FTEC 4004

E-payment Systems and Cryptocurrency Technologies

Tutorial 10

Bitcoin Data Structure and Mining

WANG Xianbo

xianbo@ie.cuhk.edu.hk

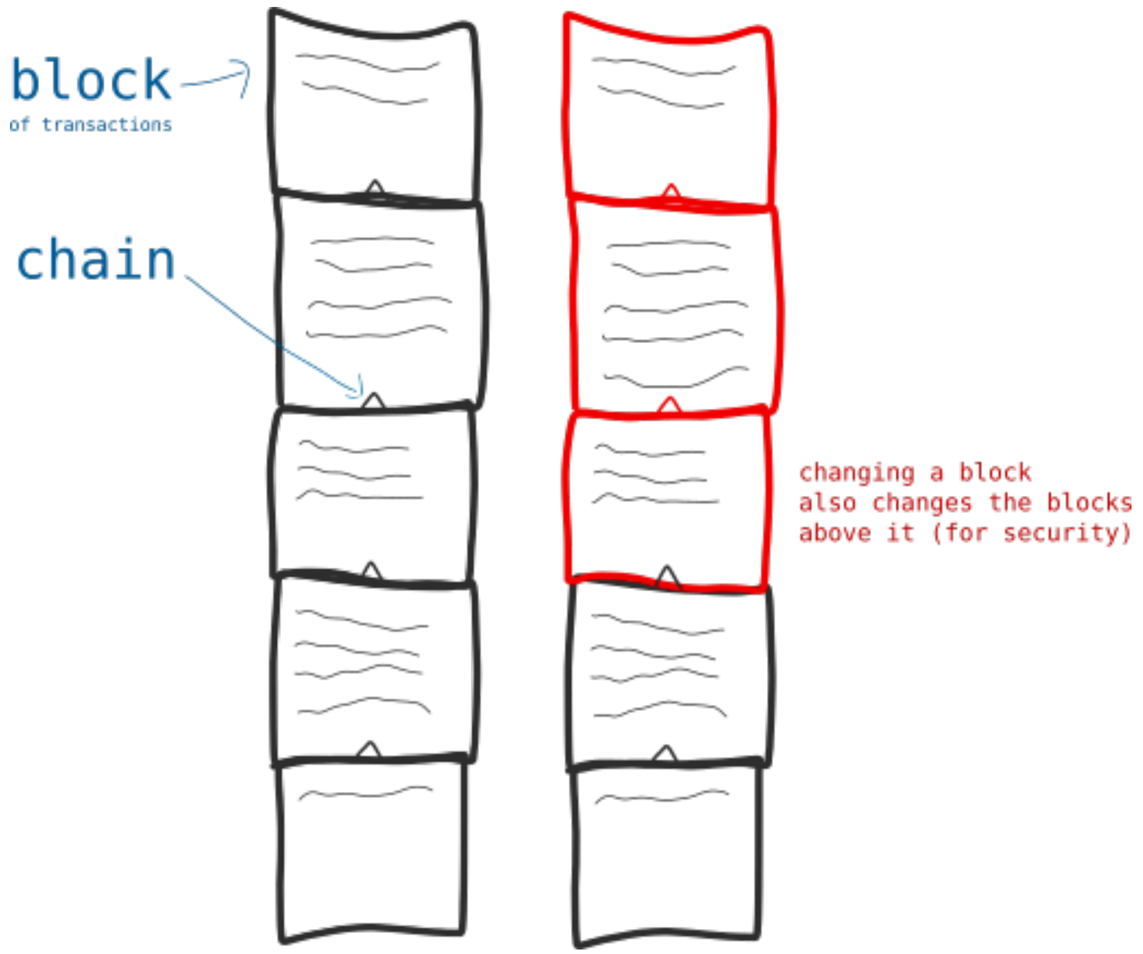
Overview of today’s topic

- Data Structure of Bitcoin Blockchain
- Mining mechanism of Bitcoin
- Q&A time for HW4

Please complete the course evaluation if you haven’t

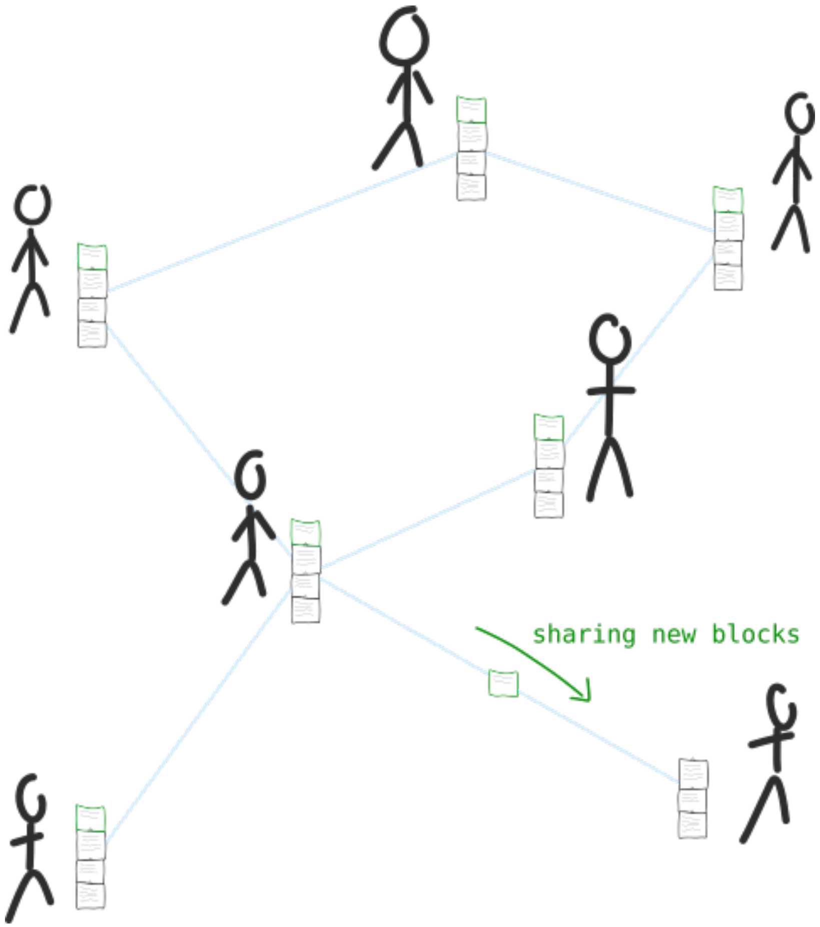
Blockchain: Chain of Blocks

- **Blockchain**: a chain of blocks, each containing a batch of transactions
- **Block**: make it easier to share transactions over the Internet
- **Chain**: make it difficult to tamper with transaction records



