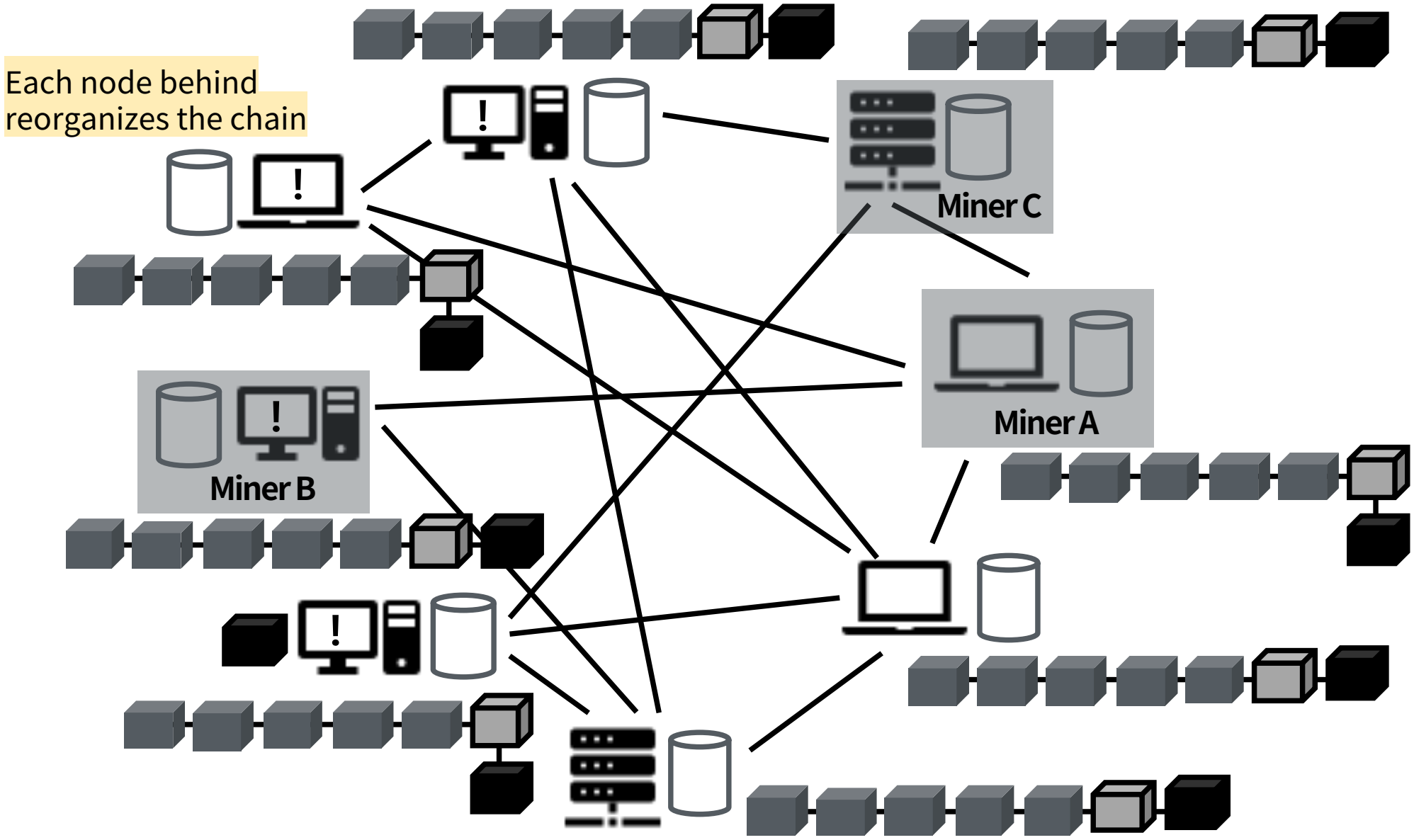
블록체인 충돌 해소 과정



블록체인 충돌 해소 과정

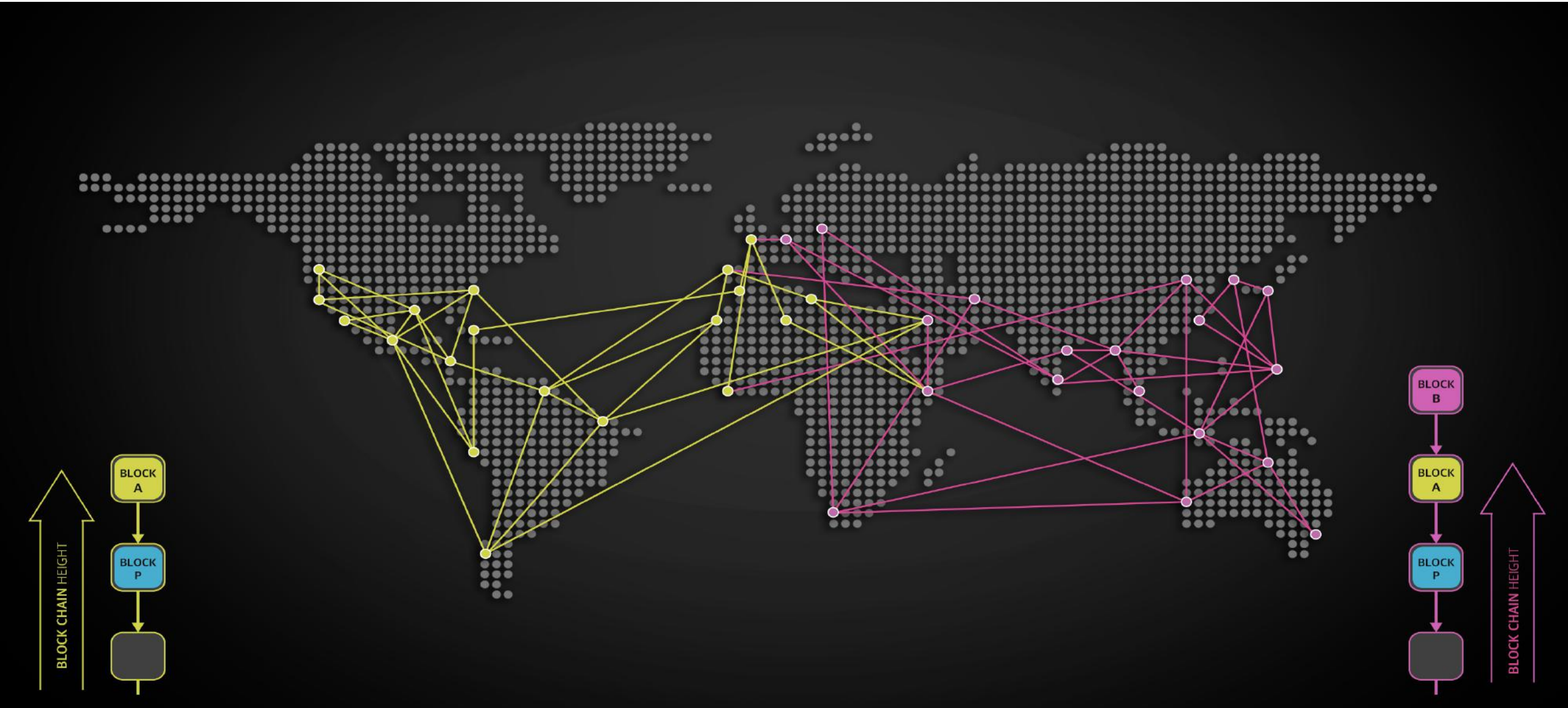

Miner C found the nonce with the previous hash of Miner A's and broadcasts it
Miner C
Miner A
Miner B

블록체인 충돌 해소 과정


Each node behind
reorganizes the chain
Miner C
Miner A
Miner B

블록체인 충돌 해소 과정

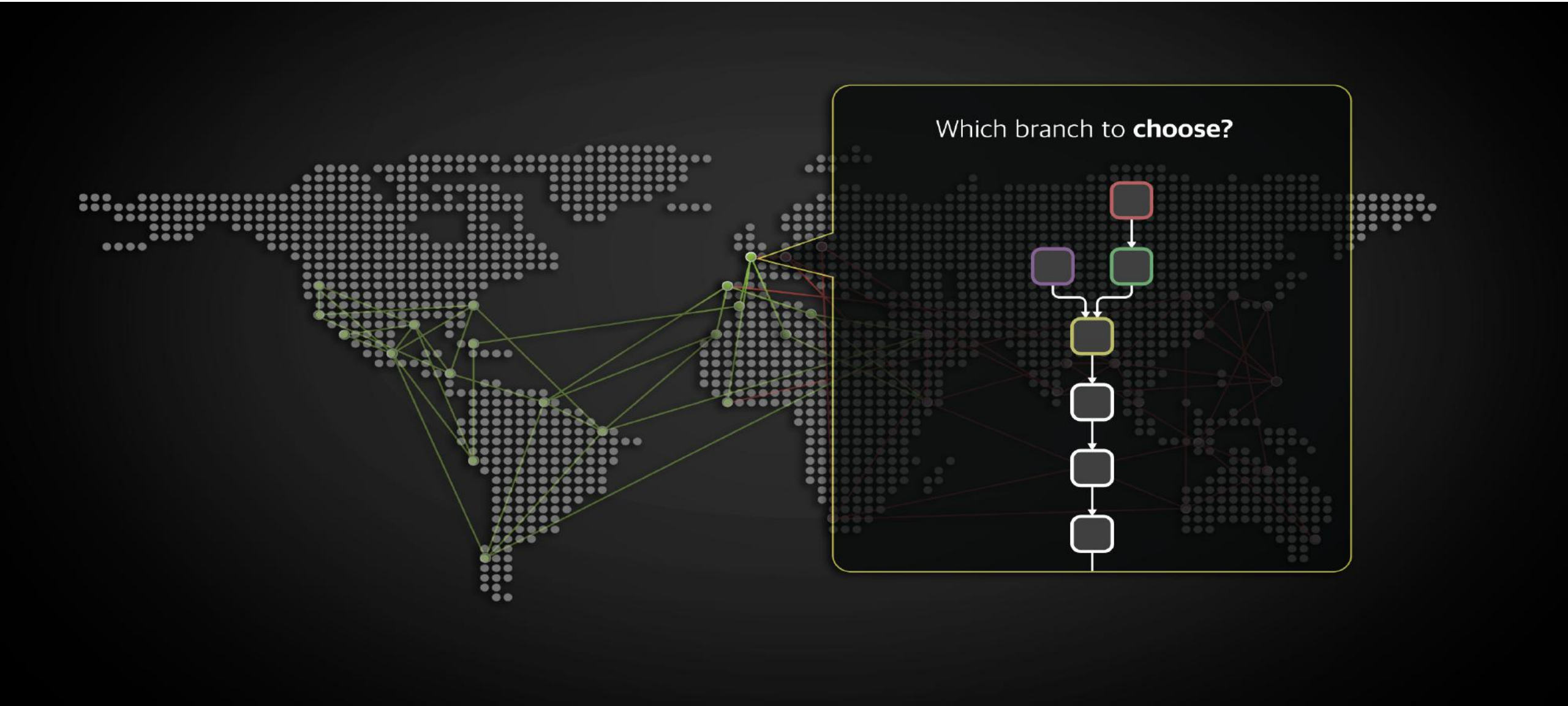
블록체인 충돌 해소 과정


BLOCK
A
BLOCK
P
BLOCK CHAIN HEIGHT
BLOCK
B
BLOCK
A
BLOCK
P
BLOCK CHAIN HEIGHT

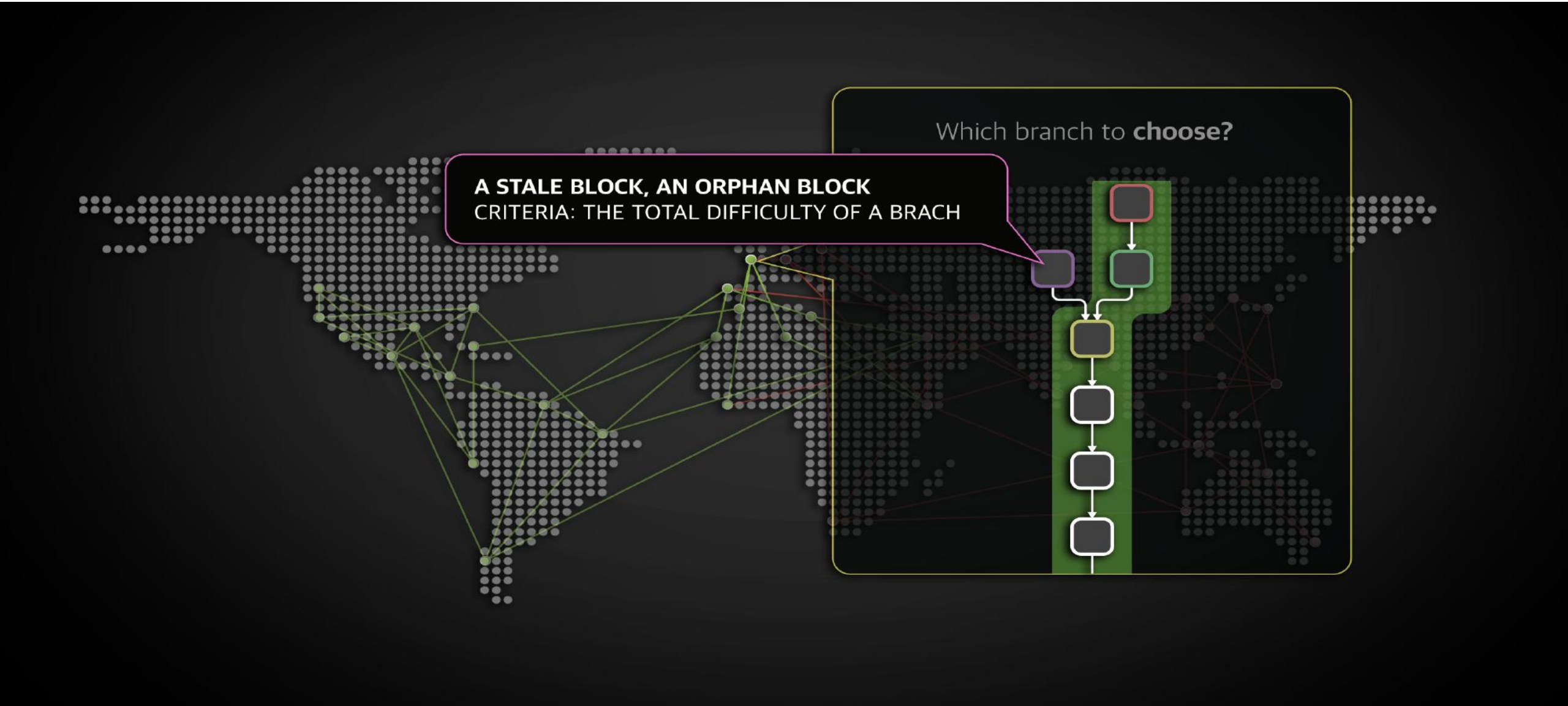
블록체인 충돌 해소 과정


BLOCK
C
BLOCK
A
BLOCK
P
BLOCK CHAIN HEIGHT
BLOCK
B
BLOCK
A
BLOCK
P
BLOCK CHAIN HEIGHT

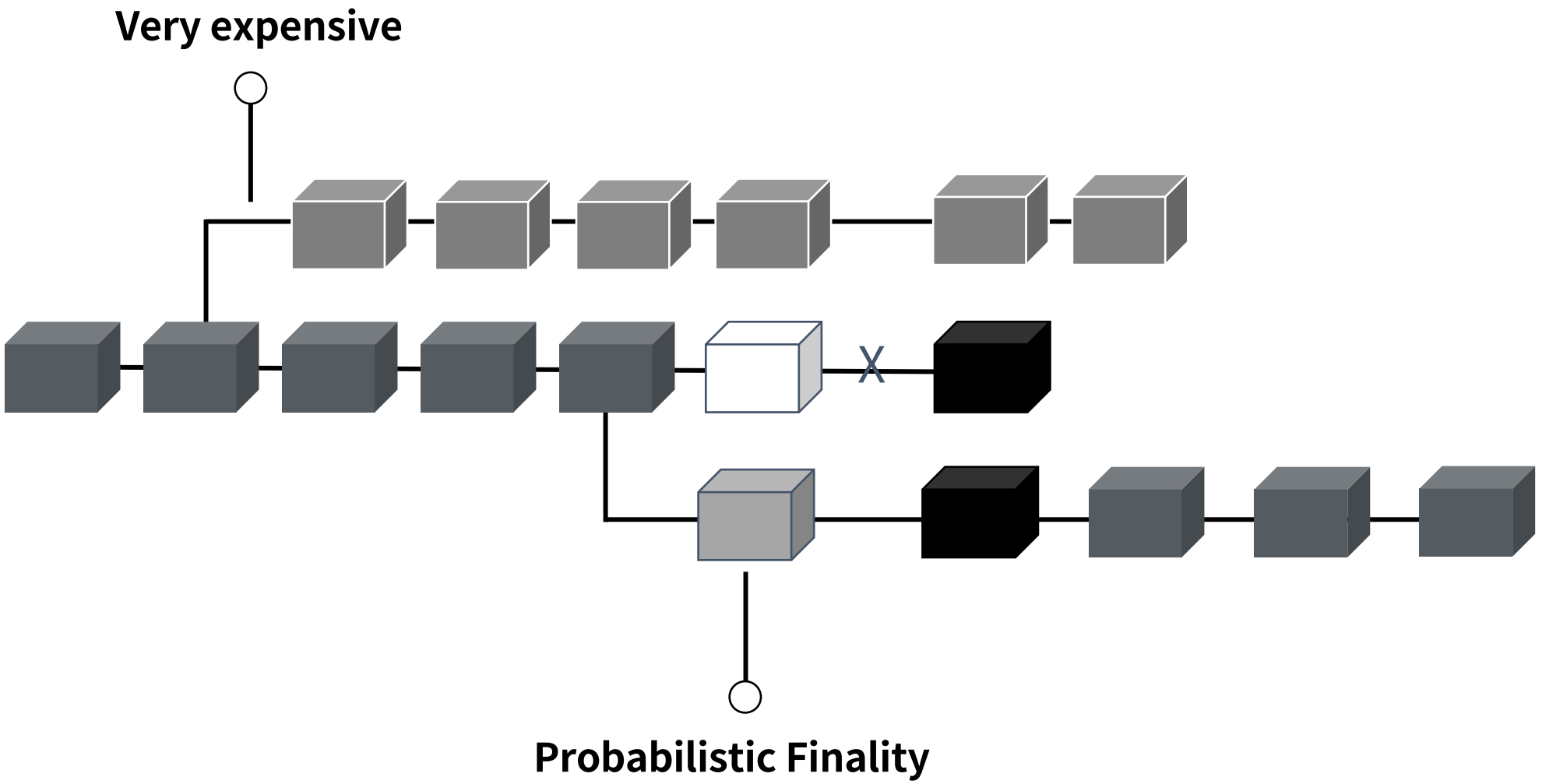
블록체인 충돌 해소 과정



블록체인 충돌 해소 과정


Which branch to **choose?**
A STALE BLOCK, AN ORPHAN BLOCK
CRITERIA: THE TOTAL DIFFICULTY OF A BRACH

