CSE446: Blockchain & Cryptocurrencies

Lecture – 8: Bitcoin-3

BRAC UNIVERSITY

Inspiring Excellence

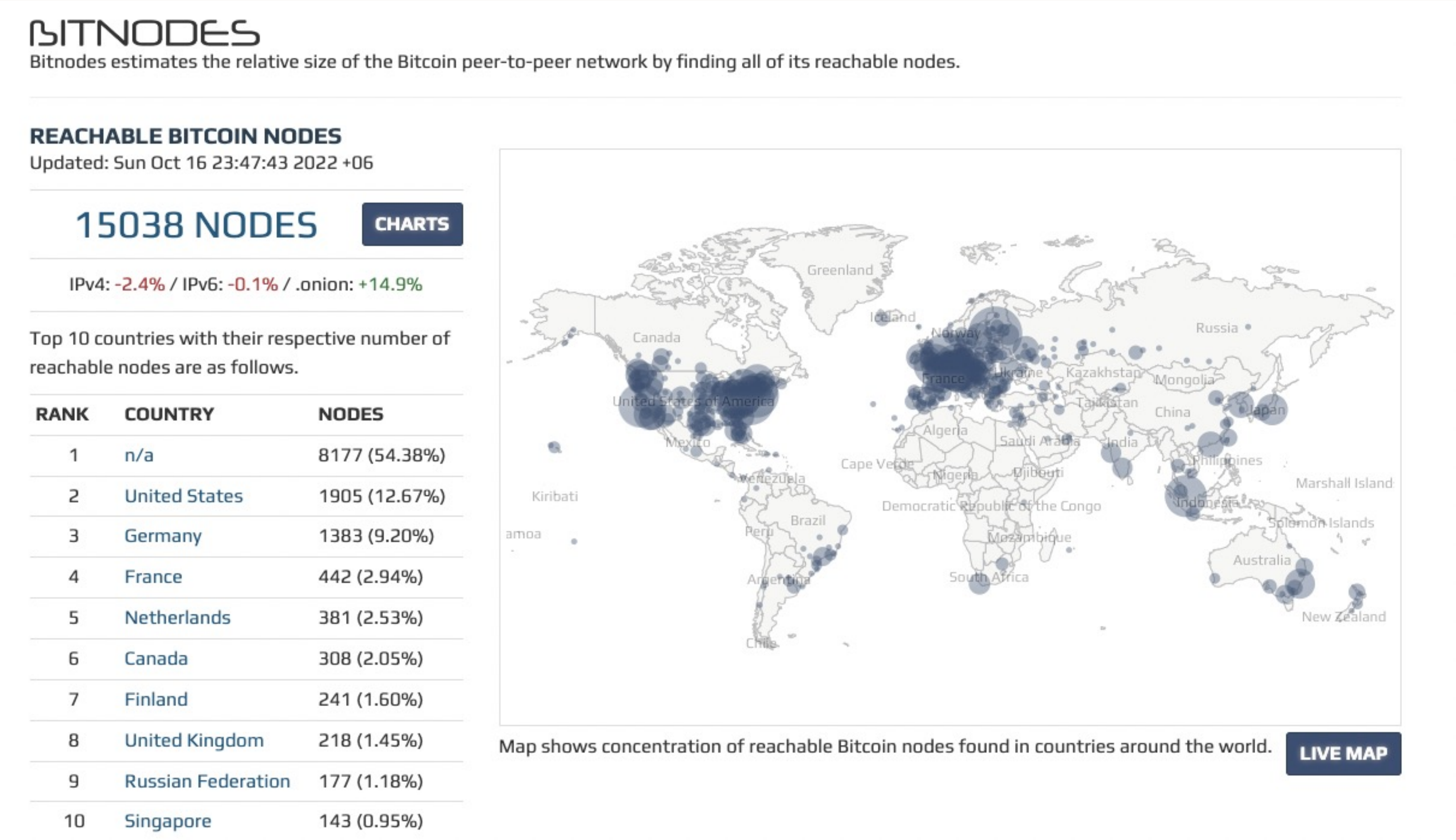
Agenda

- Bitcoin components
  - Users
  - Node & Network
  - Blockchain

Bitcoin Node & Network

- All nodes are connected to a common p2p network
- Every node runs a bitcoin implementation (bitcoind, bcoin, etc.)
    - implementations are open source
- Anyone can freely join the network
- Nodes do not have to trust the network!
- Everybody assumes that neighbours may lie (byzantine behaviour)
- Every node receives messages, acts on them and passes these messages to its known neighbours according to protocols
    - malicious nodes can suppress messages and behave beyond protocols rules

Bitcoin Node & Network



BITNODES

Bitnodes estimates the relative size of the Bitcoin peer-to-peer network by finding all of its reachable nodes.

REACHABLE BITCOIN NODES

Updated: Sun Oct 16 23:47:43 2022 +06

15038 NODES

CHARTS

IPv4: -2.4% / IPv6: -0.1% / .onion: +14.9%