Blockchain Network: Shared Public Ledger

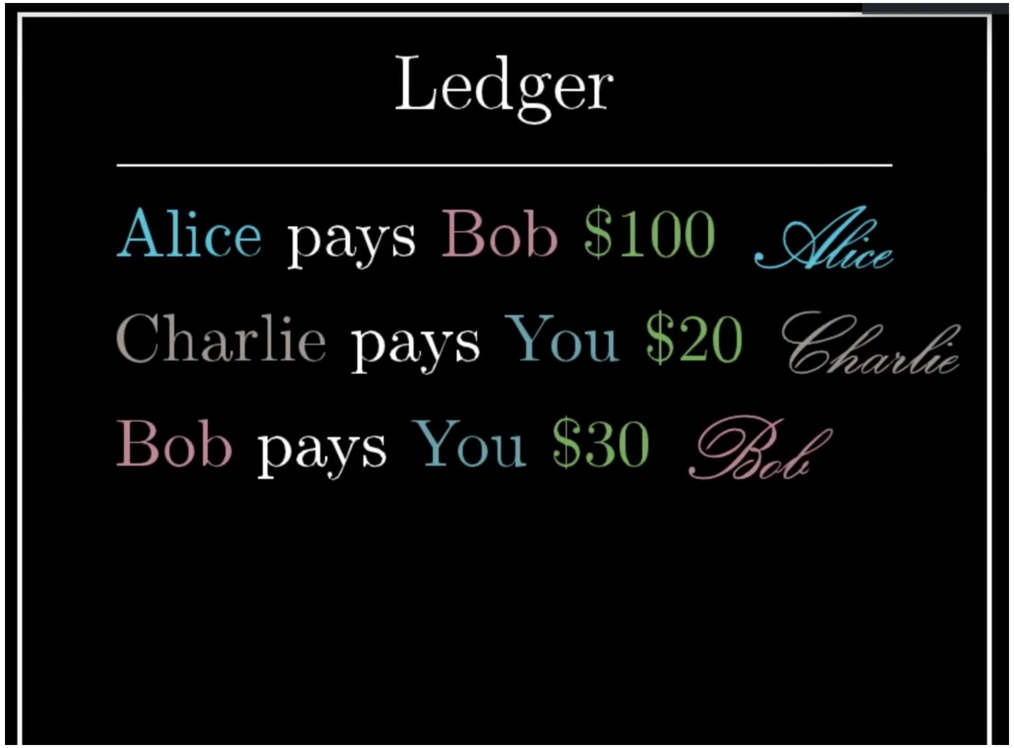
- Everyone runs a copy of the blockchain.
- When new transactions are added, nodes in the network should be updated correspondingly.
- Data can be altered, need a mechanism to reach consensus in the network.



Transaction

Digital Signature

- Everyone can add transaction records to the public ledger, how to prevent forgery? Use signature!
- Digital Signature
    - Sign: Sender sign the transaction with his private Key
    - Verify: Everyone can use sender's public key to verify the transaction
    - Wallet address = hash(public key)