블록체인 충돌 해소 과정

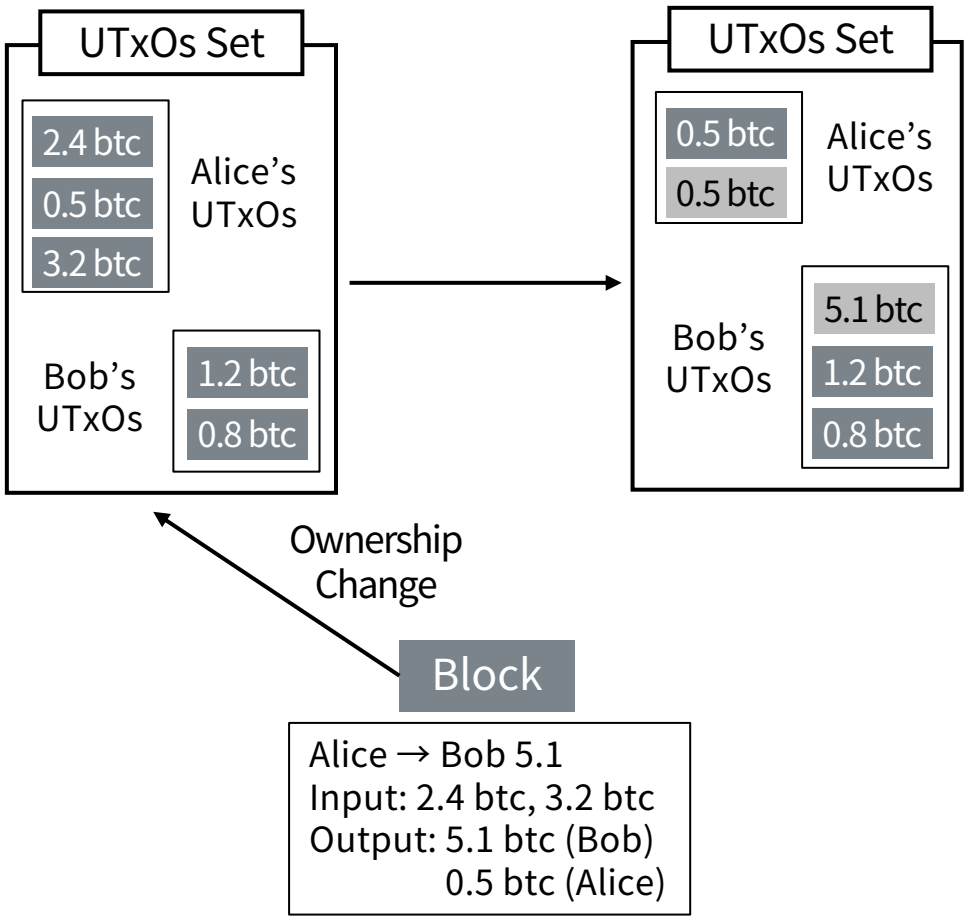


RULES

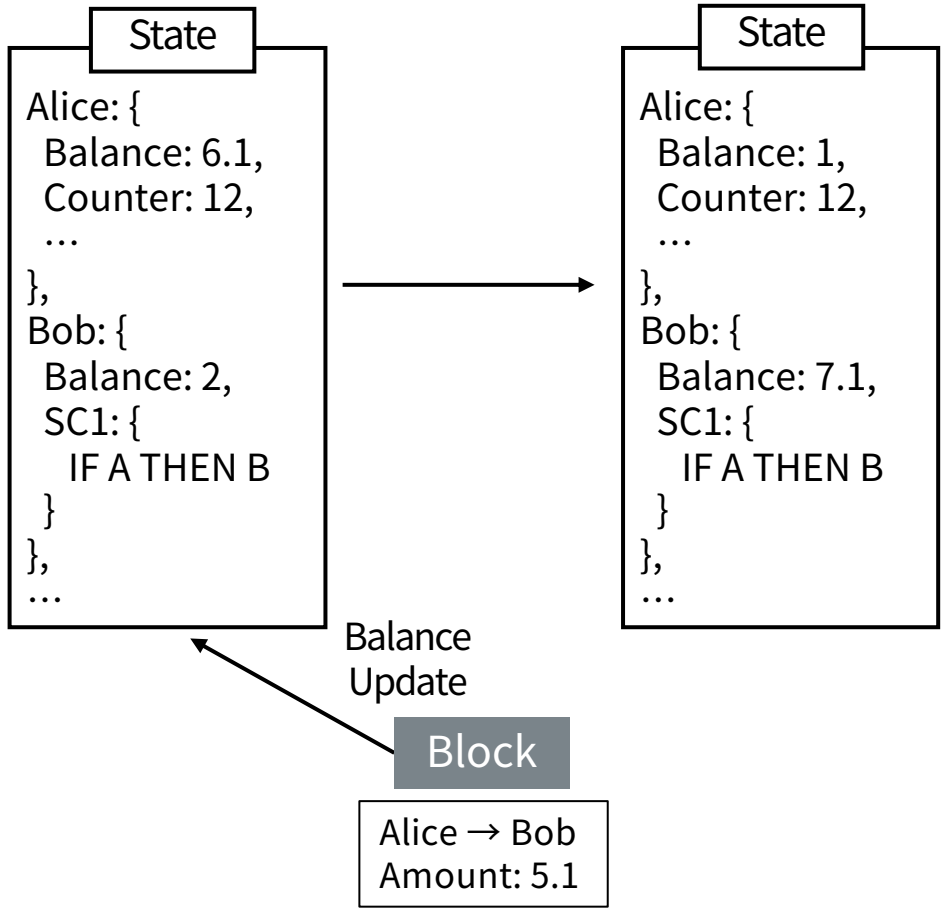
1. 비트코인을 송금하는 사람은 거래에 자신의 디지털 서명을 포함시켜야 한다.
2. 각 블록은 이전 블록의 해시, 오퍼레이션과 논스로 구성된다.
3. 제일 먼저 블록의 해시를 난이도보다 낮게 만든 채굴자가 보상을 받는다.
4. 채굴자는 블록에 보상(코인베이스 트랜잭션)을 포함시킨다.
5. 새 블록의 해시는 $\mathcal{H}ash$(이전 블록의 해시 + 오퍼레이션 + 논스)이며, 논스를 바꿔가며 목표 해시를 찾는다.
6. 누적 난이도 총합이 가장 큰 체인이 적법한 체인이다.

Abstract Blockchain

UTxO vs. Account-based Model

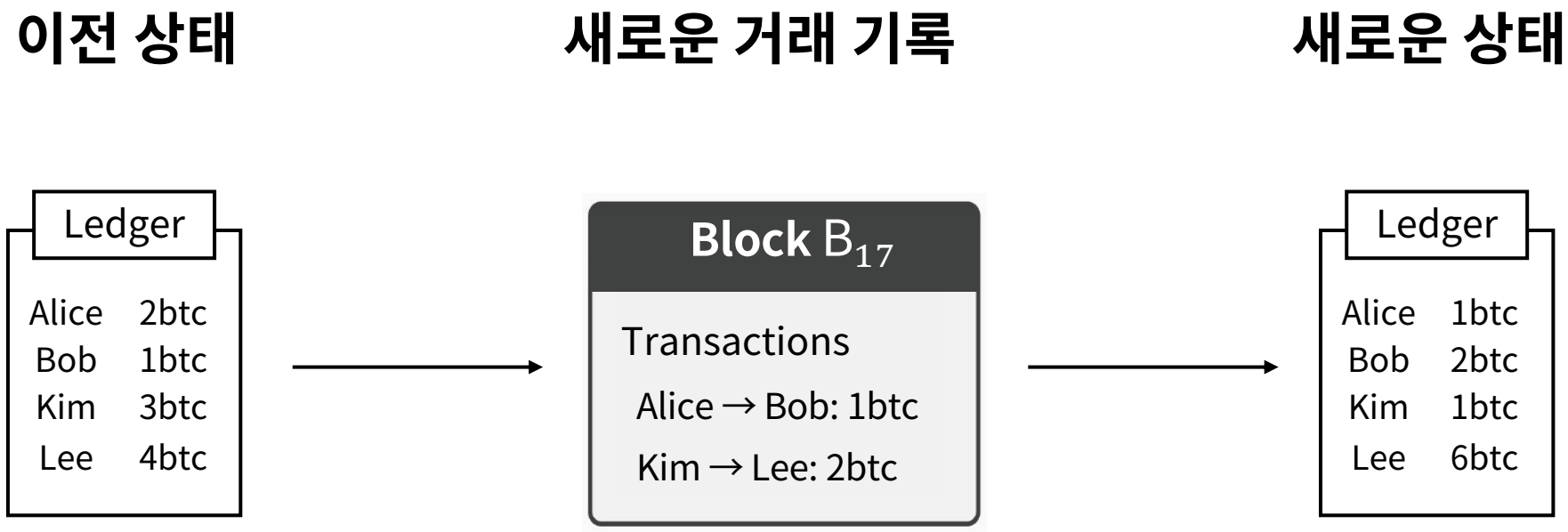


The current **sum of UTxOs** of Alice?



The current **value of balance** in Alice's account?

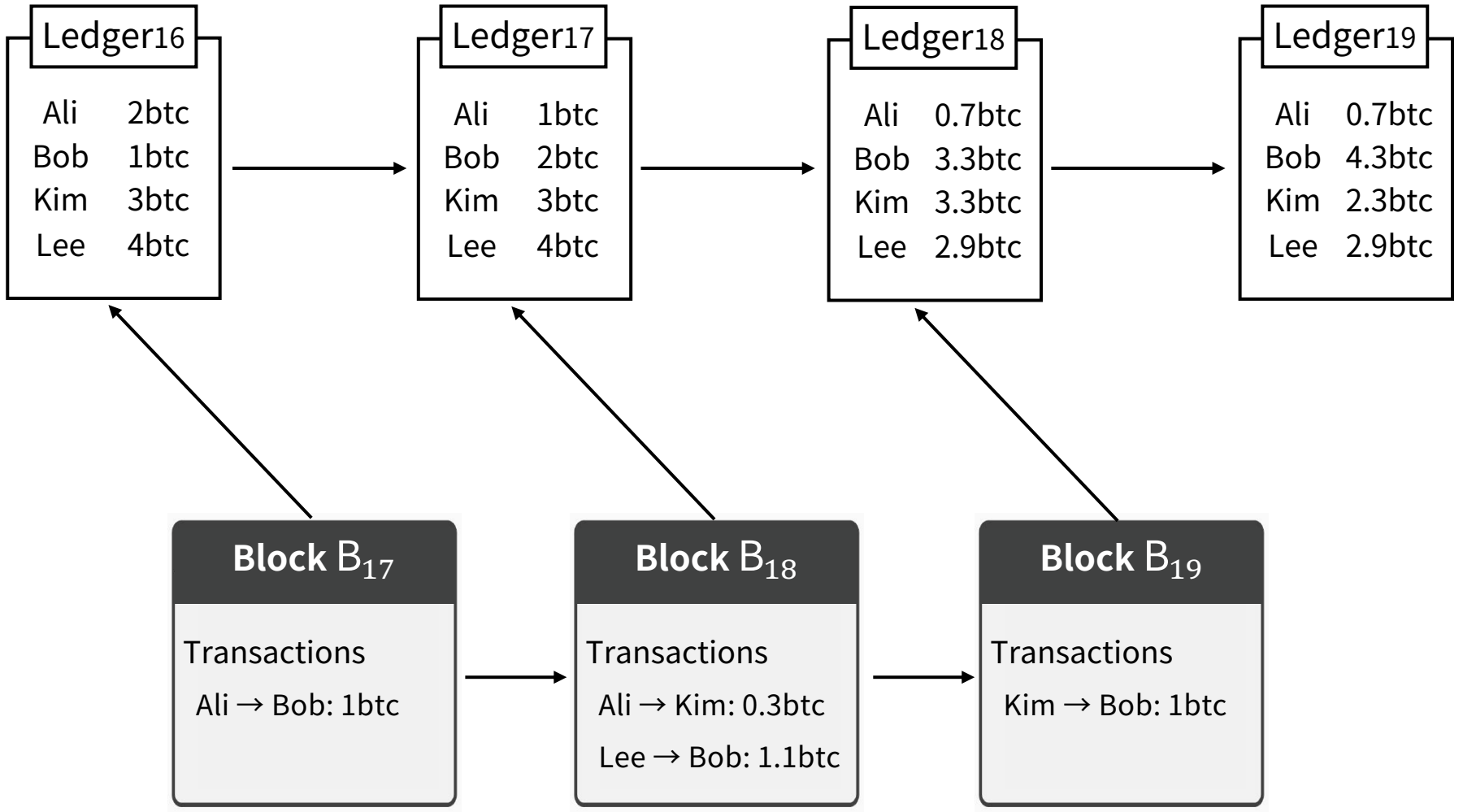
Block as an **Operator**



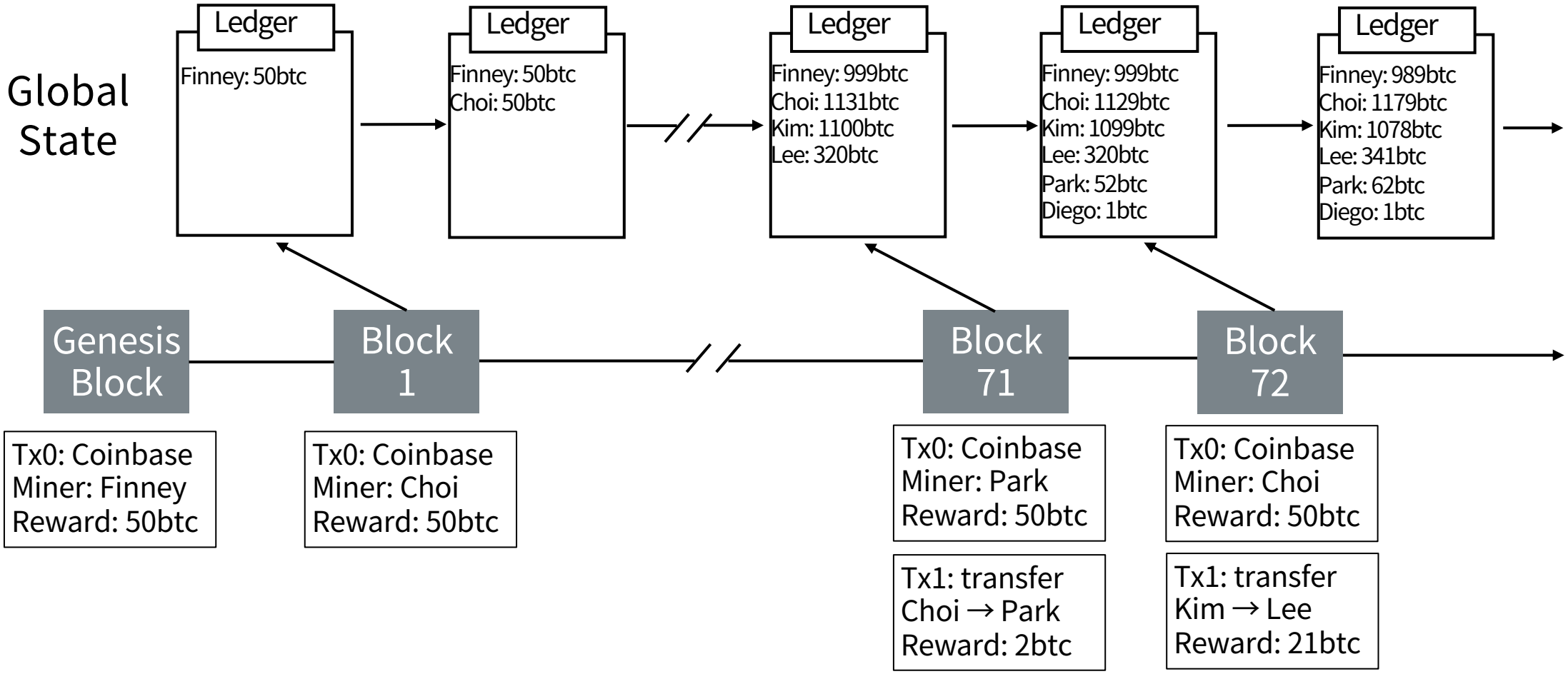
Blockchain = A chain of blocks

거래의 결과 (상태 값)

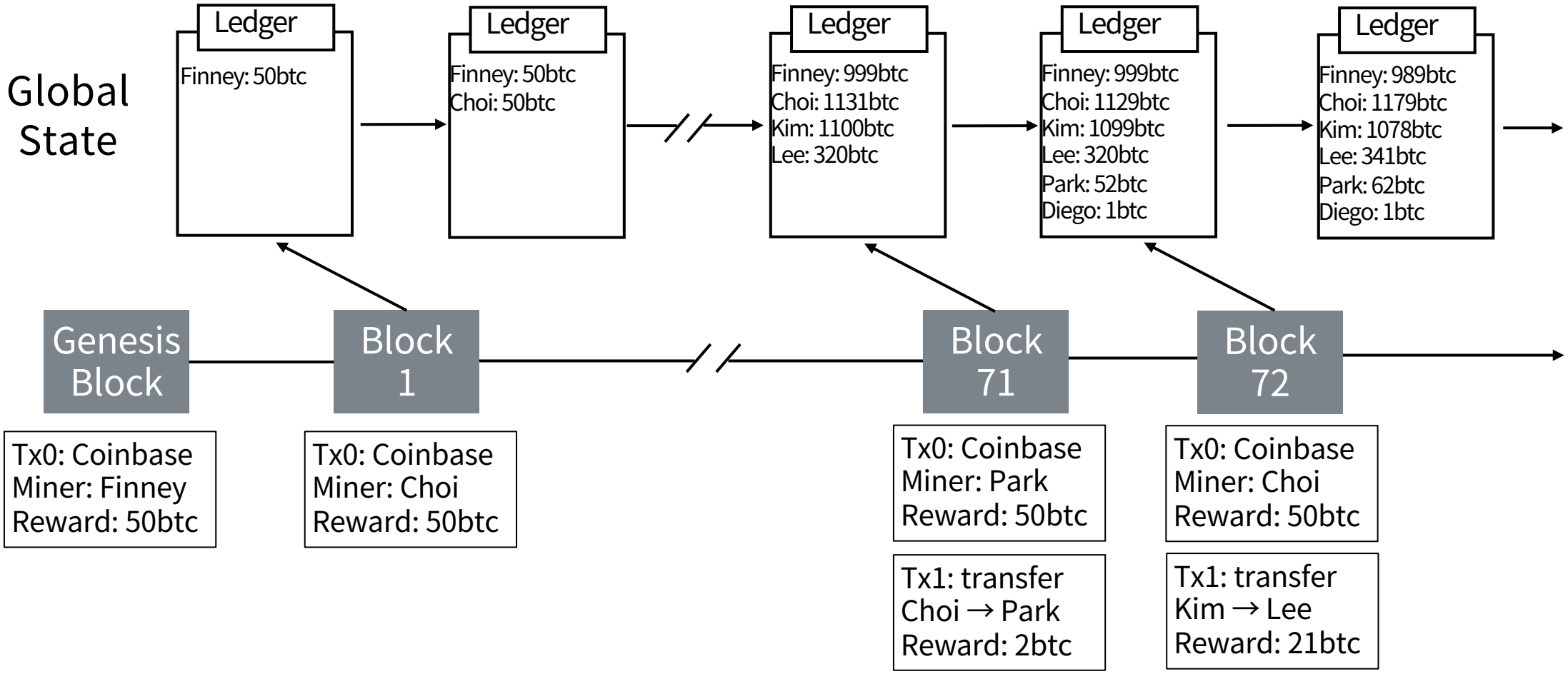
거래 기록들 (블록체인)