Top 10 countries with their respective number of reachable nodes are as follows.

RANK	COUNTRY	NODES
1	n/a	8177 (54.38%)
2	United States	1905 (12.67%)
3	Germany	1383 (9.20%)
4	France	442 (2.94%)
5	Netherlands	381 (2.53%)
6	Canada	308 (2.05%)
7	Finland	241 (1.60%)
8	United Kingdom	218 (1.45%)
9	Russian Federation	177 (1.18%)
10	Singapore	143 (0.95%)

Map shows concentration of reachable Bitcoin nodes found in countries around the world.

LIVE MAP

https://bitnodes.io/

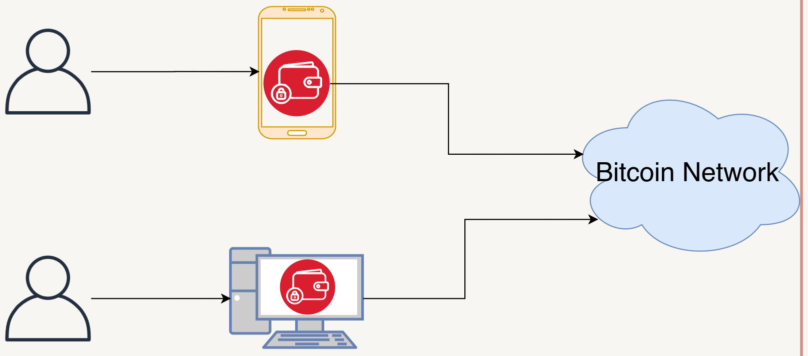
Live map available: https://bitnodes.io/nodes/live-map/

Bitcoin Node

- Bitcoin has four types of nodes:
  - Wallet node
  - Light node
  - Full node
  - Miner node

Bitcoin Node types: wallet node (user)

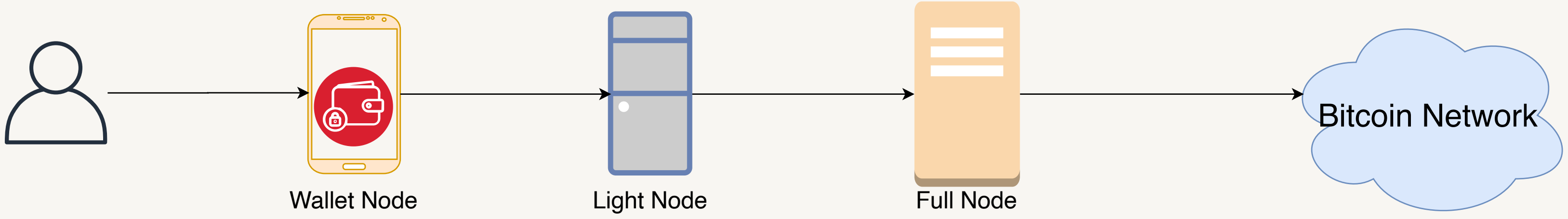
- The wallet owner owns different private keys
- He is the owner of all stored currencies on these addresses
- He sends money by signing and publishing new transactions to a connected light node, full node or miner node



Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Bitcoin Node types: light node (software)

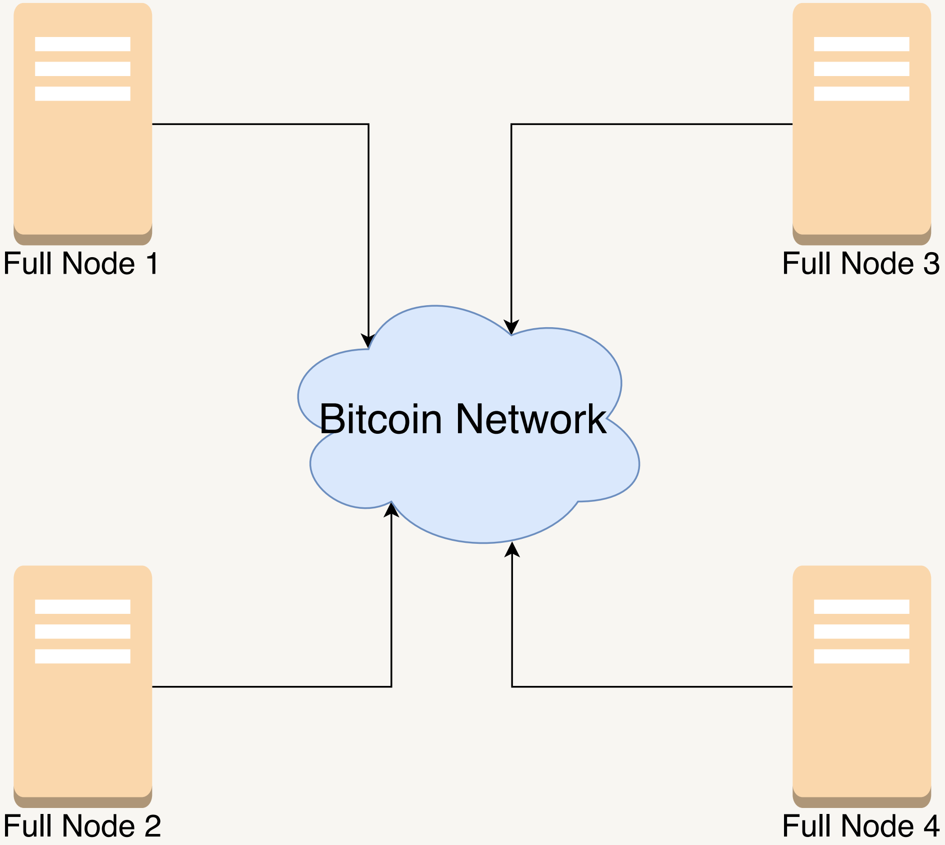
- The light node can act as a relay for transactions of one wallet owner
- It validates whether a single transaction of the wallet owner was executed correctly
- The light node also requires a full node to connect to the network
- Almost no relevance in practice today
- Today, centralised services are used to create transactions



Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Bitcoin Node types: full node (software)