Ledger

Alice pays Bob $100 Alice

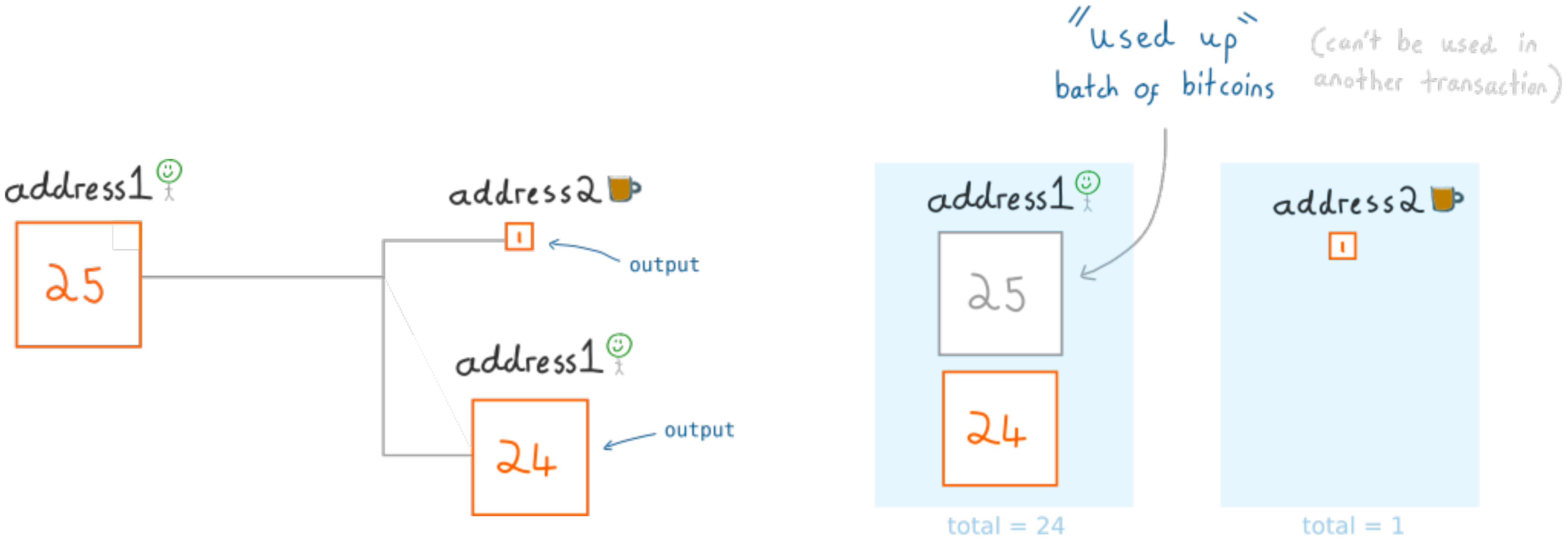
Charlie pays You $20 Charlie

Bob pays You $30 Bob

Transaction

UTXO (Unspent Transaction Output) model

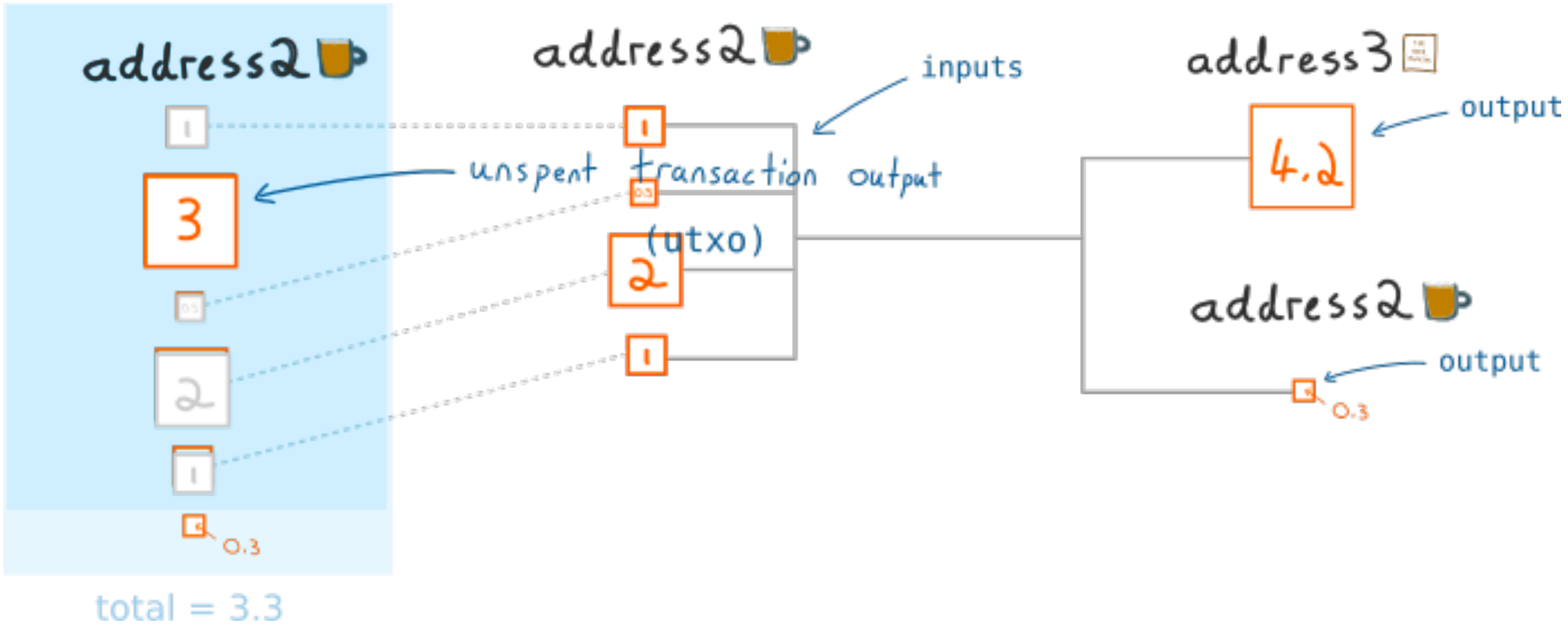
1. Buy a cup of coffee (1 BTC)



Transaction

UTXO (Unspent Transaction Output) model

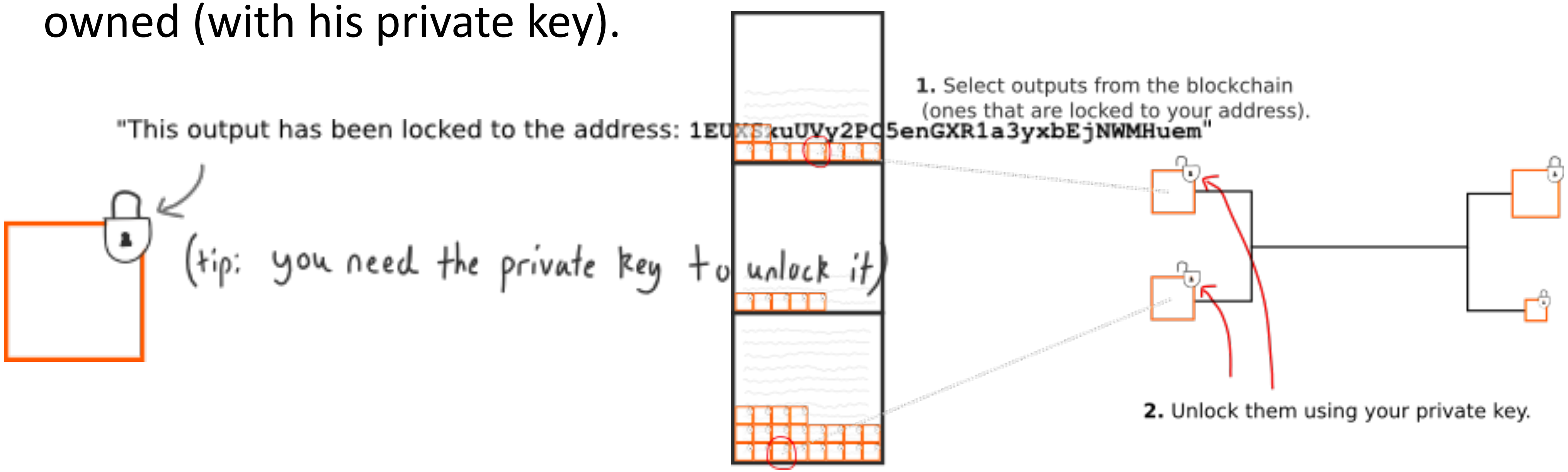
1. Buy a cup of coffee (1 BTC)
2. Coffee shop cost 4.2 BTC to renew the coffee machine



Transaction

Nasty implementation details: Lock & Unlock

- Bitcoin uses output lock to implement the sign/verify function.
  - Outputs are created with **locks** that are bound to **public keys** (receiver's addresses) and can only be **unlocked** (prove ownership) with corresponding (receiver's) **private keys**.
  - To create new outputs, the sender first needs to unlock the some outputs he owned (with his private key).