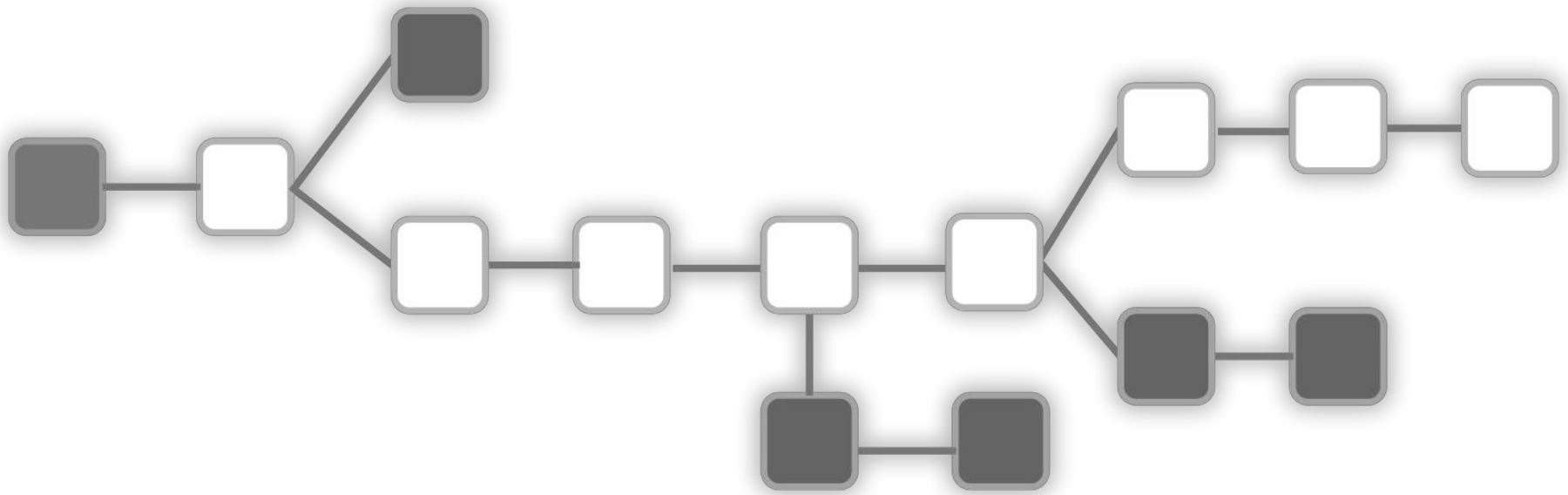
Block as an **Operator**



Block as an **Operator**



Abstract Blockchain



A blockchain

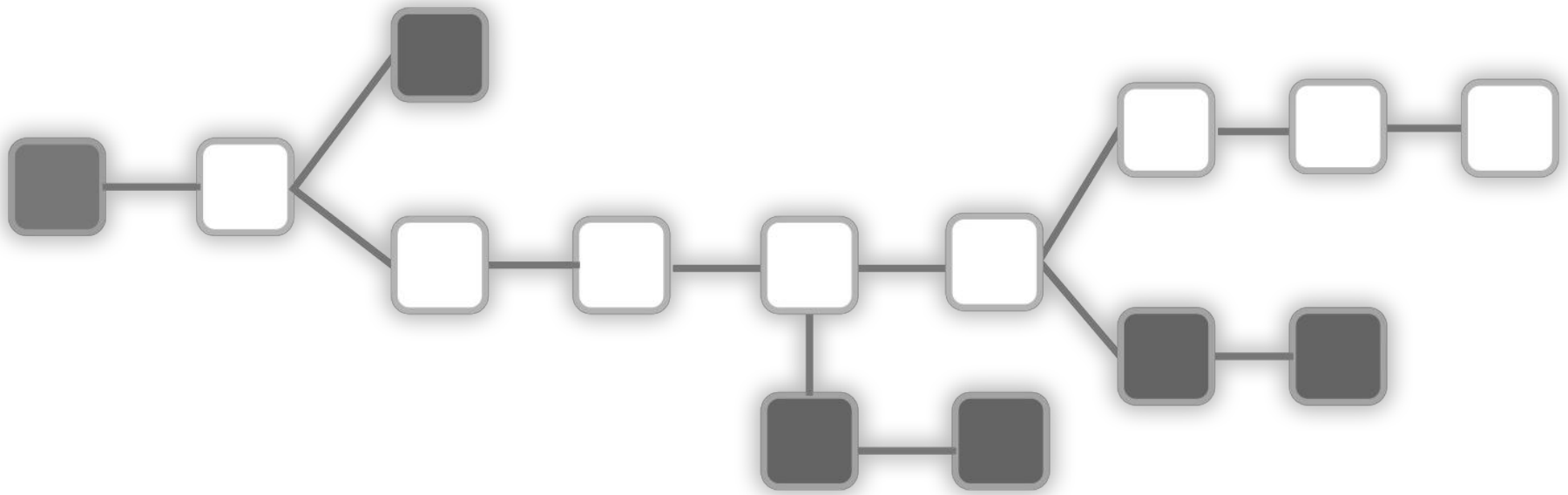
represents **a single state** being concurrently edited and

keeps the state as **a shared ledger** to avoid conflicts

as a series of **transformations** applied to an initial state (genesis state).

A set of blocks, each bundling together multiple operations in it.

Abstract Blockchain

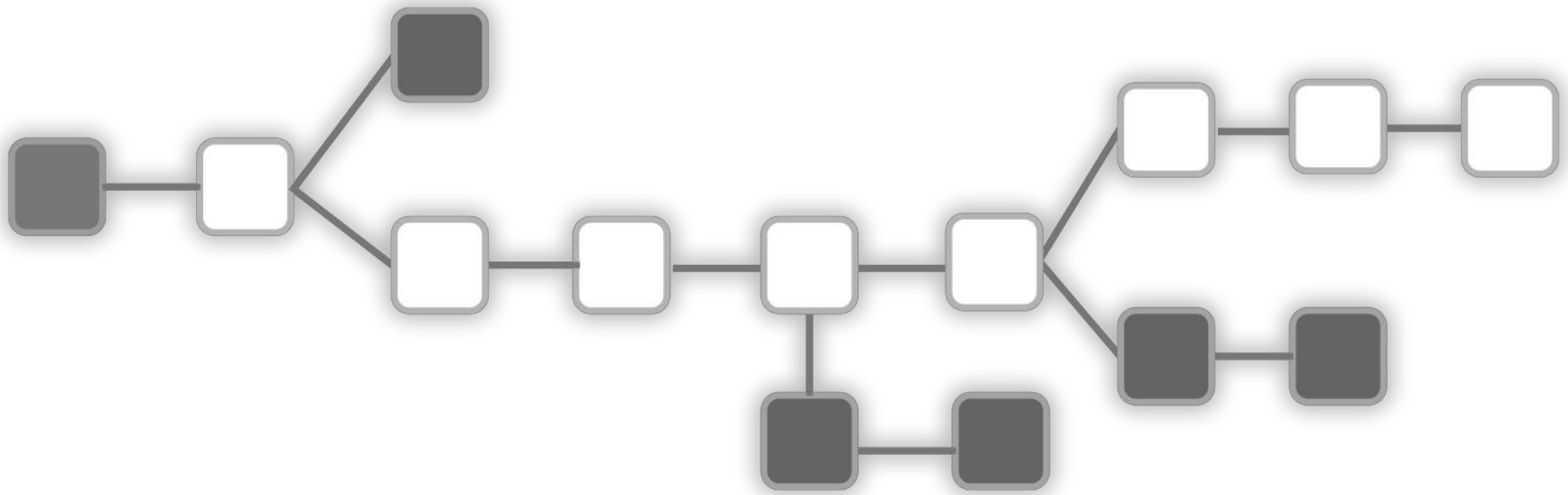


A blockchain protocol

is a monadic implementation of **concurrent mutations** of **a global state** through a series of operations such that blocks are defined as operators.

apply: $S \mapsto O \mapsto S$ or $S' = f(S, O)$

Abstract Blockchain



Blocks

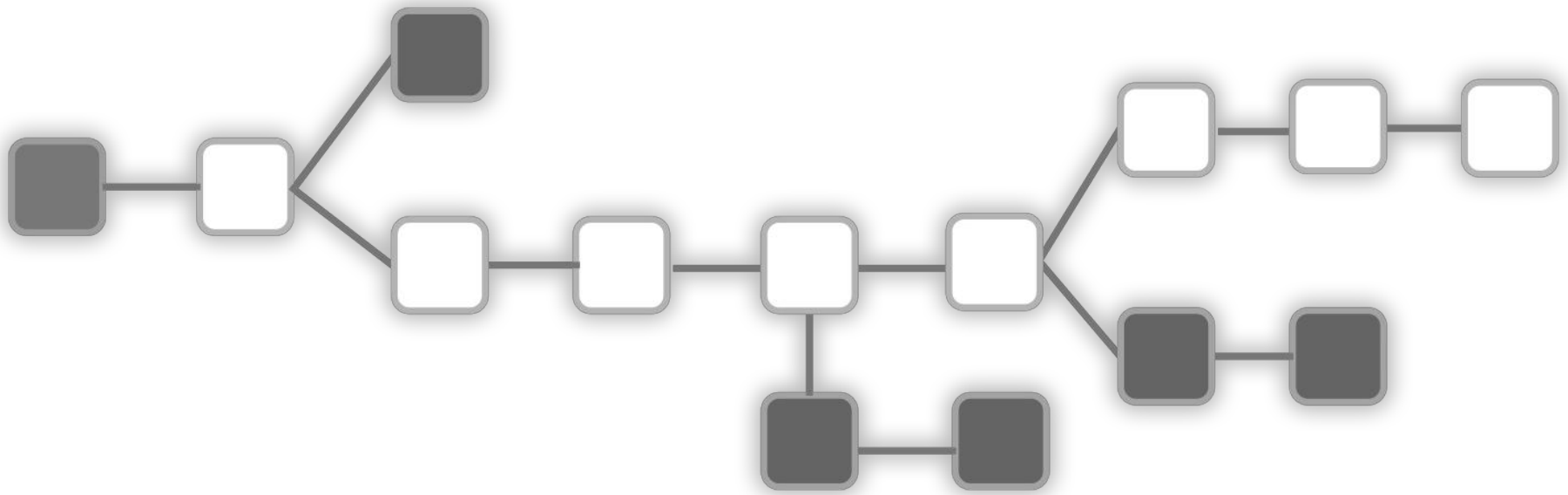
are created asynchronously by many concurrent nodes

forms a **tree structure**,

which needs **well-ordering** for choosing **the valid one** (unique, canonical).

score : $S \mapsto N$ or $N = f(S)$

Abstract Blockchain

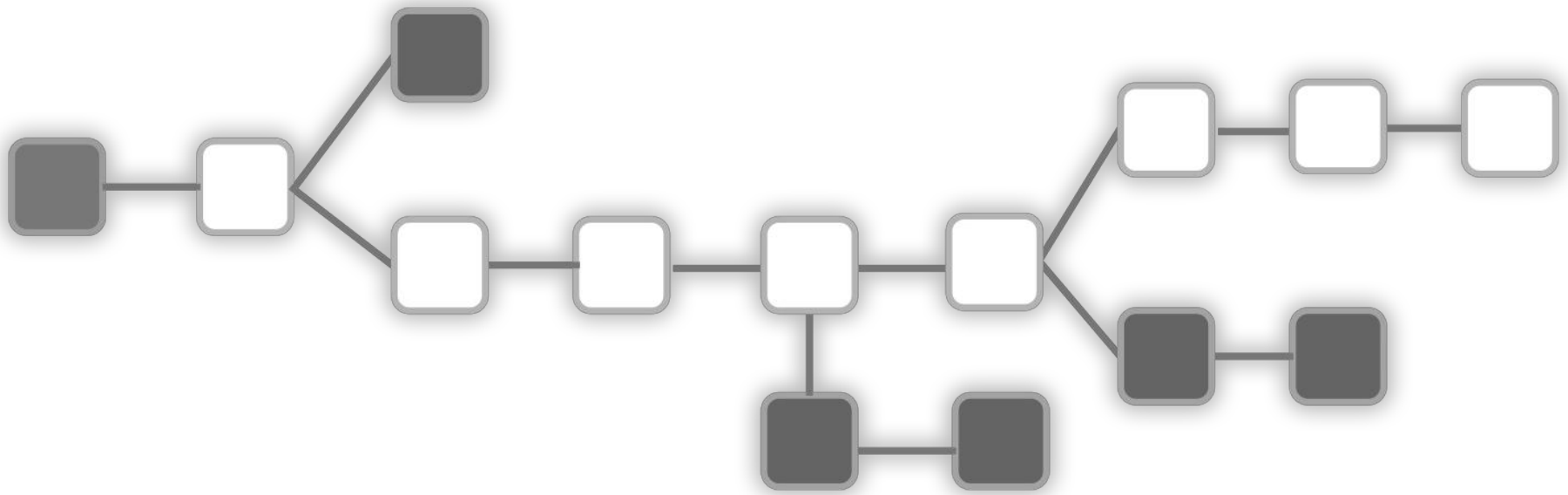


All existing block chain implementation generally

apply: $S \mapsto O \mapsto S$ or $S' = f(S, O)$

score: $S \mapsto N$ or $N = f(S)$

Abstract Blockchain



Bitcoin

State = A set of UTxOs + total work + block index

Operations = transactions

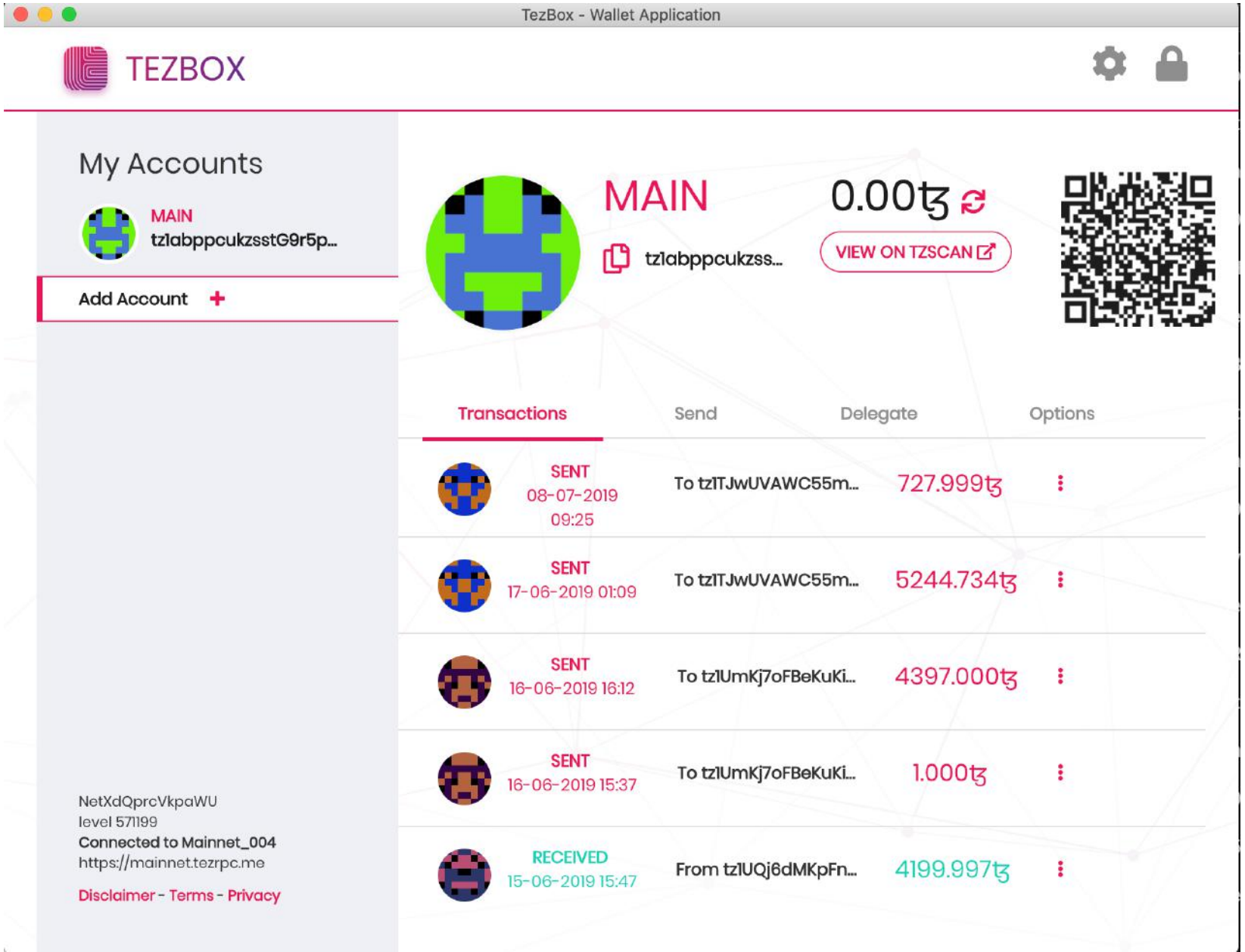
Score = the one with the greatest total difficulty