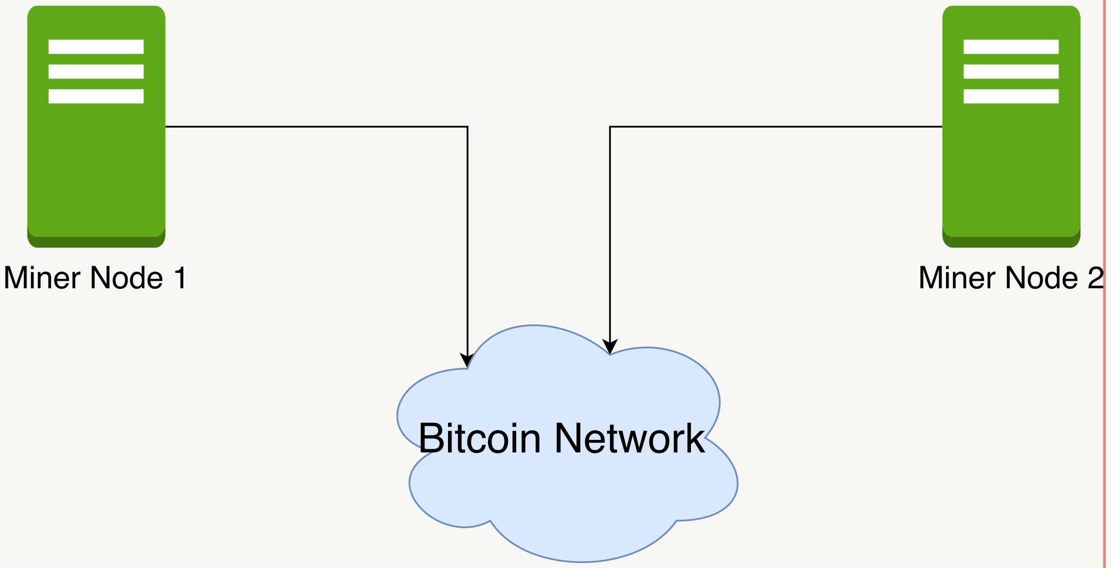
- The full node maintains the complete blockchain
- Its record of the chain is complete
  - it contains every single transaction and block until the genesis (first) block
- Is connected to other full nodes and exchanges information
- Namely:
  - Validates every transaction and block it receives
  - Relays all new transactions and blocks



Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

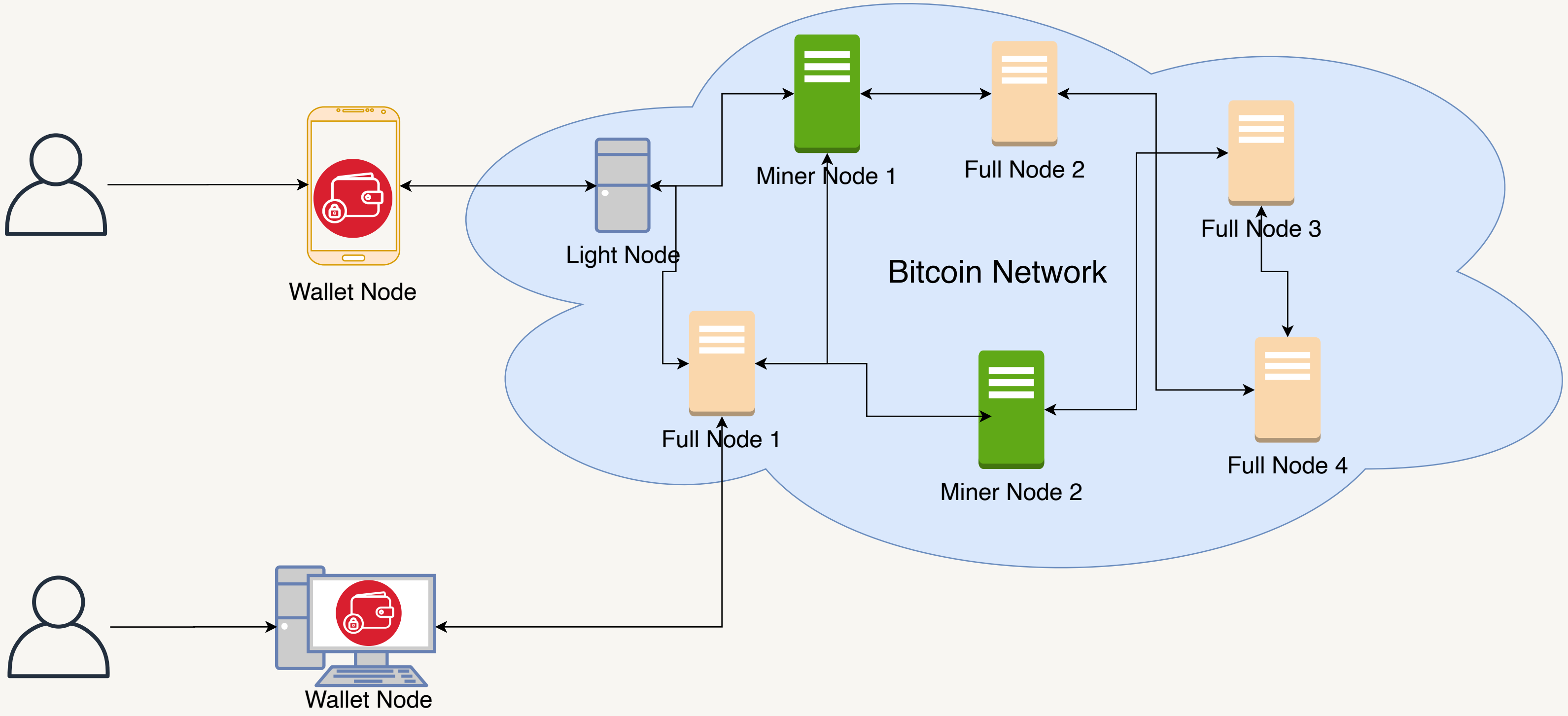
Bitcoin Node types: miner node (software)