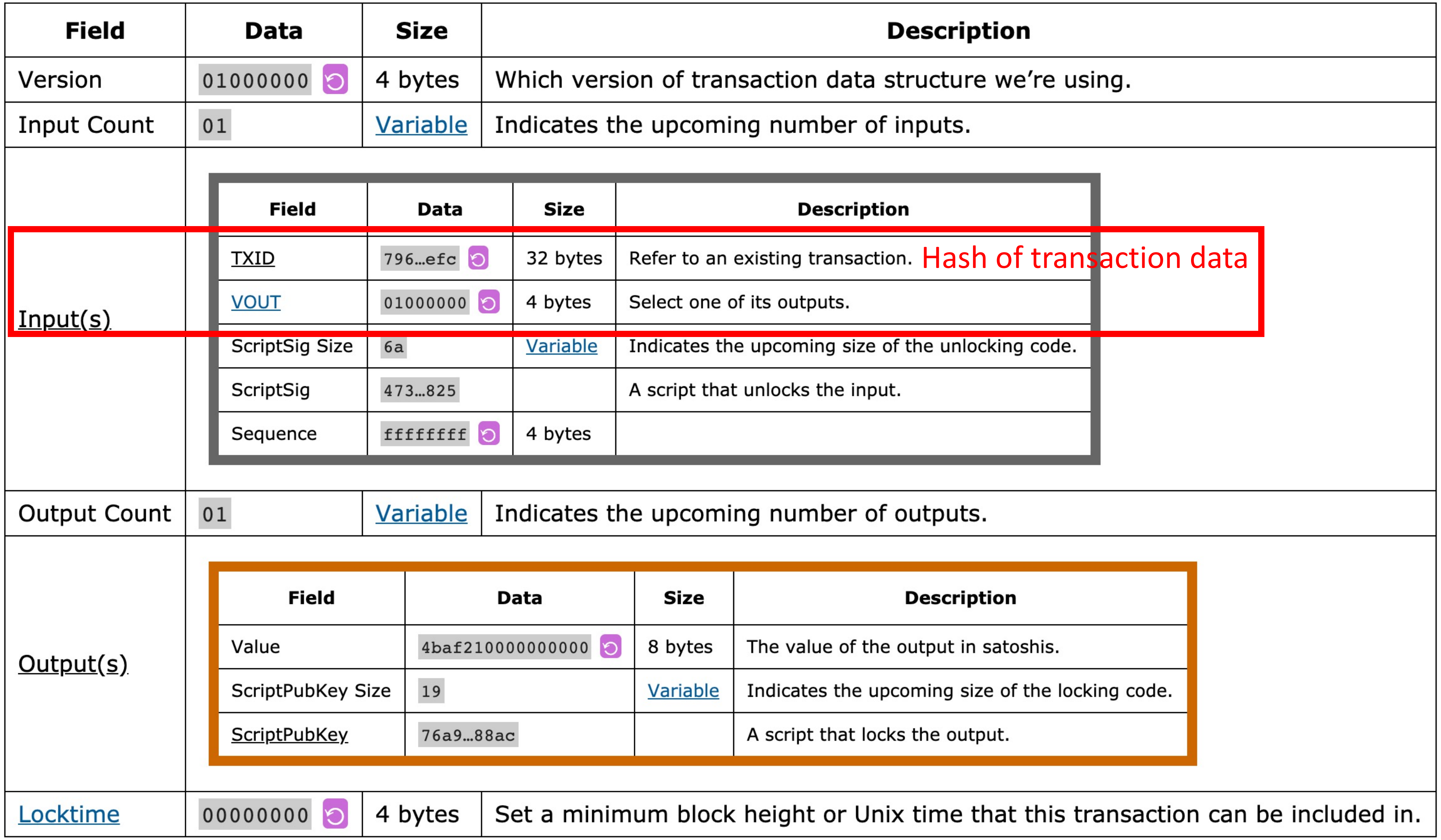
"This output has been locked to the address: 1EUXSxuUVy2PQ5enGXR1a3yxbEjNWMHuem"

(tip: you need the private key to unlock it)