블록체인 간접 체험

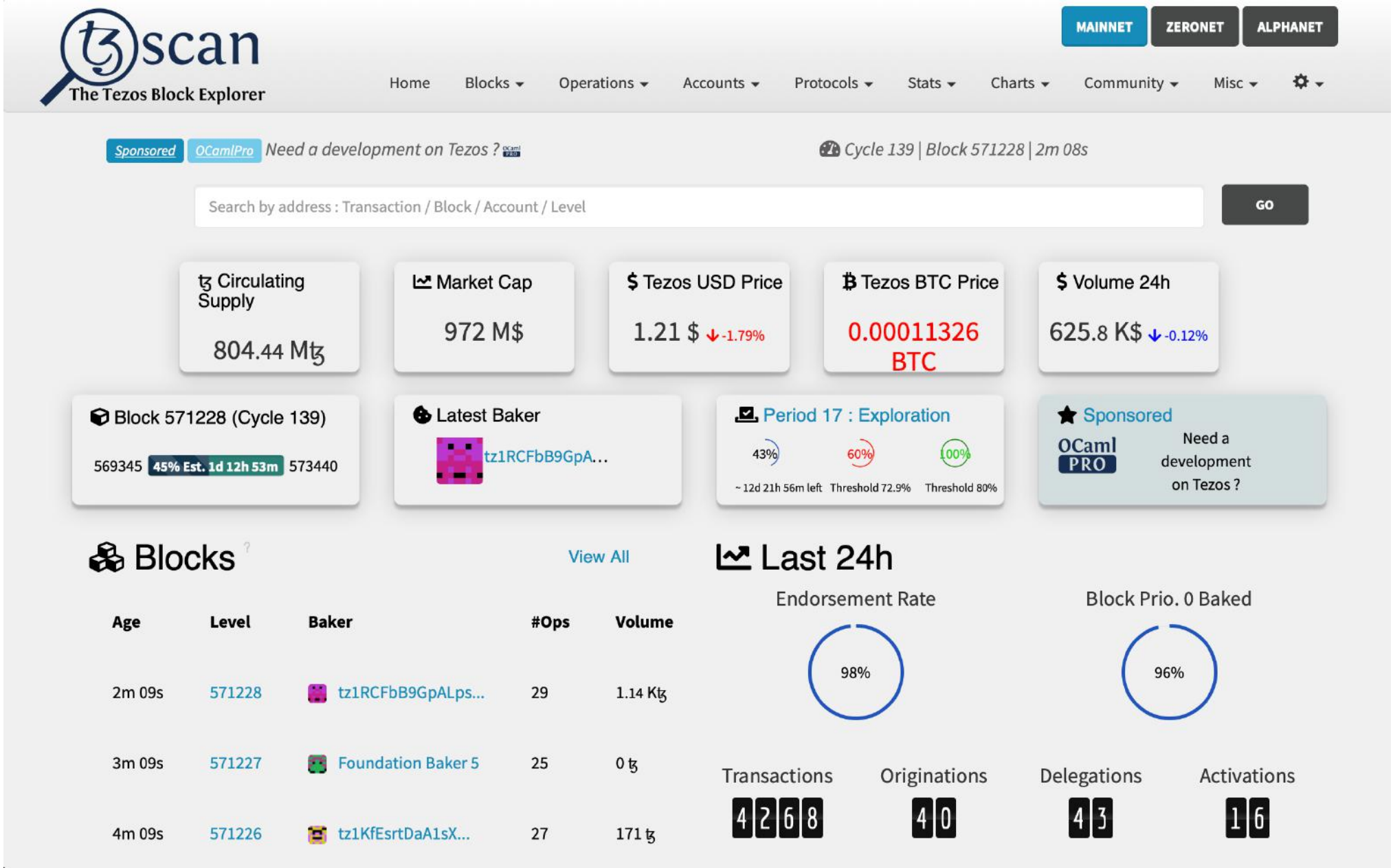
Tezos Wallet Application (Built-in)

```
Aug 19 21:33:34 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:34-00:00, Treat
ed: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.chain_1: Update current head to BMQD78RZdBWJSCMuH5anyPkZJrLj61fai9bvRcaZE
2GzPi6QKYT (fitness 00::0000000001104652), same branch
Aug 19 21:33:34 - validator.chain_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-
00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.block: Block BLxNjiejt3NJKxXxCj36igEiTaqnhDyi4QWF7r1tapghH17H6MT successf
ully validated
Aug 19 21:33:34 - validator.block: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00
:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: switching to new head BLxNjiejt3NJKxXxCj
36igEiTaqnhDyi4QWF7r1tapghH17H6MT
Aug 19 21:33:34 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:34-00:00, Treat
ed: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.chain_1: Update current head to BLxNjiejt3NJKxXxCj36igEiTaqnhDyi4QWF7r1ta
pghH17H6MT (fitness 00::0000000001104672), same branch
Aug 19 21:33:34 - validator.chain_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-
00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:35 - validator.block: Block BKpBPJjkRMtfhPN1RthqCD2kprzxgVsBhgA6kv7Ah4CaBkSJvZZ successf
ully validated
Aug 19 21:33:35 - validator.block: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00
:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:35 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: switching to new head BKpBPJjkRMtfhPN1Rt
hqCD2kprzxgVsBhgA6kv7Ah4CaBkSJvZZ
Aug 19 21:33:35 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:35-00:00, Treat
ed: 2019-08-19T12:33:35-00:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:35 - validator.chain_1: Update current head to BKpBPJjkRMtfhPN1RthqCD2kprzxgVsBhgA6kv7Ah
4CaBkSJvZZ (fitness 00::0000000001104693), same branch
Aug 19 21:33:35 - validator.chain_1: Pushed: 2019-08-19T12:33:35-00:00, Treated: 2019-08-19T12:33:35-
00:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:36 - validator.block: Block BKx6gUL9EsUsJdeRRaZhsMabqDt8cs3pC7ReNTD3YX6Y4oDgFu8 successf
ully validated
Aug 19 21:33:36 - validator.block: Pushed: 2019-08-19T12:33:35-00:00, Treated: 2019-08-19T12:33:35-00
:00, Completed: 2019-08-19T12:33:36-00:00
Aug 19 21:33:36 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: switching to new head BKx6gUL9EsUsJdeRRa
ZhsMabqDt8cs3pC7ReNTD3YX6Y4oDgFu8
Aug 19 21:33:36 - prevalidator.NetXdQprcVkpa.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:36-00:00, Treat
```

Tezos Wallet Application (TezBox)



Tezos Block Explorer (TzScan)



테조스 플랫폼

Tezos , **the last** crypto currency

Tezos: A Self-Amending Crypto-Ledger Position Paper (2014)

Conclusion

We've presented issues with the existing cryptocurrencies and offered Tezos as a solution. While the irony of preventing the fragmentation of cryptocurrencies by releasing a new one does not escape us, Tezos truly aims to be the *last* cryptocurrency.

No matter what innovations other protocols produce, it will be possible for Tezos stakeholders to adopt these innovations. Furthermore, the ability to solve collective action problems and easily implement protocols in OCaml will make Tezos one of the most reactive cryptocurrency.

Tezos , **the last** crypto currency

Problems

The Protocol Fork Problem

Shortcomings of Proof-of-Work

Smart Contracts

Correctness

Contents

Tezos , **the last** crypto currency

Problems

- **The Protocol Fork Problem**
- **Shortcomings of Proof-of-Work**
- **Smart Contracts**
- **Correctness**

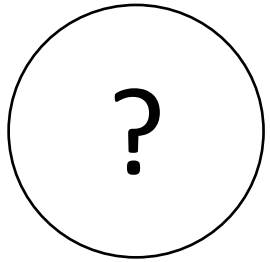
Solutions

- **Self-Amending & On-chain governance**
- **LPoS**
- **Smart Contracts**
- **Formal Verification**
- **Abstract Blockchain**

Tezos , **the last** crypto currency

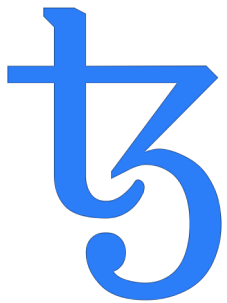
1st Gen. Bitcoin
First ever Cryptocurrency
and blockchain

2nd Gen. Ethereum
Turing complete
Smart contract



Who's the next?
Scalability
Inter-operability
Sustainability

Motivations
Protocol fork
Proof of Work
Smart Contract
Correctness



Abstract blockchain
Self-Amending
On-chain governance
LPoS
Smart contract
Formal Verification
Abstract Blockchain

PoS 합의 알고리즘

Why PoS? No more PoW

Inefficient

PoW의 지나친 에너지 소모와 환경 파괴

Hash puzzle 경쟁으로 인한 중복 계산

Centralization

ASIC과 마이닝 풀의 등장

Mis-alignment of interest

채굴자의 동기는 채굴 보상

(ex. 자동 해시 파워 이동 서비스)

Not scalable

Hash puzzle 경쟁으로 인한 중복 계산

Not solved by just shorter block interval (more stale blocks)

But why PoW is still the King?

Time Tested

Old-school blockchains (Bitcoin, Litecoin, Ethereum) are using PoW

Bitcoin network has never been hacked

Secure

Real proof by burning physical resources

Hard to attack

(Dis)Incentive – Block reward and computing power is a kind of stake

Fair

Very fair guaranteed by cryptography

What is PoS?

PROOF

블록에 포함된 트랜잭션과 해당 블록이 적법하다는 증거 또는 증명

STAKE

특정 퍼블릭 키 해시(주소)가 보유한 지분의 상대적 가치

특정 주소의 staking 토큰 수 / 네트워크 전체 staking 토큰 수

PROOF OF STAKE

지분의 비율에 따라 선택된 주체가 새 블록을 생성

FOLLOW-THE-SATOSHI (Follow-the-coin)

새롭게 생성(mined, minted)된 토큰의 **모든 최소 단위에** 고유한 시리얼 넘버를 부여

시리얼 넘버 한 개를 (무작위로) 선택하고, 이 토큰을 보유한 주소가 블록을 생성함

더 많은 토큰을 보유할수록, 블록 생성 확률이 높음

PoS also should be **fair** and **secure**

Fair & Secure

블록 생성의 기회가 지분의 비율에 따라 공정하게 주어져야 함
스케줄 생성에 개인이 큰 영향을 끼쳐서는 안 됨
이미 정해진 스케줄이 변경되어서도 안 됨
악의적인 행동을 효과적으로 막을 수 있어야 함 (Nothing at stake, Long range attack)

PoS also should be **fair** and **secure**

Validator selection rule

Disincentive for malicious behavior

Incentive for honest behavior

PoS also should be **fair** and **secure**

Validator selection rule

Disincentive for malicious behavior

Incentive for honest behavior

PoS of Tezos, **A mix of several ideas**

Validator Selection Rule	**Multiple snapshots** **Random seed**
Dis-incentive for Malicious Behavior	**Safety deposit** **Plenitude rule**
Incentive for Honest Behavior	**Block reward** **Accussing reward**

PoS of Tezos, **A mix of several ideas**

Validator Selection Rule	**Multiple snapshots Random seed**
Dis-incentive for Malicious Behavior	**Safety deposit Plenitude rule**
Incentive for Honest Behavior	**Block reward Accussing reward**

용어 정리

Baking

Block producing , Staking

Baker

Block producer, validator, miner, forger

Delegation

Only staking right, NOT ownership

Cycle

Period defined in protocol

4,096 Blocks (1 block = 1 min)

Validator selection rule = Roll snapshot + Random seed

Follow the roll

1 roll = 8,000 xtz (베이커 보유분 + 위임 받은 수량)

베이킹 파워가 롤 단위로 내림(Rounded down)

8,000 ~ 15,999 xtz = 1 roll

FTS 알고리즘 측면에서는 효율적

Roll snapshot을 통해 베이킹 파워를 측정

PoS with Optional Delegation

Validator selection rule = Roll snapshot + Random seed

Random seed

베이커들이 결정

예측이 사실상 불가능한 무작위 숫자열

Validator selection rule = Roll snapshot + Random seed

Roll snapshot

지분을 얼마나 보유하고 있는가

+

Random seed

예측이 사실상 불가능한 무작위 숫자열

Baking rights

블록 B는 베이커 A가 생성한다

Validator selection rule = Roll snapshot + Random seed

Cycle

1 Cycle = 4,096 블록 = 4,096분 = 2일 20시간 16분 (best case)

사이클 단위로 베이킹 스케줄이 **미리** 결정됨

Baking rights in cycle 1

Block 4,097: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)

Block 4,098: 베이커(Satoshi, Charles, Dan, Vitalik, Kwon, ...)

Block 4,099: 베이커(Justin, Kate, Arthur, Satoshi, Satoshi, ...)

…

Block 8,192: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)

Validator selection rule = Roll snapshot + Random seed

Block 8,192: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)

가장 높은 우선순위의 베이커는 이전 블록 생성 후 **1분** 후 베이킹 가능

2번째 우선순위의 베이커는 이전 블록 생성 후 **2분** 후 베이킹 가능

…

16분이 지난 경우, **보증금 없이** 베이킹 가능

체인별 블록 생성 시간을 통해서 체인의 score를 가늠할 수 있음

Validator selection rule = Roll snapshot + Random seed

Roll snapshot in **Cycle N**

지분을 얼마나 보유하고 있는가

+

Random seed in **Cycle N+1**

예측이 사실상 불가능한 무작위 숫자열

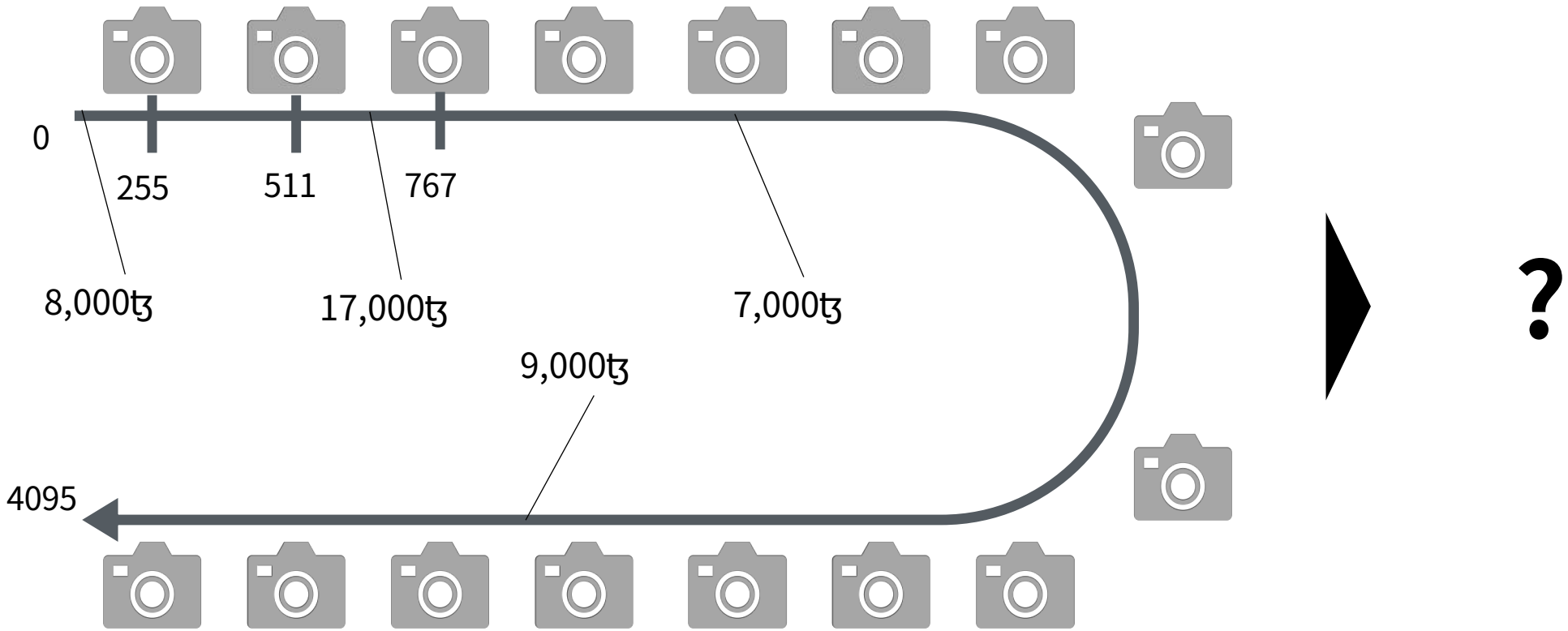
Baking rights in **Cycle N+7**

블록 B는 베이커 A가 생성한다

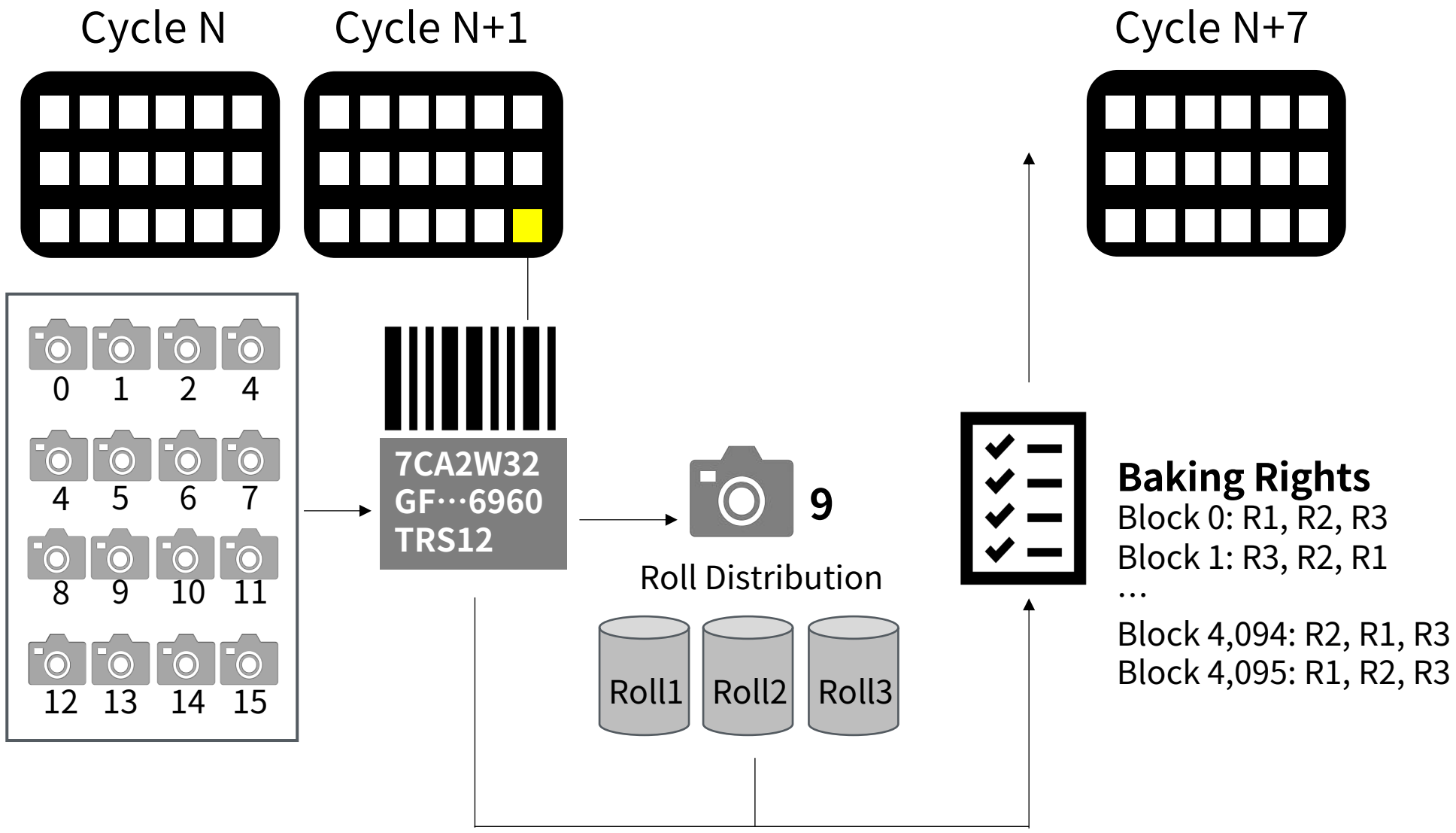
Validator selection rule = Roll snapshot + Random seed