- The miner needs the same record as a full node to work properly
- It also is connected with other nodes and maintains the network
- Additionally, the miner is responsible for creating new blocks by trying to solve the mining puzzle
- The miner gets rewarded for creating new blocks



Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Bitcoin network



Bitcoin P2P network

- Bitcoin nodes communicate in a decentralised fashion, meaning that no single entity or node is superior, all nodes are equal
    - Ad-hoc protocol (runs on TCP port 8333)
    - Ad-hoc network with random topology
- New nodes can join at any time
- Forget non-responding nodes after 3 hr

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Bitcoin P2P network

- To communicate, they need to have clear rules
  - How to find other nodes (bootstrapping)
  - How to send and receive transactions
  - How to send and receive blocks
  - How to sync the blockchain
- The basic network uses a peer-to-peer gossip protocol for
  - Node discovery, node status maintenance
  - Messages about new blocks or transactions

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

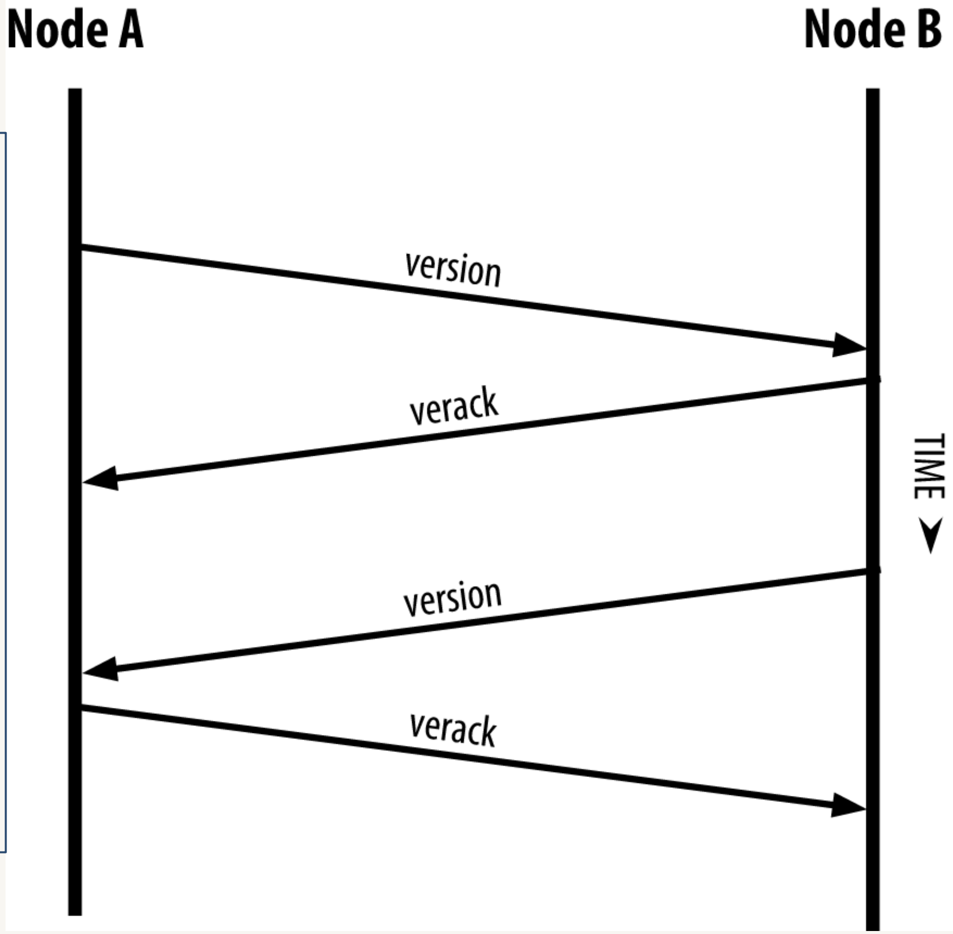
Node discovery/bootstrapping

- Adding a new node into the network is called bootstrapping
- The new node needs to discover other nodes in the network to connect to the P2P network
- How does it know who to connect to?
  - Hard-coded DNS-services which offer IP-addresses of nodes
  - Hard-coded seed addresses (last resort)
  - Command-line provided addresses
  - Text-file provided addresses

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Node discovery/bootstrapping

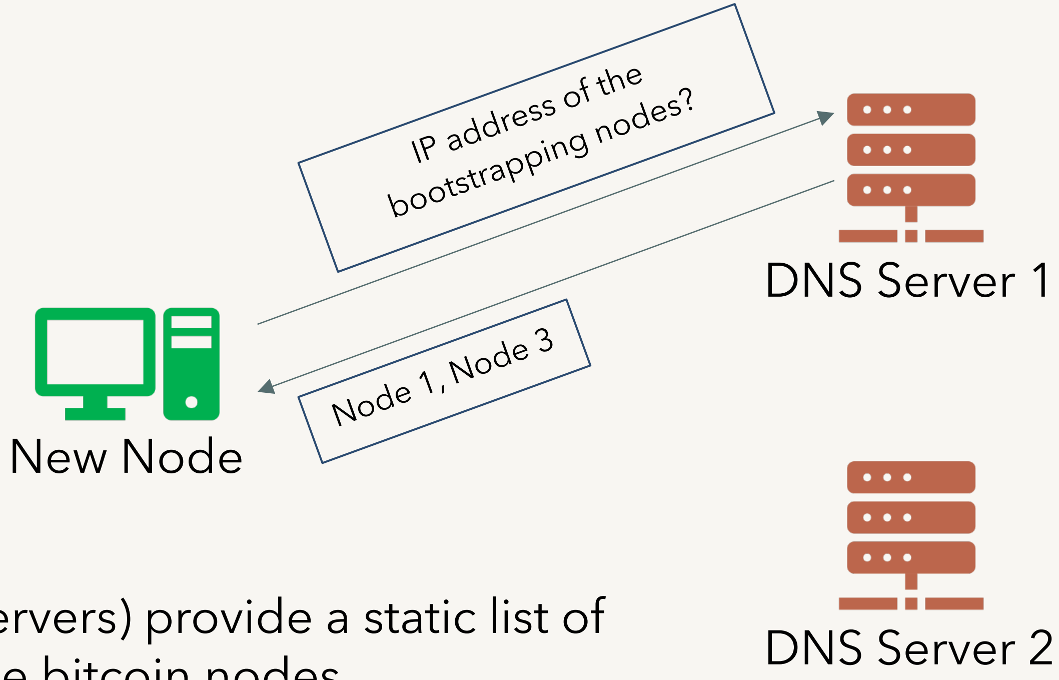
- The first message to another Bitcoin peer is the version message
- Using version each node check if the other node is compatible
- If compatible, the other node sends the version acknowledgement (verack) message