1. Select outputs from the blockchain (ones that are locked to your address).

2. Unlock them using your private key.

Actual Data Structure of Transaction

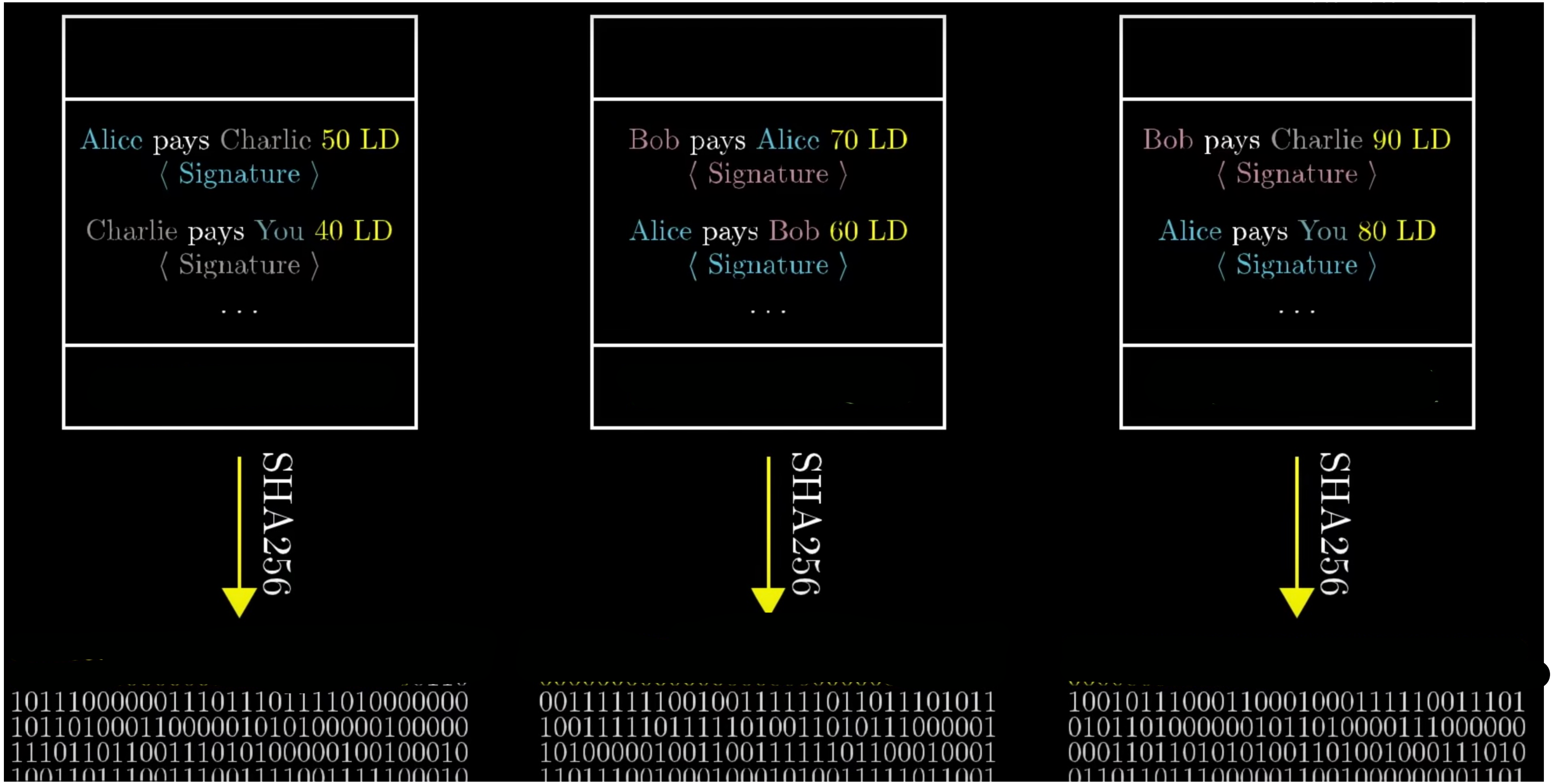


<table>
<tr><th>Field</th><th>Data</th><th>Size</th><th>Description</th></tr>
<tr><td>Version</td><td>01000000</td><td>4 bytes</td><td>Which version of transaction data structure we’re using.</td></tr>
<tr><td>Input Count</td><td>01</td><td>Variable</td><td>Indicates the upcoming number of inputs.</td></tr>
<tr><td>Input(s)</td><td colspan="3">
<table>
<tr><th>Field</th><th>Data</th><th>Size</th><th>Description</th></tr>
<tr><td>TXID</td><td>796…efc</td><td>32 bytes</td><td>Refer to an existing transaction.</td></tr>
<tr><td>VOUT</td><td>01000000</td><td>4 bytes</td><td>Select one of its outputs.</td></tr>
<tr><td>ScriptSig Size</td><td>6a</td><td>Variable</td><td>Indicates the upcoming size of the unlocking code.</td></tr>
<tr><td>ScriptSig</td><td>473…825</td><td></td><td>A script that unlocks the input.</td></tr>
<tr><td>Sequence</td><td>ffffffff</td><td>4 bytes</td><td></td></tr>
</table>
</td></tr>
<tr><td>Output Count</td><td>01</td><td>Variable</td><td>Indicates the upcoming number of outputs.</td></tr>
<tr><td>Output(s)</td><td colspan="3">
<table>
<tr><th>Field</th><th>Data</th><th>Size</th><th>Description</th></tr>
<tr><td>Value</td><td>4baf210000000000</td><td>8 bytes</td><td>The value of the output in satoshis.</td></tr>
<tr><td>ScriptPubKey Size</td><td>19</td><td>Variable</td><td>Indicates the upcoming size of the locking code.</td></tr>
<tr><td>ScriptPubKey</td><td>76a9…88ac</td><td></td><td>A script that locks the output.</td></tr>
</table>
</td></tr>
<tr><td>Locktime</td><td>00000000</td><td>4 bytes</td><td>Set a minimum block height or Unix time that this transaction can be included in.</td></tr>
</table>

Hash of transaction data

Blocks

Hash a block for immutability



Alice pays Charlie 50 LD
⟨ Signature ⟩
Charlie pays You 40 LD
⟨ Signature ⟩
. . .

Bob pays Alice 70 LD
⟨ Signature ⟩
Alice pays Bob 60 LD
⟨ Signature ⟩
. . .