Multiple snapshots (16 in a cycle) and random selection

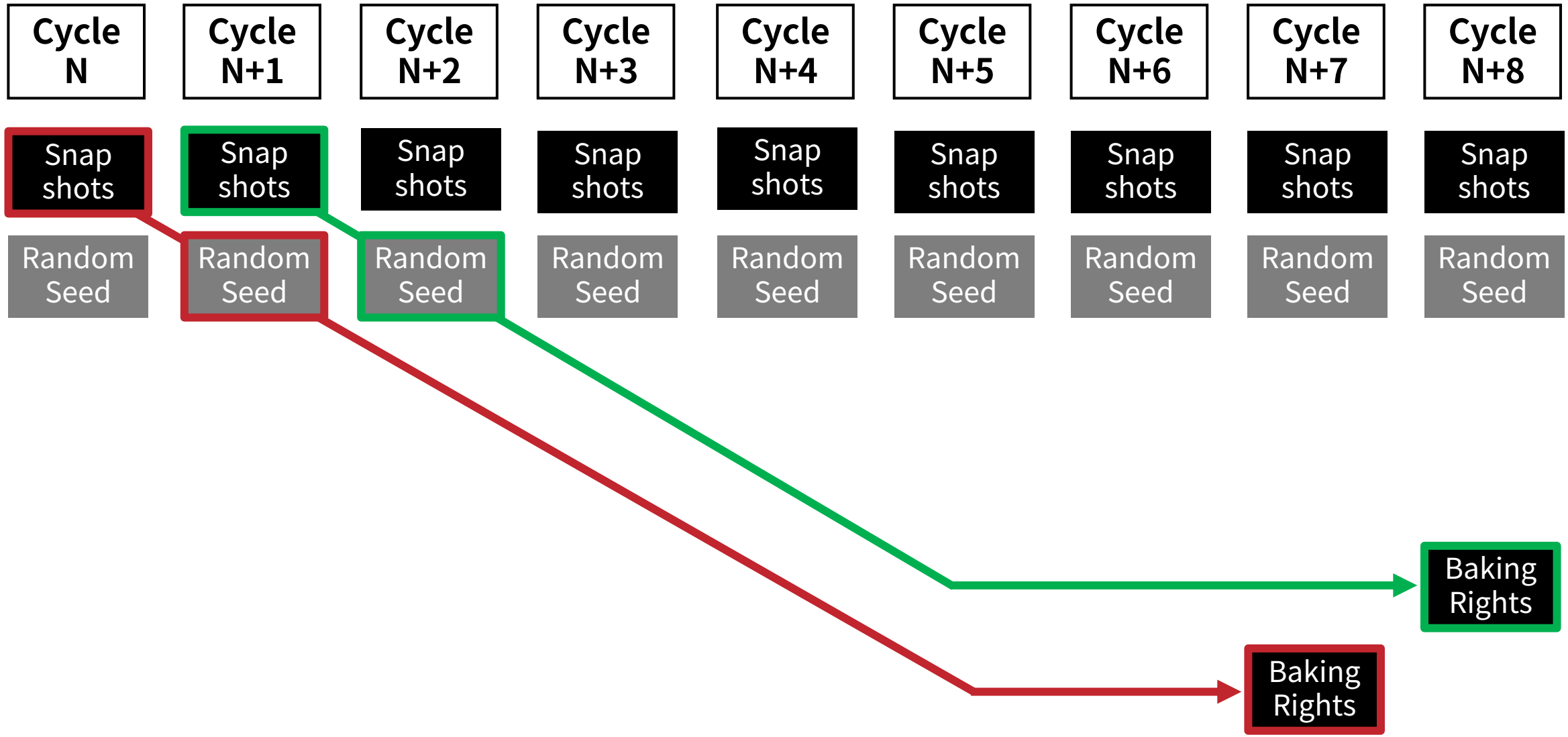
Proof of **STAKING**



Validator selection rule = Roll snapshot + Random seed



Validator selection rule = Roll snapshot + Random seed

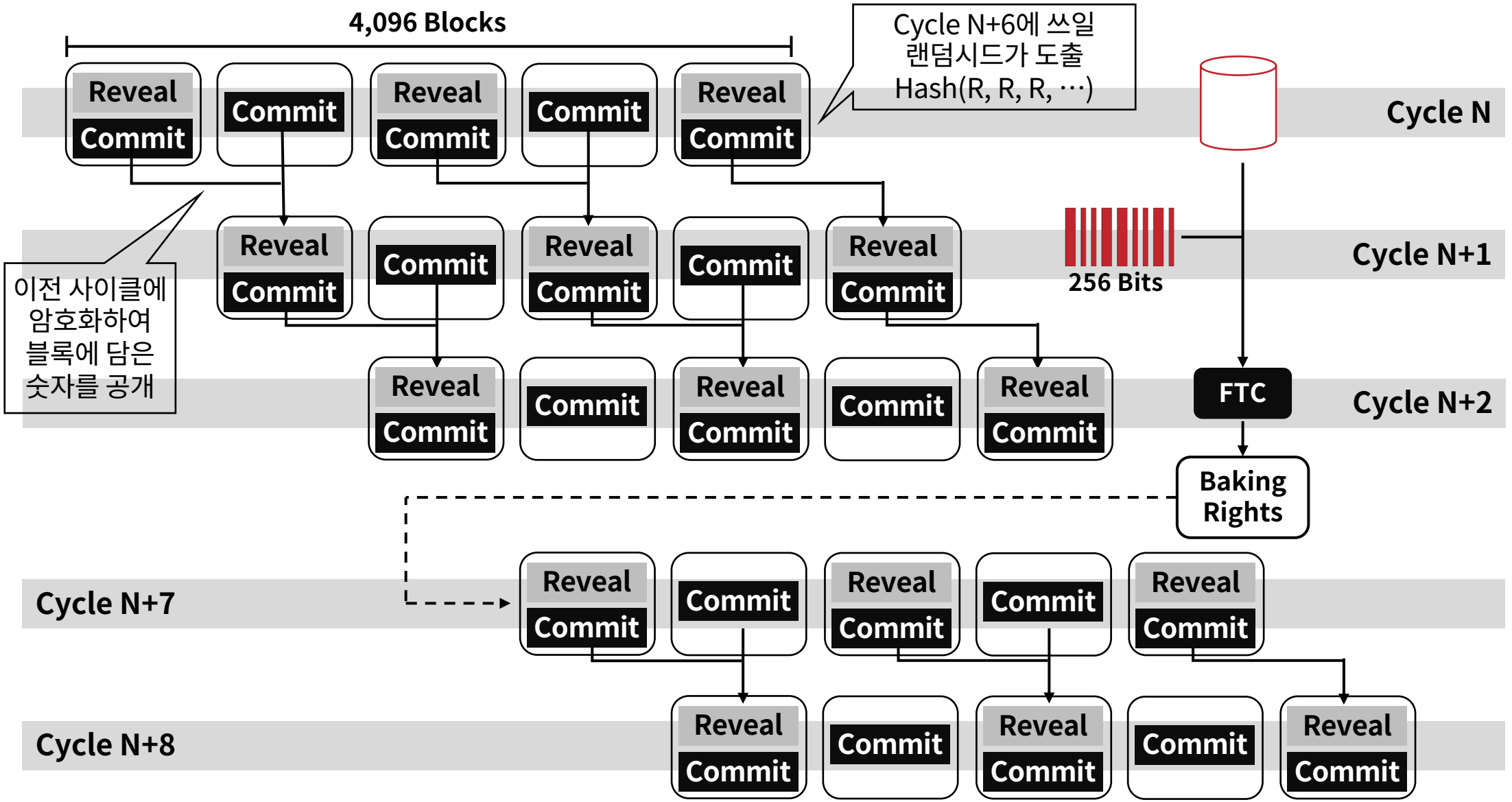


Validator selection rule = Roll snapshot + Random seed

Random seed

베이커는 미리 정해진 베이킹 스케줄에 따라 블록B를 생성
임의의 숫자를 암호화(**hash commitment**)하여 블록B에 포함
이 숫자는 다음 사이클에 공개(**reveal**)해야 함
숫자를 공개하지 않은 경우, 블록B의 보상과 수수료 몰수
매 사이클 마지막에 공개된 숫자를 모두 결합, 해싱하여 랜덤 시드 생성
공개는 32블록마다 자유롭게 가능

Validator selection rule = Roll snapshot + Random seed



PoS of Tezos, **A mix of several ideas**

Validator Selection Rule	Multiple snapshots Random seed
Dis-incentive for Malicious Behavior	**Safety deposit** **Plenitude rule**
Incentive for Honest Behavior	Block reward Accussing reward

Disincentive = Deposit + Endorsement

Endorsement

블록당 **32명**의 Endorser가 **미리** 정해짐 (Roll snapshot + Random seed)

이전 블록에 **투표**(endorse, notarize)

$\mathcal{B}lock_i$ 가 베이킹 된 후, 인도서들은 $\mathcal{B}lock_i$ 에 대해 서명을 제출함

$\mathcal{B}lock_i$ 베이커가 이 투표를 블록에 포함

체인의 score(fitness) = 인도스먼트의 합

인도싱이 블록에 담기고, Canonical chain이 되어야 보상

보상 = 최대 2 ꜩ (2 / dT)

Disincentive = Deposit + Endorsement

Safety Deposit

베이킹, 인도싱에 요구되는 조건

5사이클 동안 동결(locked, frozen)

악의적인 행동 시 몰수(forfeit, slashed)

Baking	Number	Deposits	Rewards
Baking	1	**512 ꜩ**	16 ꜩ
Endorsing	Up to 32	**64 ꜩ**	2/dT ꜩ

PoS with Optional Delegation

Chain-based PoS consensus with a use of endorsements to speed-up confirmation times and reduce selfish baking

PoS of Tezos, **A mix of several ideas**

Validator Selection Rule	Multiple snapshots Random seed
Dis-incentive for Malicious Behavior	Safety deposit Plenitude rule
Incentive for Honest Behavior	**Block reward Accussing reward**

Incentive

Baking reward

Steal: 후순위 베이커가 베이킹 하는 경우

Endorsing reward

최대 2 ꜩ (2 / dT)

Accusing reward

악의적인 행동을 신고

악의적인 행동이 적발된 베이커는 Safety deposit 몰수

몰수된 토큰의 절반은 소각(proof of burn), 절반은 신고자에게 지급

합의 알고리즘 심화

합의, 분산 시스템의 목적

Consensus protocol

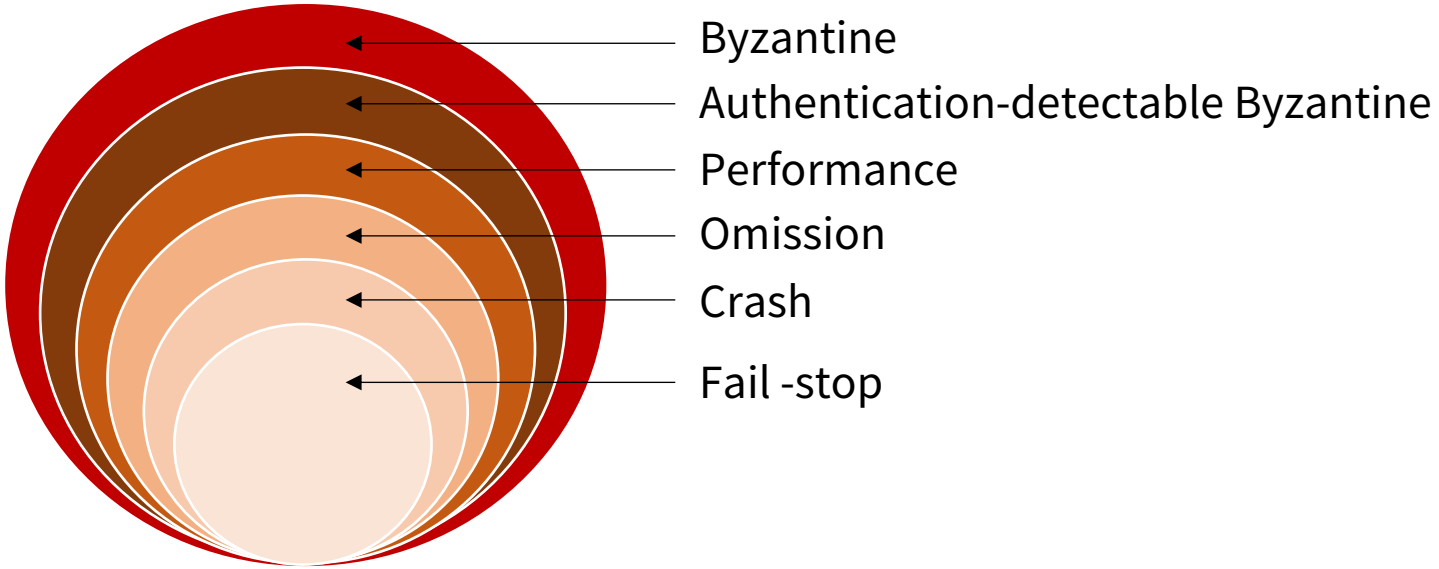
The mechanism where consensus is built around the a single chain

Byzantine fault tolerance

The characteristic of distributed systems which tolerates failures called Byzantine faults.

The Byzantine Generals Problem (Lamport, 1982)

No solution with fewer than 3m + 1 generals can cope with m traitors. (OM Algorithm)



합의, 분산 시스템의 목적

Safety vs. Liveness*

A **safety** property is one which states that **something will not happen**.

- A program with the correct input cannot stop if it does not produce the correct output.

A **liveness** property is one which states that **something must happen**.

- A program will **terminate** if its input is correct.

* Proving the Correctness of Multiprocess Programs (Lamport, 1977)

합의, 분산 시스템의 목적

FLP Result*

No completely **asynchronous** consensus protocol can tolerate even **a single** unannounced process death (**crash failure**)

- both **termination** and **agreement** cannot be satisfied.

* Impossibility of Distributed Consensus with One Faulty Process (Fischer, Lynch & Patterson, 1985)

합의, 분산 시스템의 목적

Paxos protocols*

Safety is guaranteed in a **closed network** with a **quorum rule** and a multi-phase learning process.

* The Part-Time Parliament (Lamport, 1989)

합의, 분산 시스템의 목적

Practical BFT

Both safety and liveness are guaranteed but use an assumption of **partial synchrony**.

합의, 분산 시스템의 목적

Safety vs. Liveness
FLP Result
Paxos protocols
Pratical BFT

Safety vs. Liveness

BFT-based Consensus

Classical

Safety over liveness

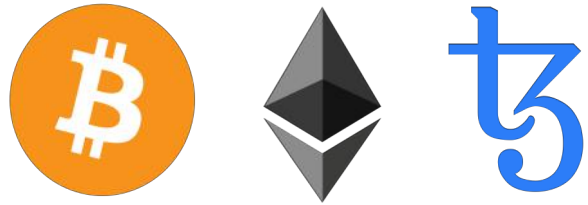
Deterministic finality

Nakamoto Consensus

Chain-based

Liveness over safety

Probabilistic finality



블록체인 거버넌스

What is governance?

Governance

All processes of governing

Interaction, Decision-making, Enforcement, ...

조직 또는 집단의 지배구조

정치적 이해관계자들의 다원적, 협력적 통치 방식 (정치학)

다양한 행위자들이 문제를 해결하는 새로운 국정운영방식 (행정학)

What is governance?

Blockchain network: 하나의 독립된 사회 또는 조직

Blockchain protocol: 네트워크에 참여한 노드들이 연결되는 방식

Blockchain protocol is a form of governance

What is governance?

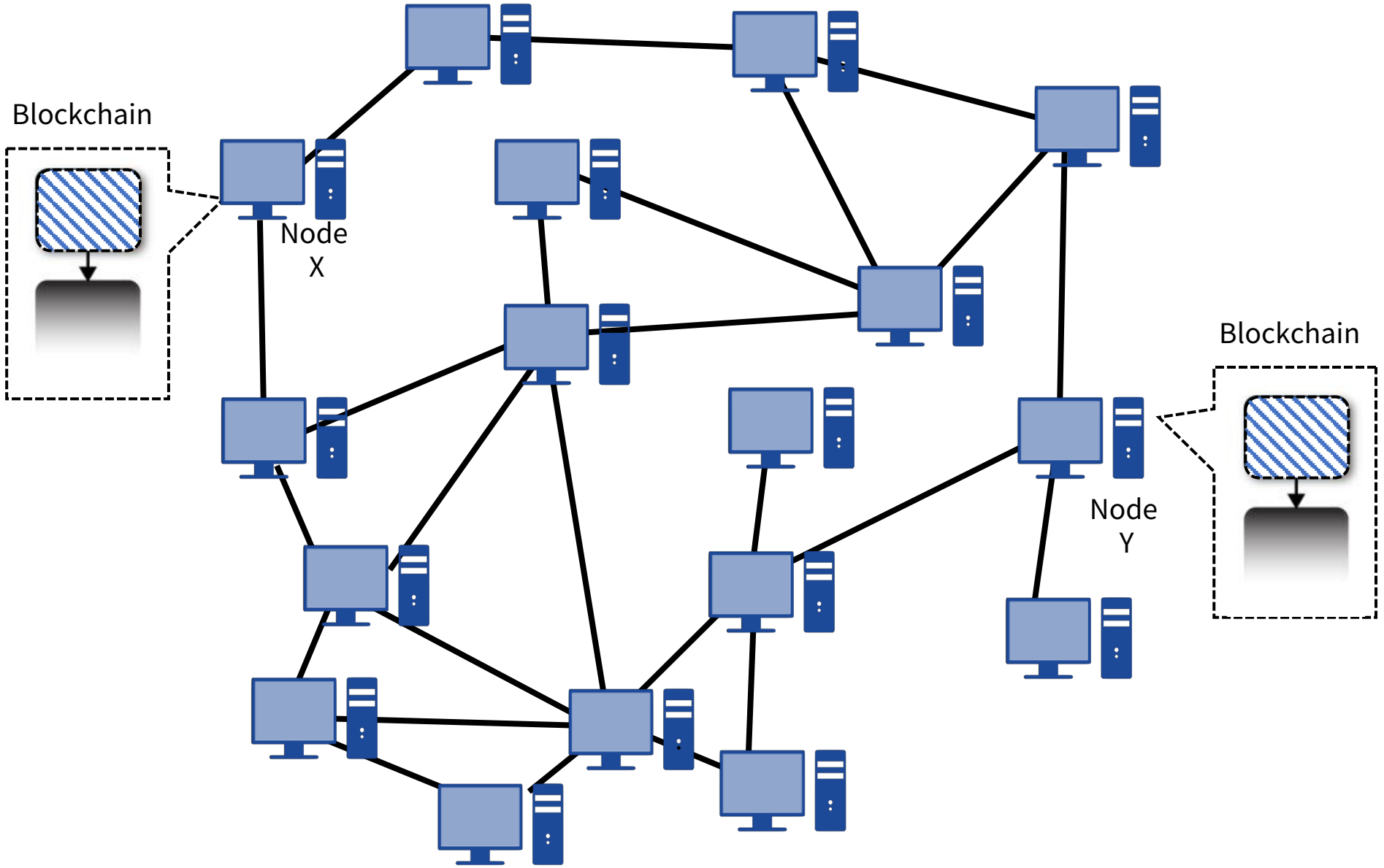
Blockchain Governance

합의 알고리즘을 포함하여 네트워크에 참여한 노드들이 공유하는 규칙의 집합체

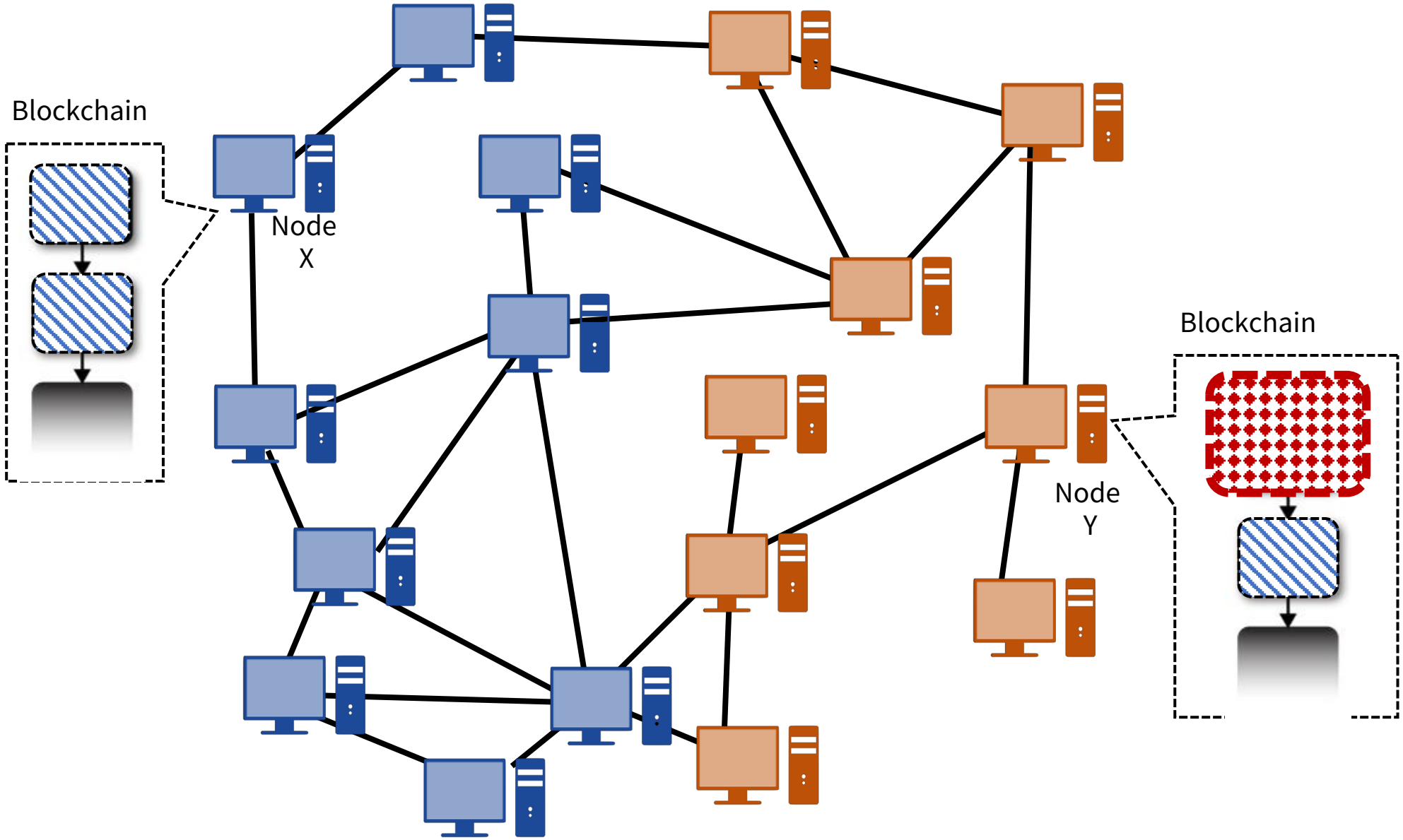
블록체인 전체 생태계에서 이루어진 모든 의사결정 방식의 체계

새로운 규칙의 생성, 변경, 폐기를 포함

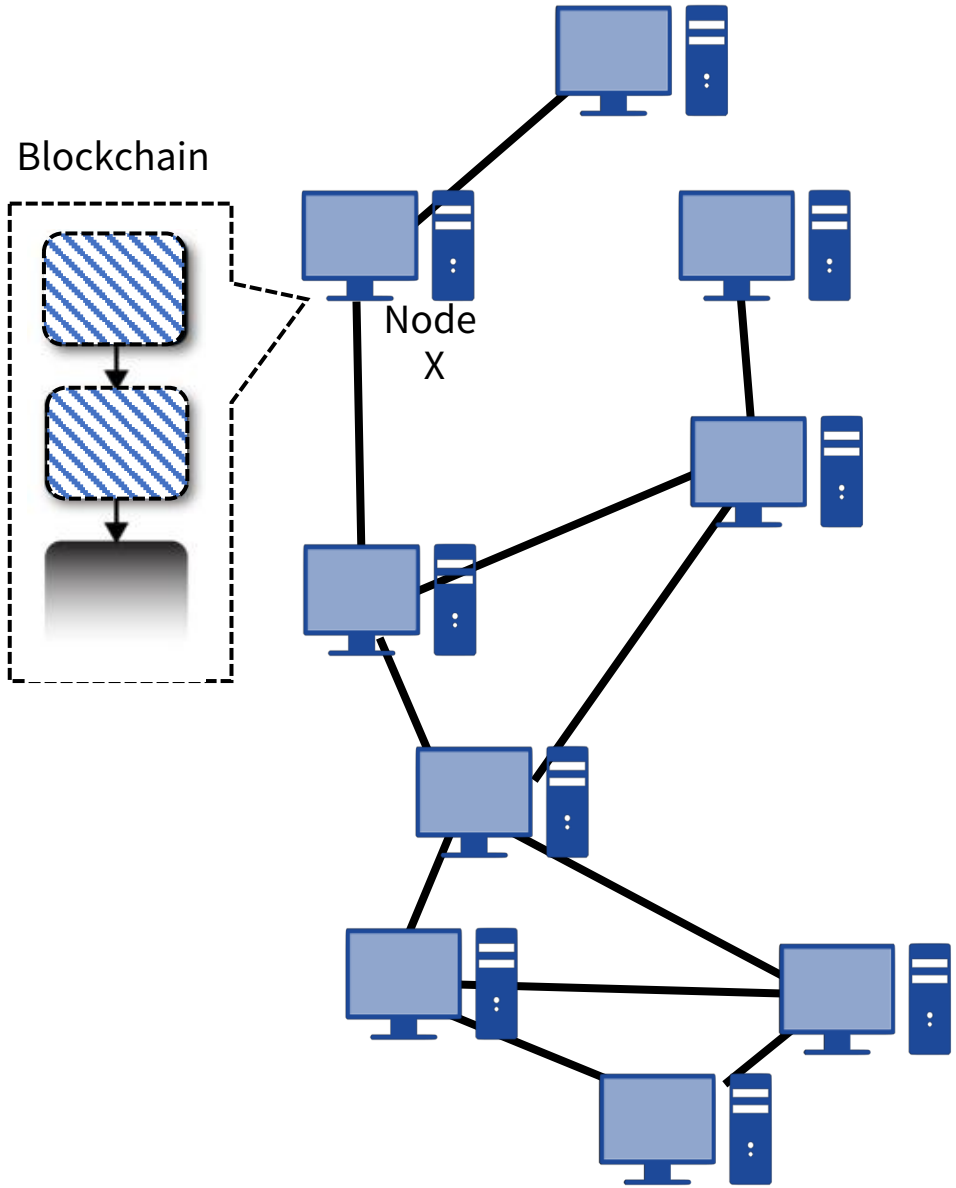
프로토콜의 변경

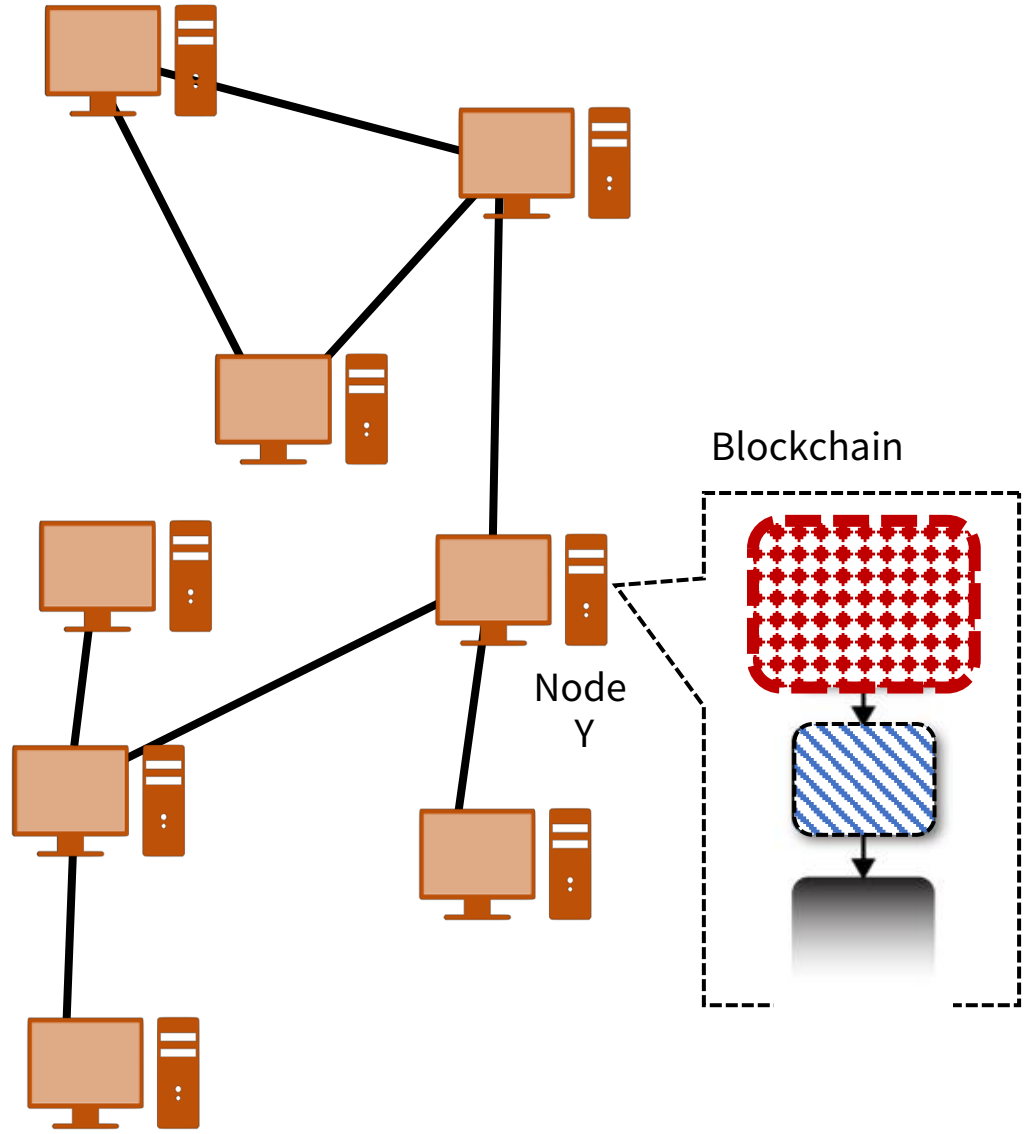


프로토콜의 변경



프로토콜의 변경





What is governance?

Procedure & Coordination

Proposal (Who propose, What propose, **Incentives**)

Discussion (**Information** gap)

Voting (1 head – 1 voting, staked-based voting, **Incentives**)

Operator’s favor vs. User’s favor

Updating the software at the same time

Migration cost

What is governance?

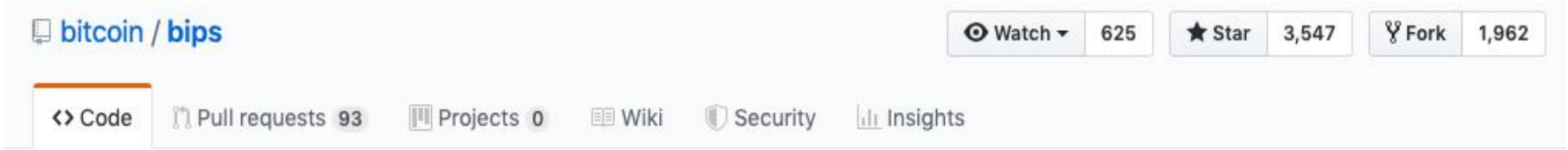
Off-chain vs. On-chain

블록체인 거버넌스가 어디에서 이루어지고 있는가

기존 방식(Off-chain)보다 더 나은 방식은 없을까

다른 건 기록하면서 왜 거버넌스는 기록하지 않지

Off-chain governance: Bitcoin Improvement Proposal (ref #10)

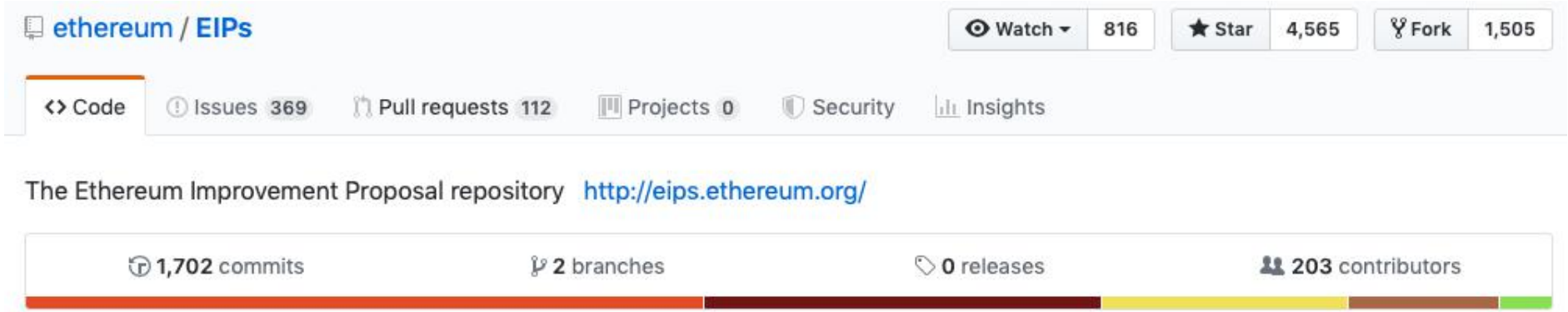


Bitcoin Improvement Proposals http://bitcoin.org

1,888 commits | 5 branches | 0 releases | 196 contributors

Number	Layer	Title	Owner	Type	Status
1		BIP Purpose and Guidelines	Amir Taaki	Process	Replaced
2		BIP process, revised	Luke Dashjr	Process	Active
32	Applications	Hierarchical Deterministic Wallets	Pieter Wuille	Informational	Final
38	Applications	Passphrase-protected private key	Mike Caldwell, Aaron Voisine	Standard	Draft
39	Applications	Mnemonic code for generating deterministic keys	Marek Palatinus, Pavol Rusnak, Aaron Voisine, Sean Bowe	Standard	Proposed

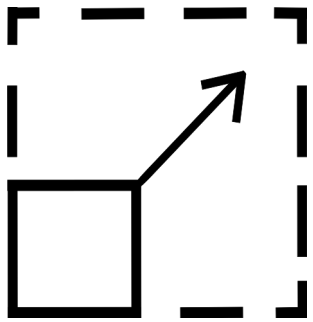
Off-chain governance: Ethereum Improvement Proposal (ref #11)

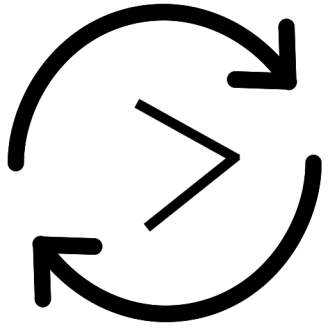


Number	Title	Author
2	Homestead Hard-fork Changes	Vitalik Buterin
7	DELEGATECALL	Vitalik Buterin
20	ERC-20 Token Standard	Fabian Vogelsteller, Vitalik Buterin
137	Ethereum Domain Name Service - Specification	Nick Johnson
140	REVERT instruction	Alex Beregszaszi, Nikolai Mushegian
141	Designated invalid EVM instruction	Alex Beregszaszi

Challenges in blockchain (feat 2.0)

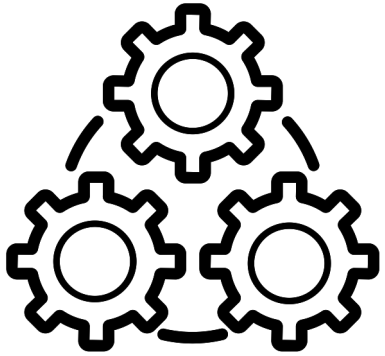
Scalability
Storage
Process time
Validation time
Propagation time
# of miners





Sustainability
Resolving conflicts
Adopting new innovations

Interoperability
Communication between
different protocol



Security
Core security
Smart contract security

바보야, 문제는 **거버넌스**야

Sustainability

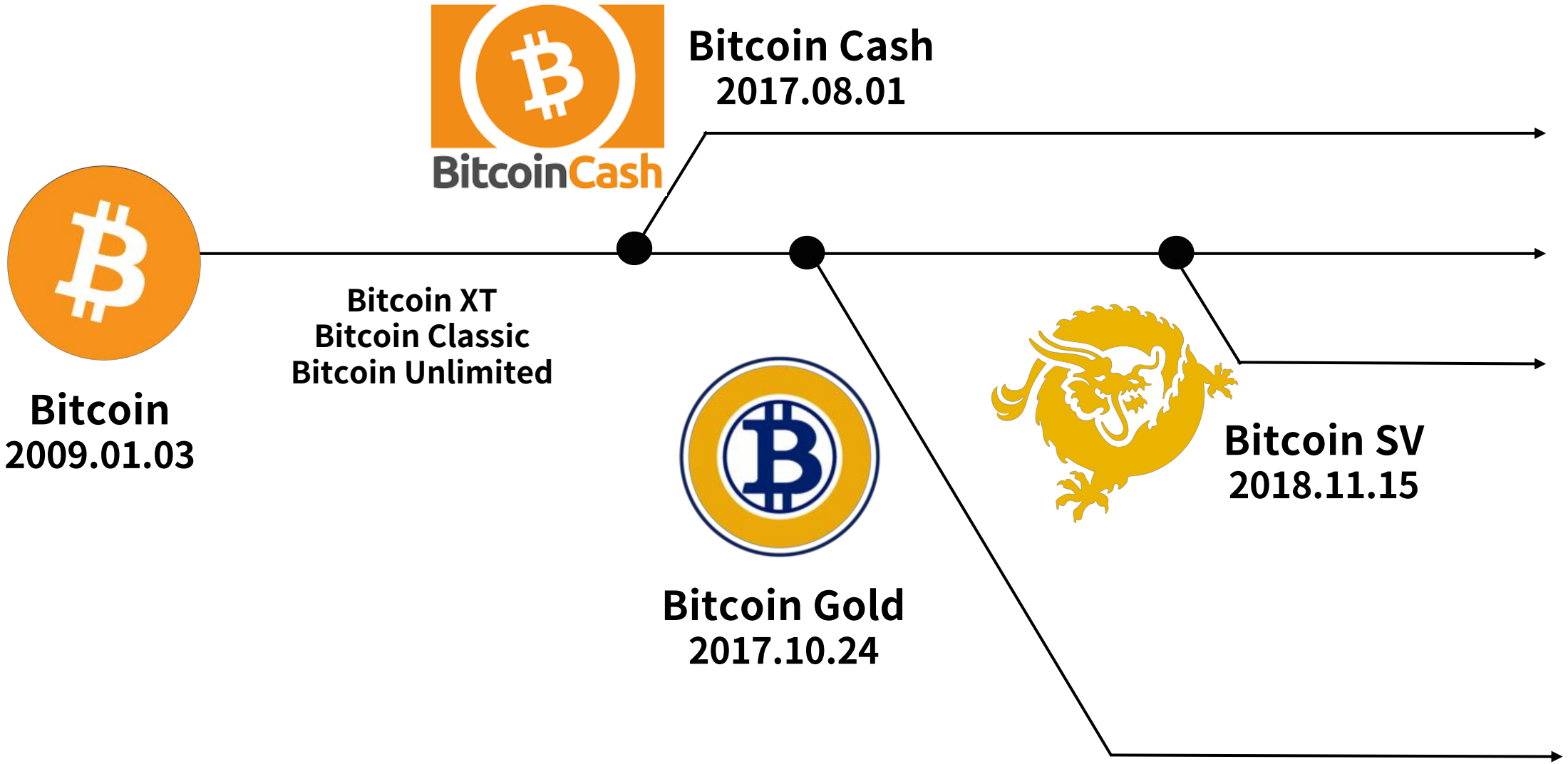
어떻게 (프로토콜의) 갈등을 해결할 것인가

새로운 혁신들을 어떻게 받아들일 것인가

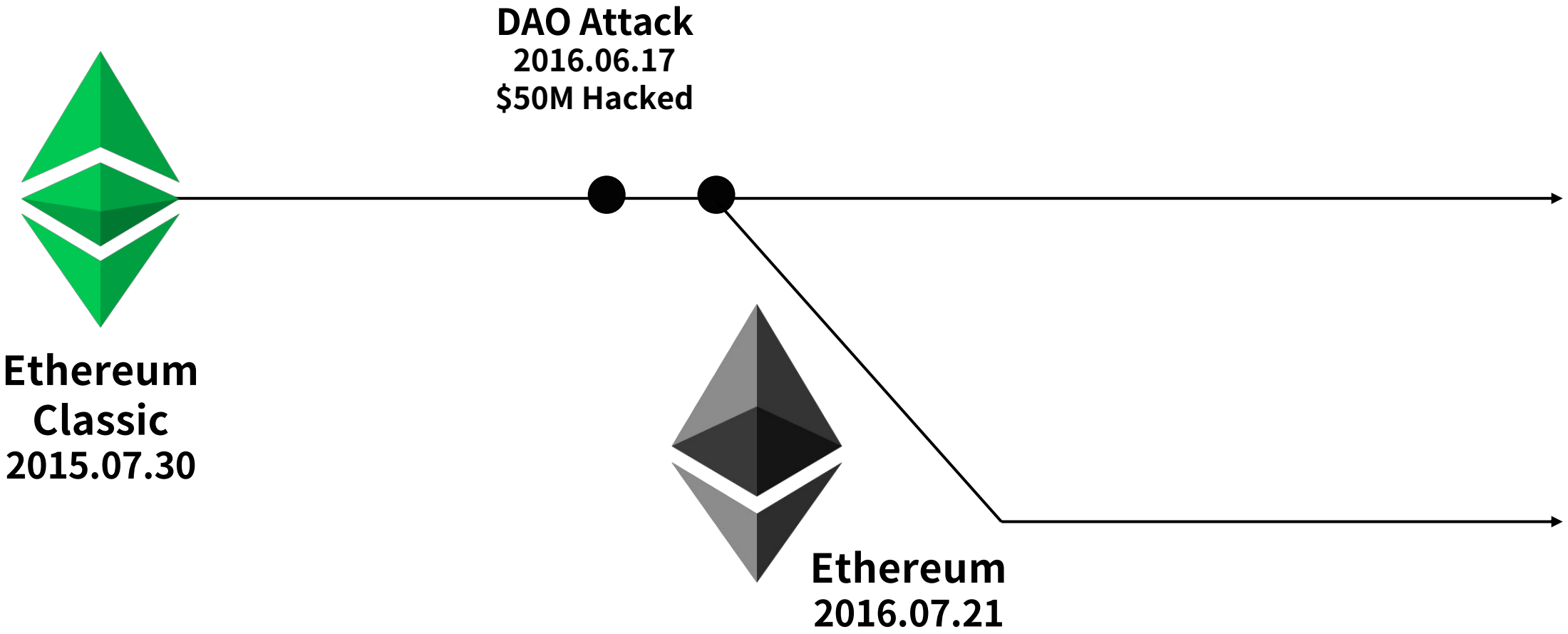
어떤 혁신을 받아들일 것인가

재단(Foundation)과 개발팀(Dev team)에 의존하는 게 맞는가

Hard fork case: Bitcoin



Hard fork case: Ethereum



Hard fork case: Ethereum

Name	On Roadmap	Date	Block
Frontier	Yes	2015.07.31	1
Frontier Thawing	Yes	2015.09.08	200000
Homstead	Yes	2016.03.15	1150000
DAO Fork	**No**	2016.07.21	1920000
EIP-150 Hard Fork	**No**	2016.10.19	2463000
Spurious Dragon	**No**	2016.11.23	2675000
Byzantium	Yes	2017.10.16	4370000