Bob pays Charlie 90 LD
⟨ Signature ⟩
Alice pays You 80 LD
⟨ Signature ⟩
. . .

```
10111000000111011101111010000000
10110100011000001010100000100000
11101101100111010100000100100010
10011011100111001111001111100010
```

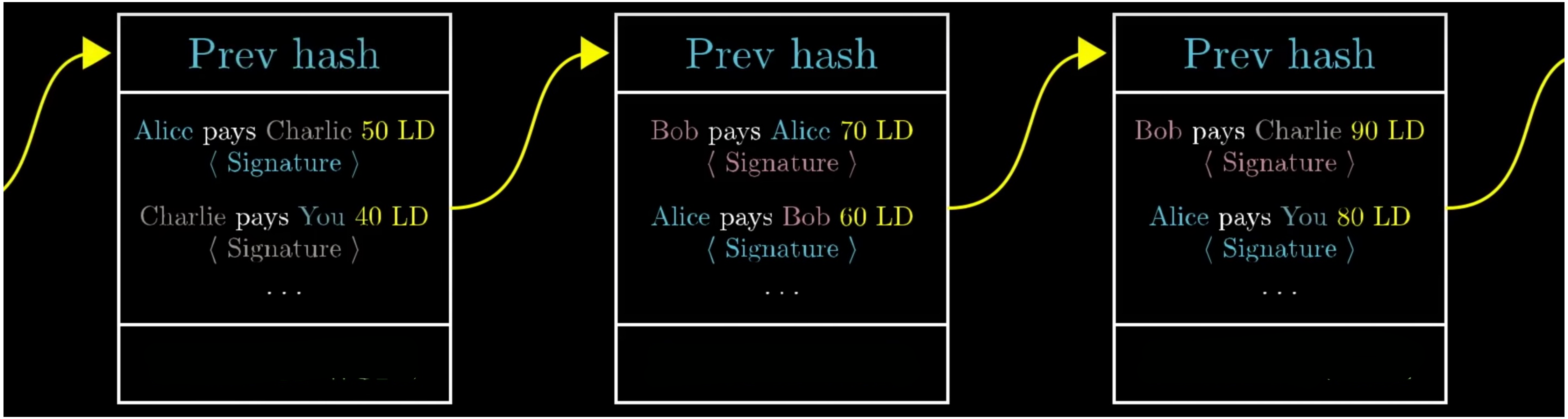
```
00111111100100111111011011101011
10011111011111010011010111000001
10100000100110011111101100010001
11011100100010001010011111011001
```

```
10010111000110001000111110011101
01011010000000101101000011100000
00011011010101001101001000111010
01101101110000011001000000001101
```

Blocks

Chain block hash values for efficiency

- Check hash of every block v.s. check only the last block hash
- Change in one bit of previous block contaminate all hash values after

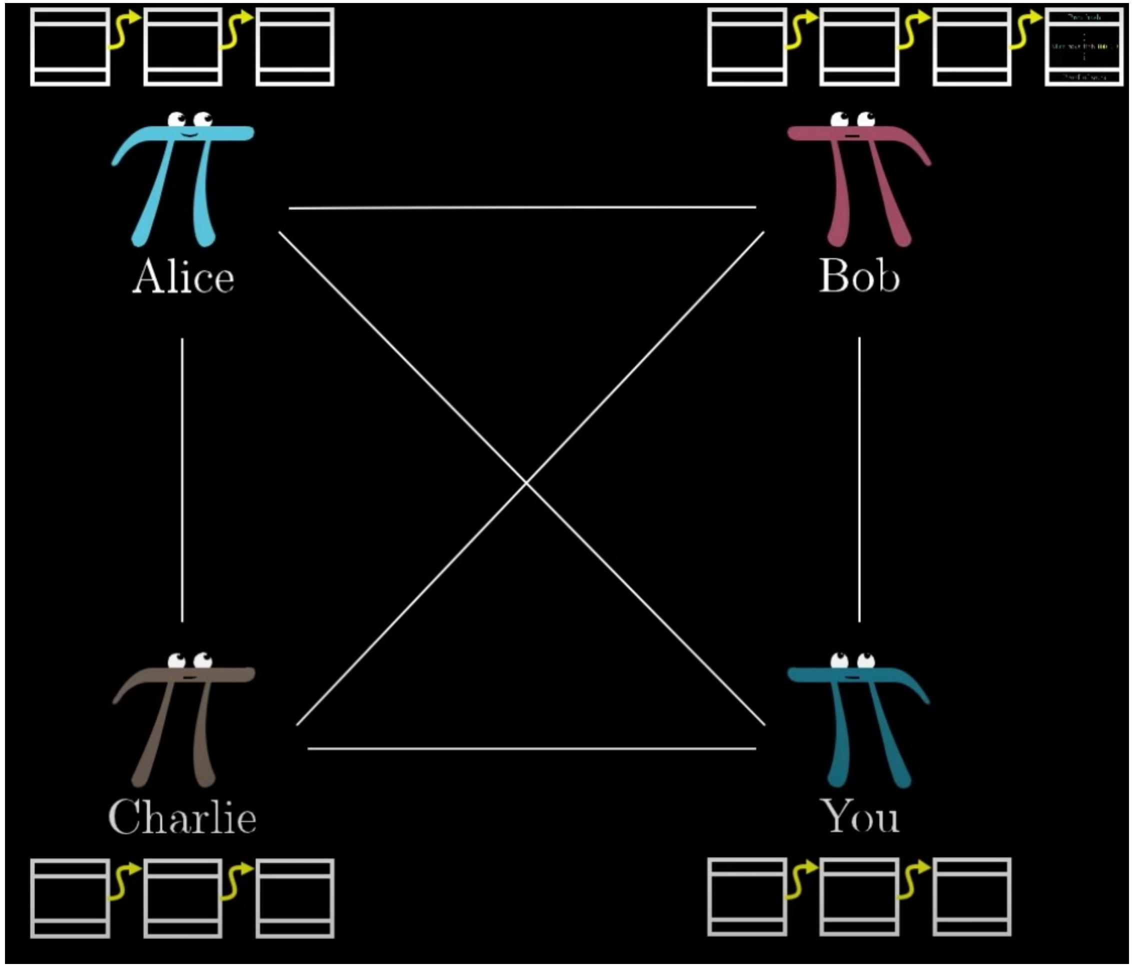


* This is a simplified illustration. In practice, Merkle Tree is used to store transactions in each block.

Blockchain Synchronation

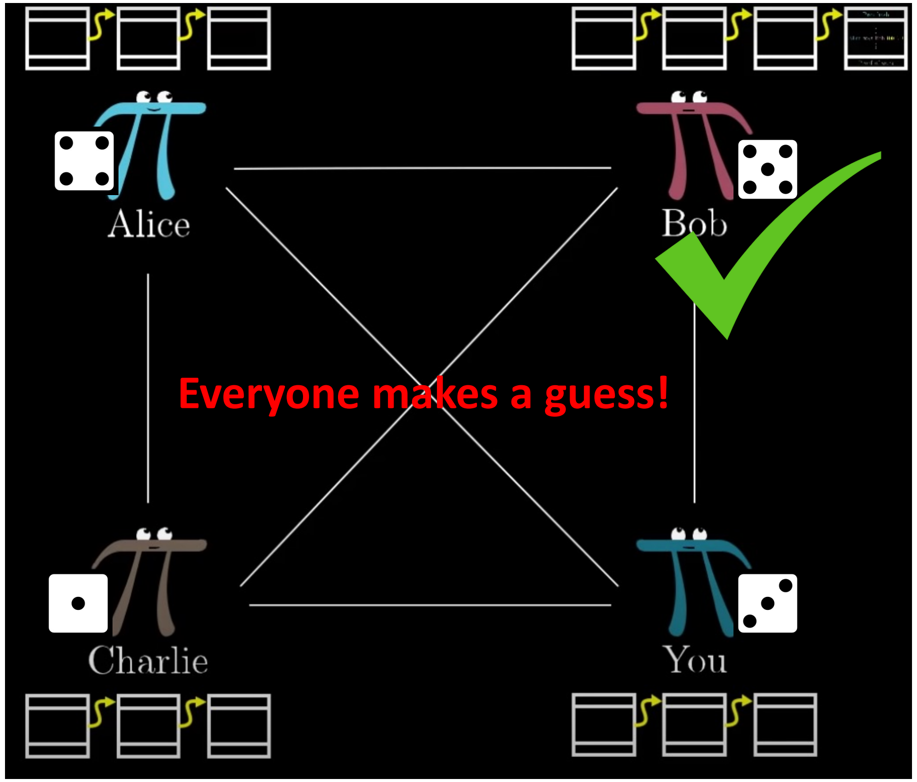
Always copy from the longest chain in the network

- Make sense: longest chain has the most updated transaction data.
- Issue: nothing prevents people forging a long fake chain.



Concensus by Solving Puzzles

- Equal rights to link next new block into the chain.
  - Only the one solve the puzzle can create next block
  - If you want to forge, you need have higher winning rate than others to keep your chain the longest

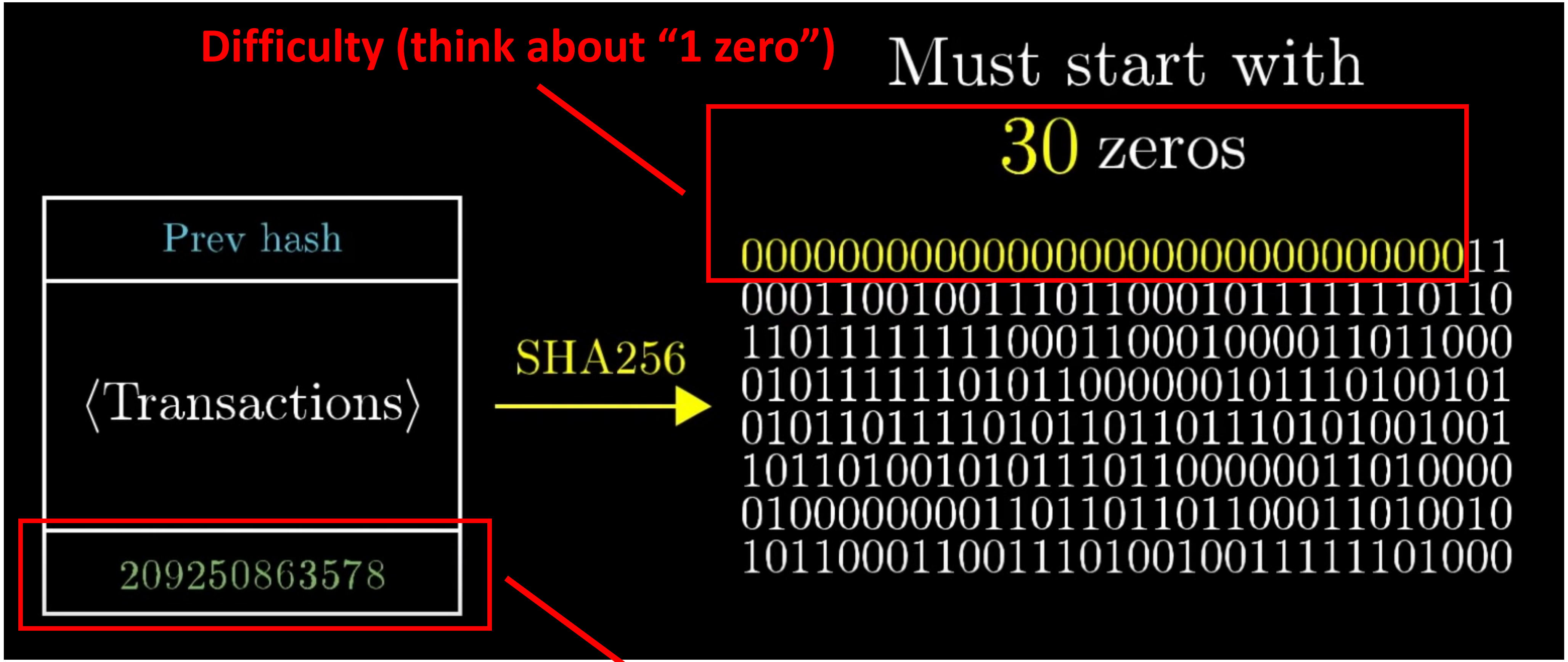


Everyone makes a guess!