Off-chain governance

Old school blockchains (Bitcoin, Ethereum, forks of them)

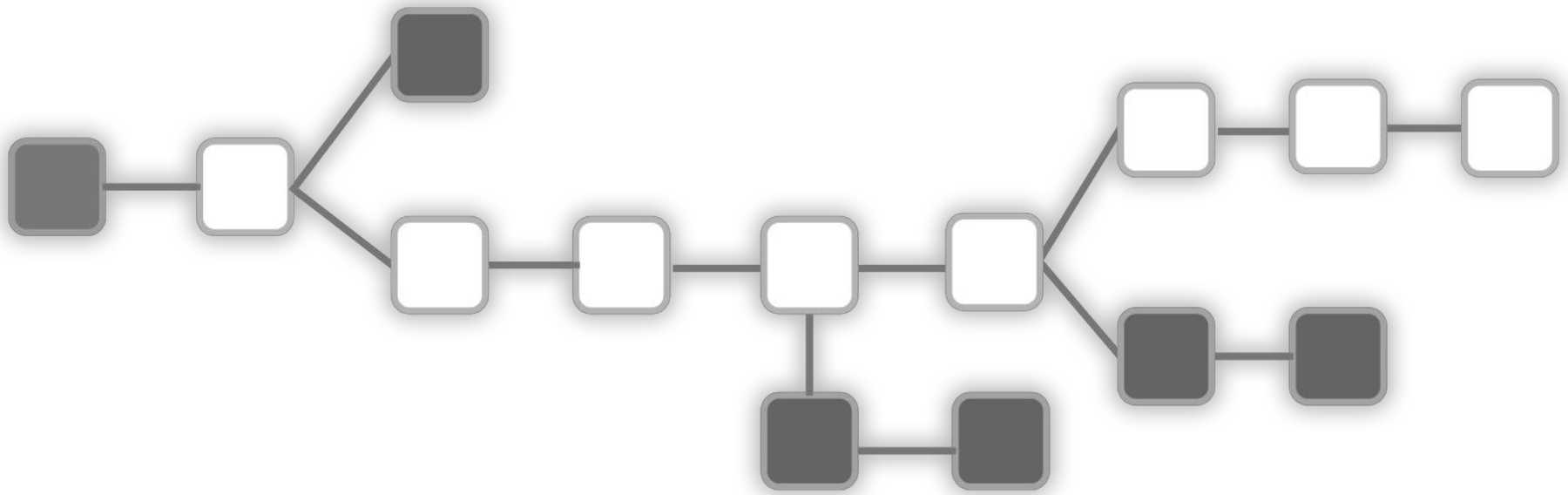
Power balancing between core developers, miners, users, business

Online forum, Improvement proposals(BIP, EIP)

(better than legacy but) **Centralized** (public lack the technical knowledge)

Opt-in (flexibility not to choose, easily fork by open-source)

Abstract Blockchain

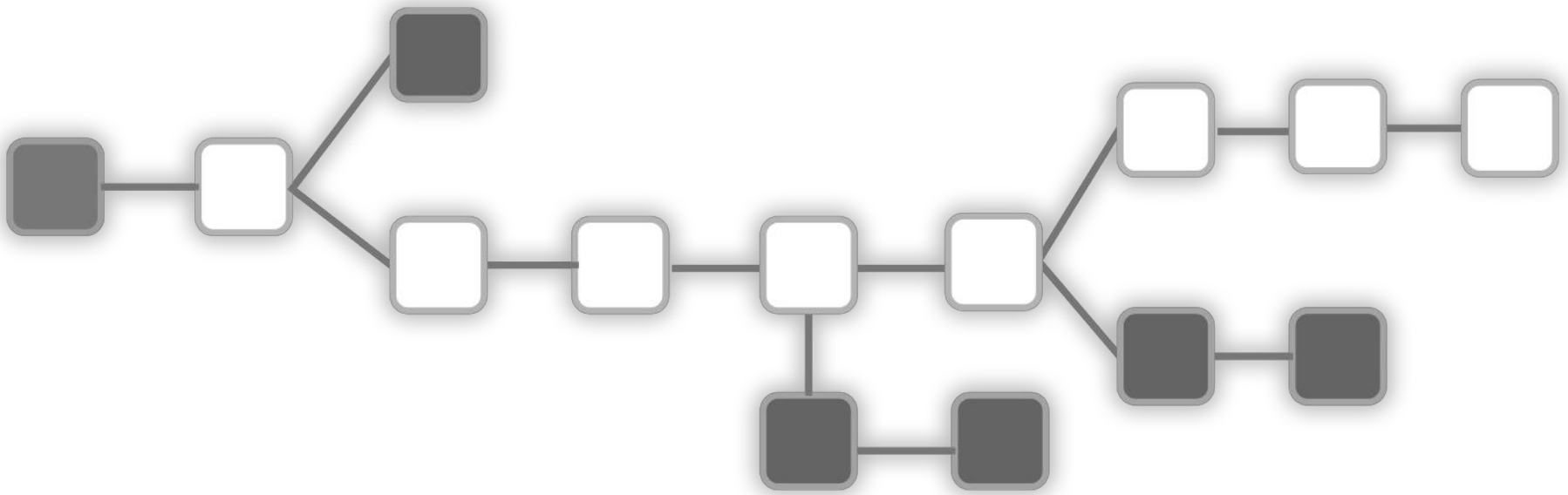


All existing block chain implementation generally

apply: $S \mapsto O \mapsto S$ or $S' = f(S, O)$

score: $S \mapsto N$ or $N = f(S)$

Abstract Blockchain



Tezos's most powerful features

set_test_protocol: replacing the protocol used in the test chain with a new protocol

promote_test_protocol: replacing the current protocol with the protocol being tested

On-chain governance case: **Tezos**

ZD Net Korea

테조스, 하드포크 없이 첫 번째 업데이트 성공

기사입력 2019.05.30. 오후 2:40 기사원문 스크랩 본문듣기 · 설정

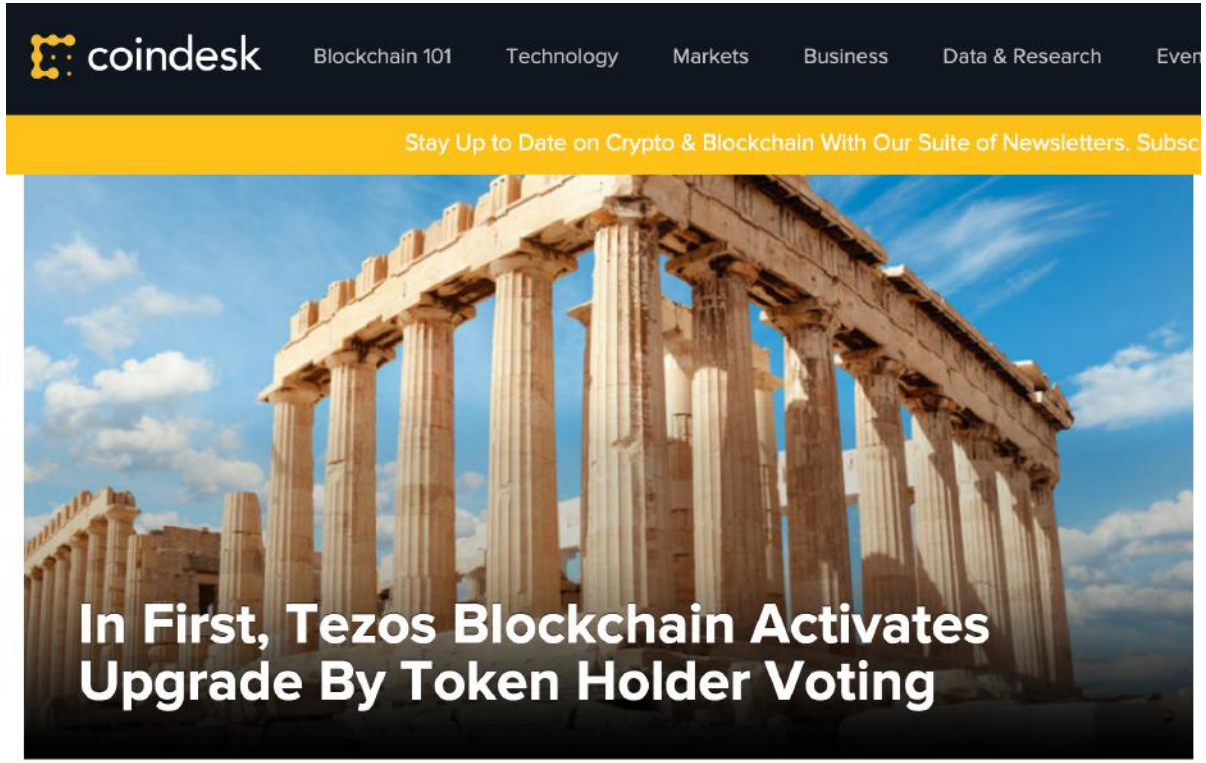
5 2 요약봇 Beta 가

"업데이트로 연산 능력 두배 증가"

(지디넷코리아=임유경 기자)블록체인 플랫폼 테조스가 온체인 거버넌스 투표를 통해 하드포크 없이 첫 번째 업그레이드를 완료했다.

테조스코리아(대표 이진우)는 테조스 블록체인에 첫번째 프로토콜 업데이트 제안이 올라온 후, 약 3개월만에 총 4단계 절차를 거쳐서 업데이트를 완료했다고 28일 밝혔다.

이번 업데이트 제안은 마지막 절차인 '적용 단계'에서 투표 참여율 84.35%에 찬성율 99.89%(Pass 제외)로 통과됐다. 458753번째 블록부터 업데이트가 적용됐다.



On-chain governance case: **Tezos**

Athens Proposal

2019년 3월 ~ 5월

참여율: 84.35% (약 4.7억개 토큰)

찬성률: 99.89%

Stake unit (10,000 ꜩ → 8,000 ꜩ)

Gas limit (x2 ↑)

Tezos is **formalizing** blockchain governance

Protocol amendments

프로토콜 수정, 추가, 제안

In-protocol voting procedures

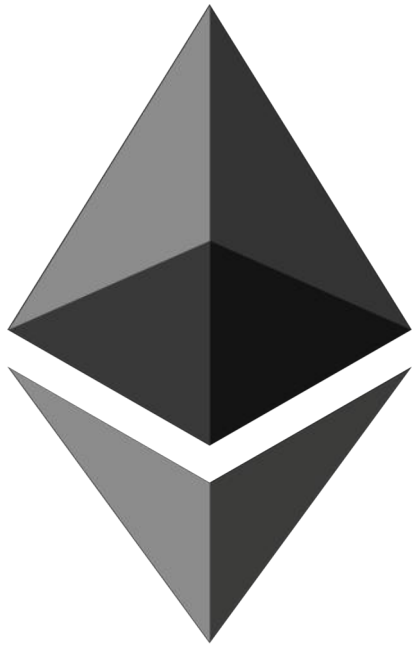
Rules about changing the rules

거버넌스의 보조 도구로서 블록체인(crypto-ledger) 사용

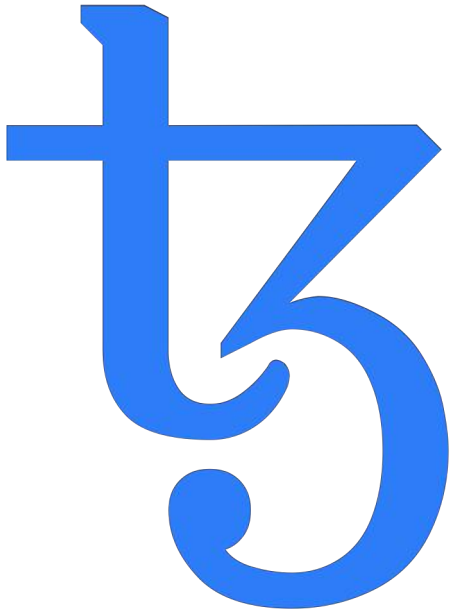
프로토콜 제안과 투표는 **primitive operations**

Bakers vote

Tezos is **formalizing** blockchain governance

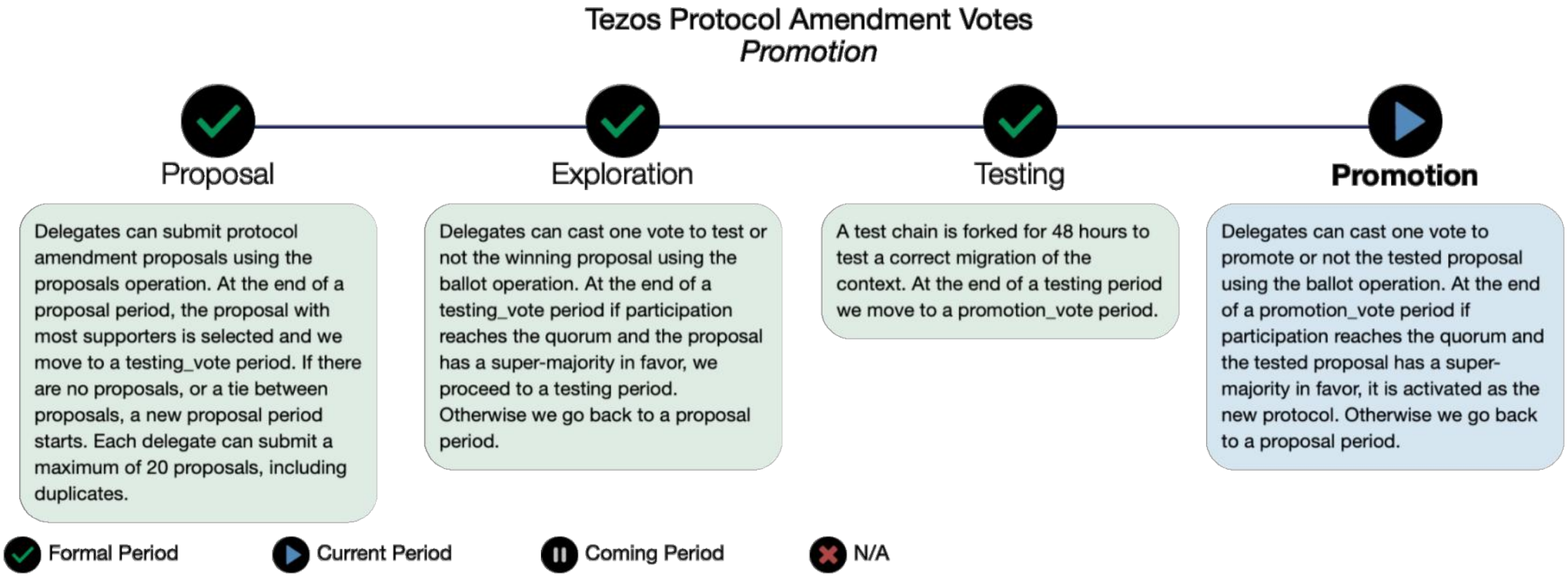


Thin protocol



Fat protocol

Tezos is **formalizing** blockchain governance



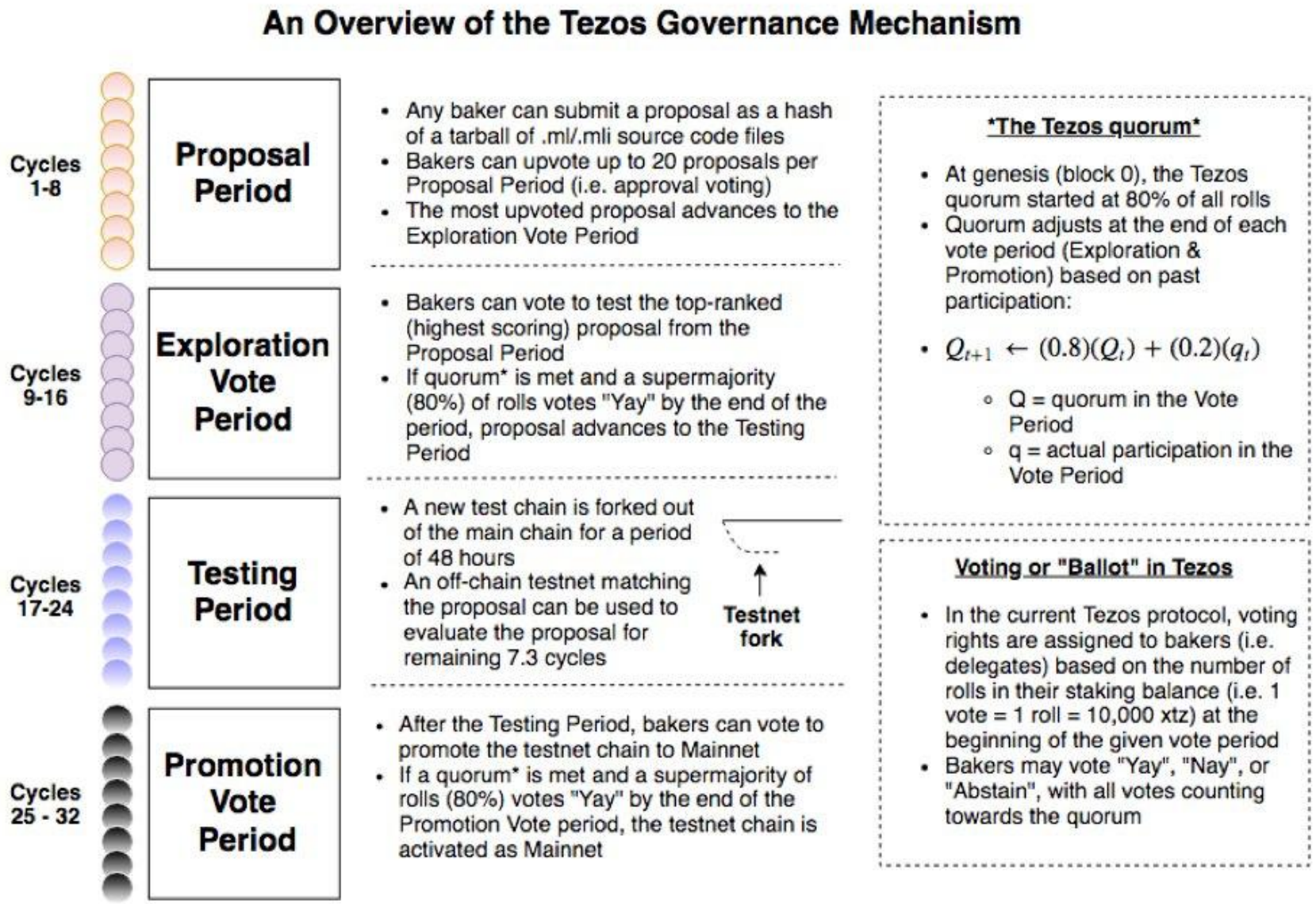

```
[ubuntu@ip-172-31-4-183:~$ tezos-client submit ballot for tezoskorea ]
Pt24m4xiPbLDhVgVfABUjirbmda3yohdN82Sp9FeuAXJ4eV9otd yay
```