God rolling a dice

Centralized 😥

Realistic Puzzle – Hash Guessing (Mining)



```
000000000000000000000000000000 11
00011001001110110001011111110110
11011111111000110001000011011000
01011111101011000000101110100101
01011011110101101101110101001001
10110100101011101100000011010000
01000000001101101101100011010010
10110001100111010010011111101000
```

Difficulty (think about "1 zero")

Make your guess here!

Proof of Work (PoW) Mining

- No one has advantage in solving the hash puzzle
  - Solving (calculating hash) speed is propotional to computational power
  - Assume everyone has equal computational power
- In real life: only need to assume no one owns more than 50% computational power.

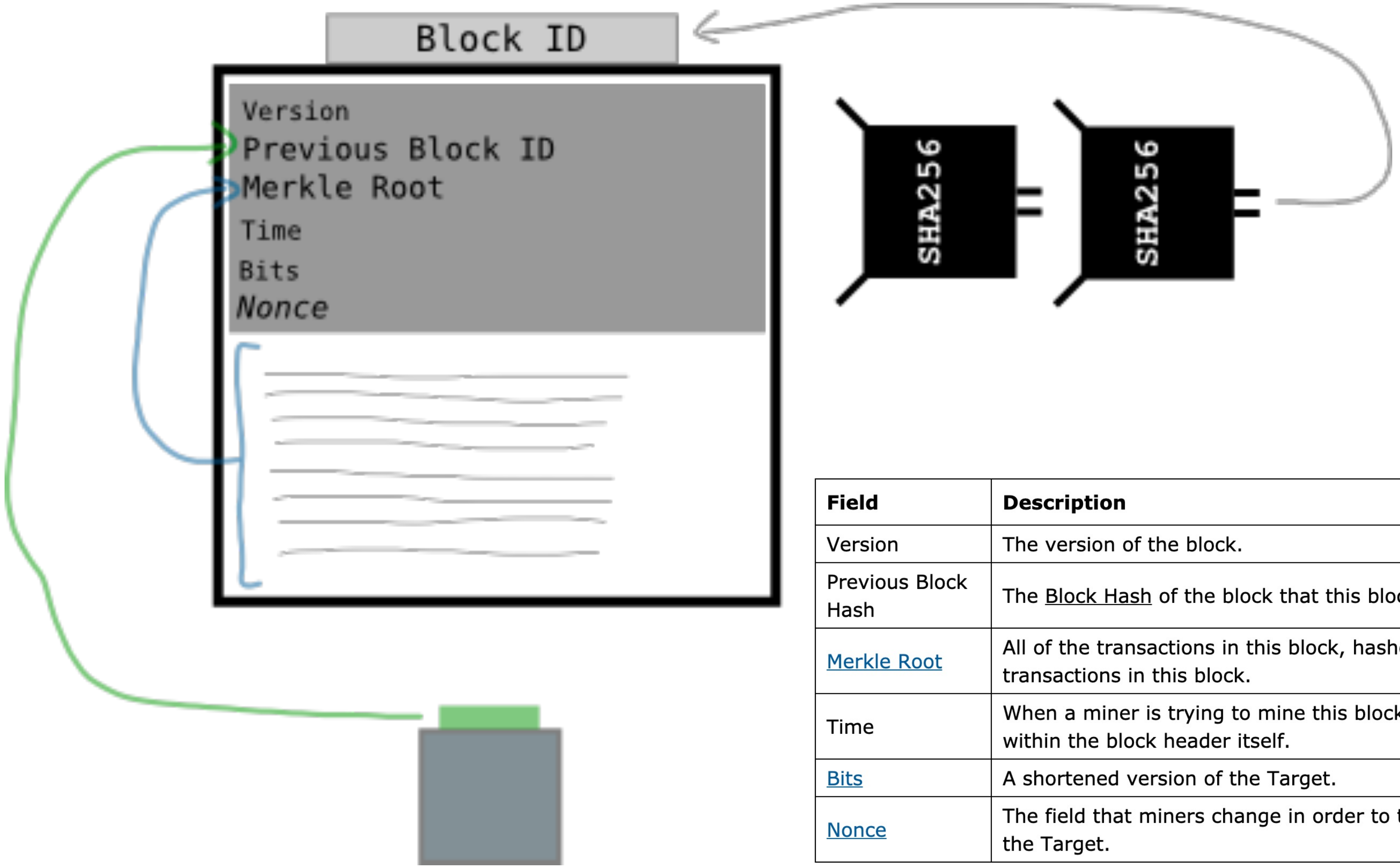
Miners

- Not everyone’s willing to do the mining, but there are people who actively do it.
- Miners incentive: profit!
- **Block reward**: new Bitcoins will be generated to reward you for solving the puzzle.
- **Transaction fees**: all fees attached to the transactions in that block are yours.

>> Why transaction fees? Isn’t block reward enough?

>> Transaction fee unit: satoshi/byte. Why?

Nasty Implmentation Details of Blocks



Field	Description
Version	The version of the block.
Previous Block Hash	The Block Hash of the block that this block is being built on top of. This is what "chains" the blocks together.
Merkle Root	All of the transactions in this block, hashed together. Basically provides a single-line summary of all the transactions in this block.
Time	When a miner is trying to mine this block, the *Unix* time at which this block header is being hashed is noted within the block header itself.
Bits	A shortened version of the Target.
Nonce	The field that miners change in order to try and get a hash of the block header (a Block Hash) that is below the Target.

Recommended Readings & References

- Lean me a Bitcoin, https://learnmeabitcoin.com/, where many figures in my slides from.
- Bitcoin Wiki, https://en.bitcoin.it/wiki/, a comprehensive wiki for Bitcoin
- *"But how does bitcoin actually work?"* by 3Blue1Brown, https://www.youtube.com/watch?v=bBC-nXj3Ng4&t=198s, the best video explaining how Bitcoin works I've seen, I also take some figures from it.
- The original Bitcoin paper, https://bitcoin.org/bitcoin.pdf
- *"A beginners' guide to using the Bitcoin testnet"*, https://www.armedia.com/blog/bitcoin-testnet-beginners-guide/, if you want to play with Bitcoin without paying any real money.