Tezos is **formalizing** blockchain governance

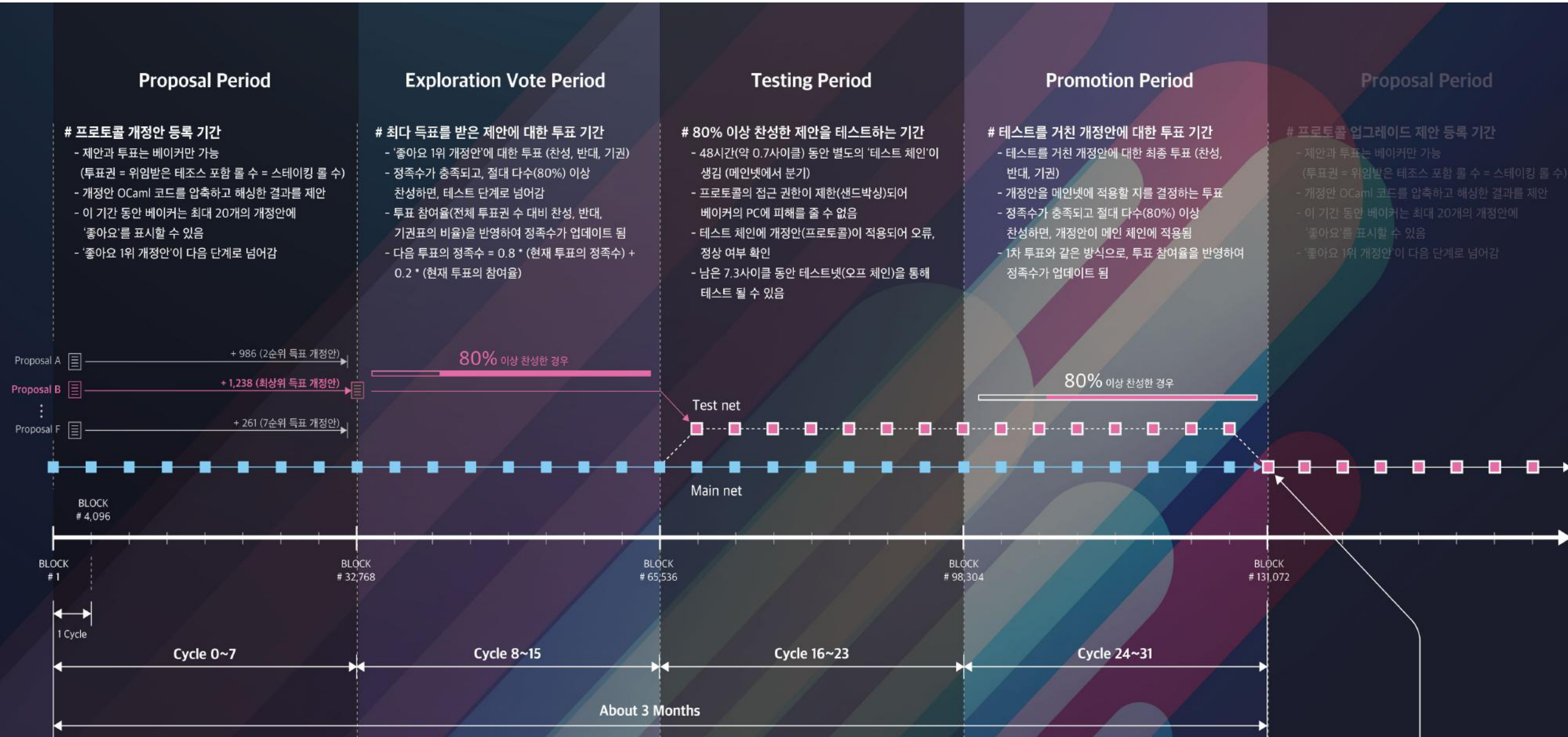
```
{
    protocol: "PsddFKi32cMJ2qPjf43Qv5GDWLDPZb3T3bF6
    next_protocol: "Pt24m4xiPbLDhVgVfABUjirbmda3yoh
  - test_chain_status: {
        status: "not_running"
    },
    max_operations_ttl: 60,
    max_operation_data_length: 16384,
    max_block_header_length: 238,
  + max_operation_list_length: […],
    baker: "tz3NExpXn9aPNZPorRE4SdjJ2RGrfbJgMAaV",
  - level: {
        level: 458752,
        level_position: 458751,
        cycle: 111,
        cycle_position: 4095,
        voting_period: 13,
        voting_period_position: 32767,
        expected_commitment: true
    },
```

```
{
    protocol: "Pt24m4xiPbLDhVgVfABUjirbmda3yohdN82Sj
    next_protocol: "Pt24m4xiPbLDhVgVfABUjirbmda3yohd
  - test_chain_status: {
        status: "not_running"
    },
    max_operations_ttl: 60,
    max_operation_data_length: 16384,
    max_block_header_length: 238,
  + max_operation_list_length: […],
    baker: "tz1NortRftucvAkD1J58L32EhSVrQEWJCEnB",
  - level: {
        level: 458753,
        level_position: 458752,
        cycle: 112,
        cycle_position: 0,
        voting_period: 14,
        voting_period_position: 0,
        expected_commitment: false
    },
```

Tezos is **formalizing** blockchain governance



Tezos is **formalizing** blockchain governance



In Tezos, **stakeholders govern** the protocol

Decentralized development

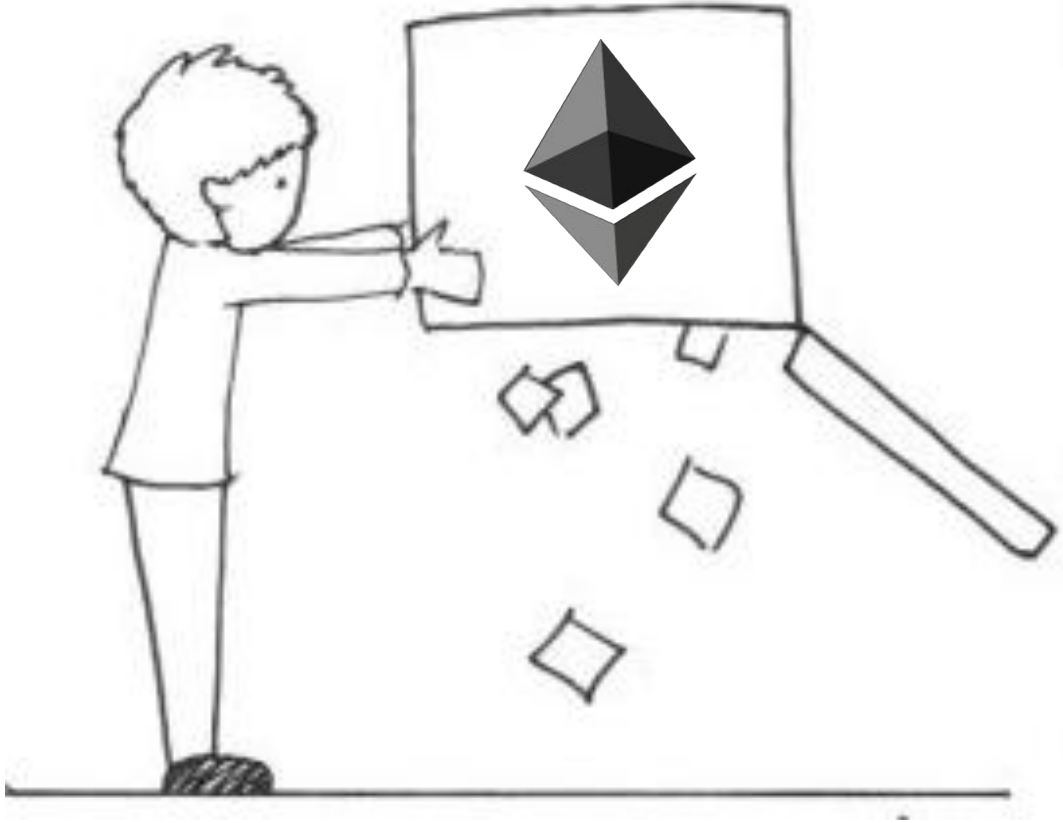
Anyone can propose an amendment

No roadmap

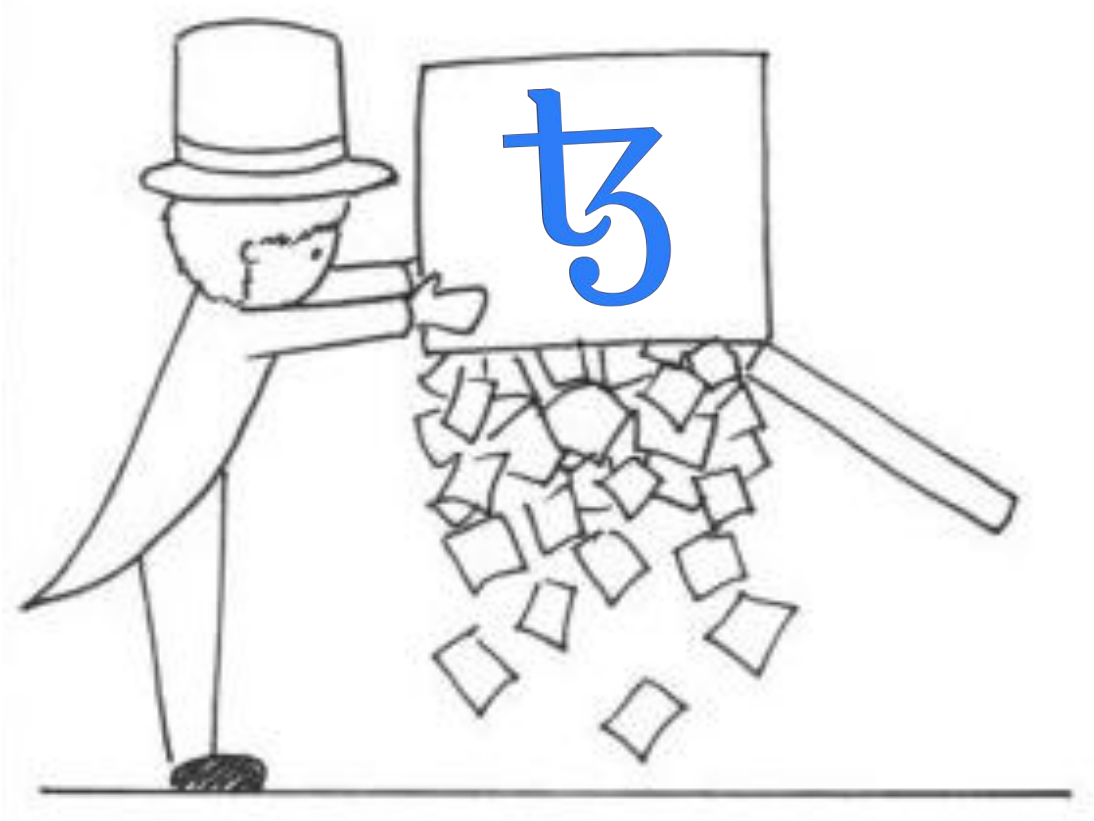
Amendments to the voting procedure itself

Funding by attaching an **invoice**

In Tezos, **stakeholders govern** the protocol



DAO Carbon Vote
4.5%



Proposal Athenes
84.35% (47,049 rolls)

Developments in parallel

Decentralized innovations

Ergo compiler for legal contract (L2, Rust binding)

Zk-Snark (L1, Rust binding)

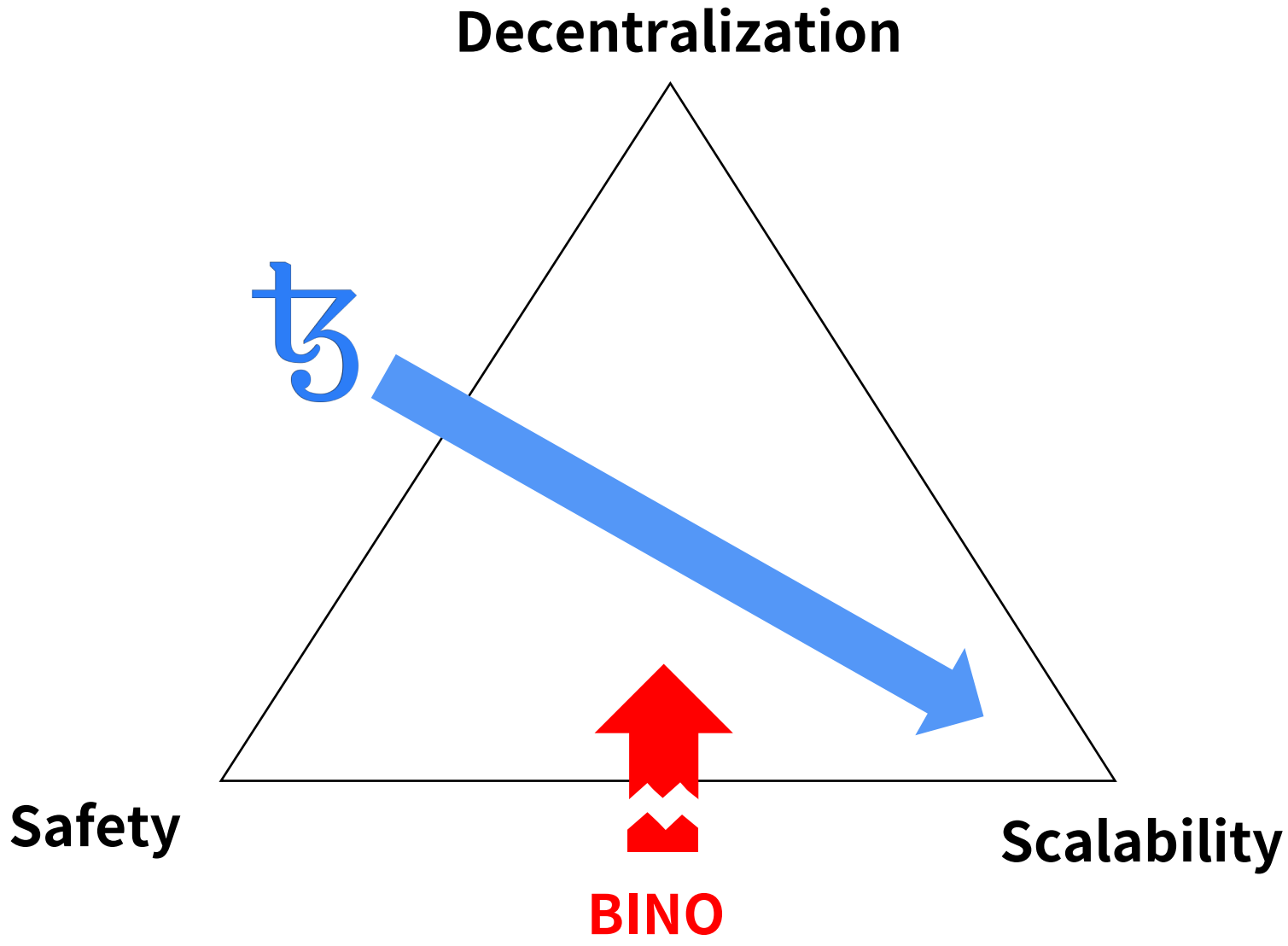
Tendermint as finality gadget (L1, Nomadic Labs, Cryptium Labs)

Sharding (L1, Emin)

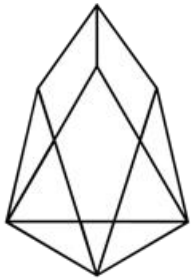
(Plasma-like) **Marigold** (L2, Nomadic Labs)

Somewhere we don’t know

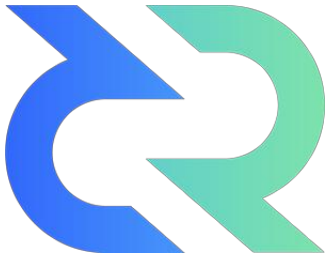
Blockchain trillemma



On-chain governance models

Polkadot. End of 2019	- Stake-weighted referenda with adaptive super-majority thresholds - Proposing, voting, tallying - Locking for increasing voting power - Voting for council (veto, 1 year term) - Switching out the entire code of the runtime
	- Governed blockchain (republicanism) - Amendable Constitution and ECAF (EOS Core Arbitration Forum) - 21 validators (Centralization, validator collisions) - Fork based protocol upgrade

On-chain governance models

 Late Q2 of 2019	- Blockchain Nervous System (Neuron with deposit, Proposals, Evaluation) - Economics, Policy, Protocol, Client, even Change rewrite - Staking reward scaled by voting participant rate - Voting by itself or following (relationships are hidden)
	- Time-lock DCR for staking with a mandatory voting - Treasury - Politeia (a web platform) - Consensus rule (DCP, new SW, upgrade, voting, activation) - Constitution

On-chain governance

Decision making on chain (ex. representative/liquid democracy, futarchy)

Leaderless governance (why nakamoto remain anonymous)

In the very **early stage**

Bootstrapping problem (it takes a very long time to develop)

Why NOT On-chain

It’s coercive (update should be Opt-in)

The danger of incentives and misaligned interests

(coin holders ≠ users of the protocol)

The dangers of direct democracy (Harvard, 2019)

Low-voter turnout (ex. DAO Carbonvote 4.5%)

Plutocracy (staked-based voting)

Why On-chain

Opt-out (reduced hard fork risk & migration cost, it’s a social consensus)

Aligned better than miner-based governance (Primary use case = money)

(coin holders ≈ users of the protocol)

More decentralized and democratic than foundation-led governance

Governance mechanism itself can be updated as well (quadratic voting)

Off-chain vs. On-chain

Why NOT On-chain

It’s coercive (update should be Opt-in)
The danger of incentives and misaligned interests
(coin holders ≠ users of the protocol)
The dangers of direct democracy (Harvard, 2019)
Low-voter turnout (ex. DAO Carbonvote 4.5%)
Plutocracy (staked-based voting)

Why On-chain

Opt-out (reduced hard fork risk & migration cost, it’s a social consensus)
Aligned better than miner-based governance (Primary use case = money)
(coin holders ≈ users of the protocol)
More decentralized and democratic than foundation-led governance
Governance mechanism itself can be updated as well (quadratic voting)