Node discovery/bootstrapping

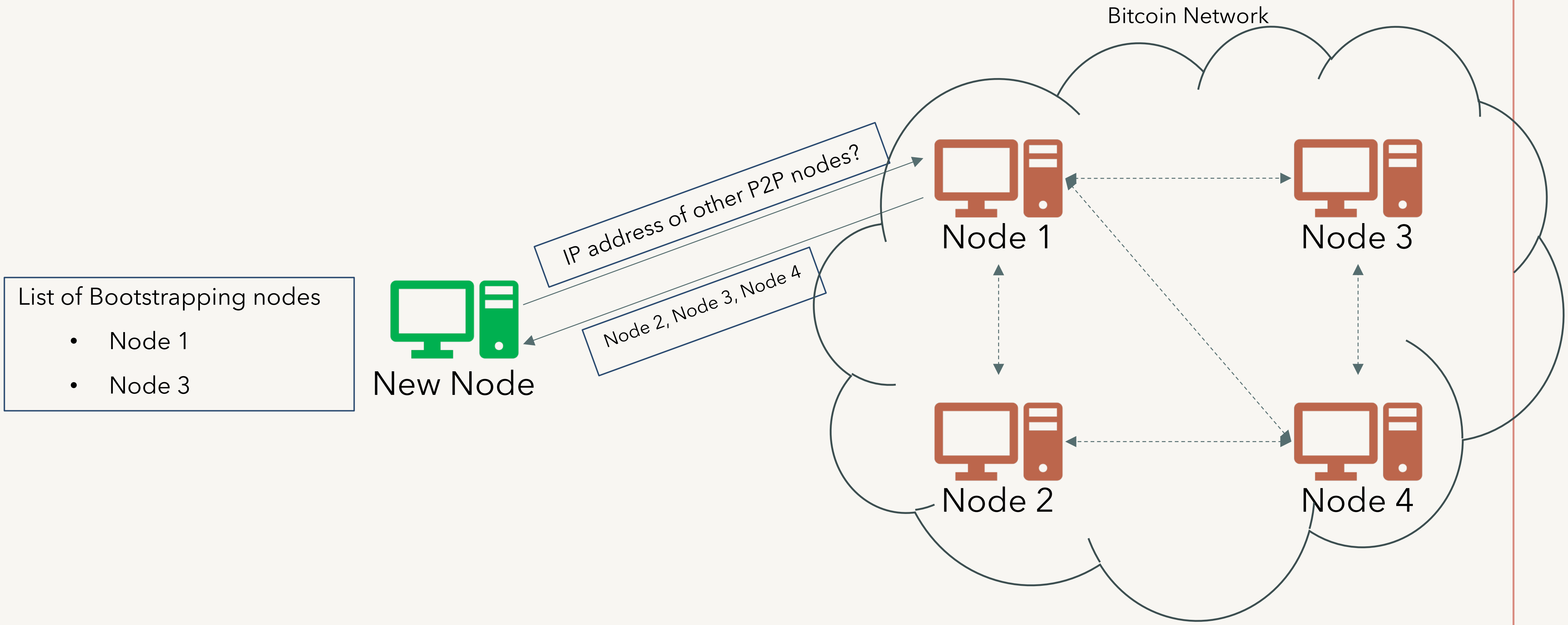
List of DNS Servers (their IP address)

- DNS Server 1
- DNS Server 2

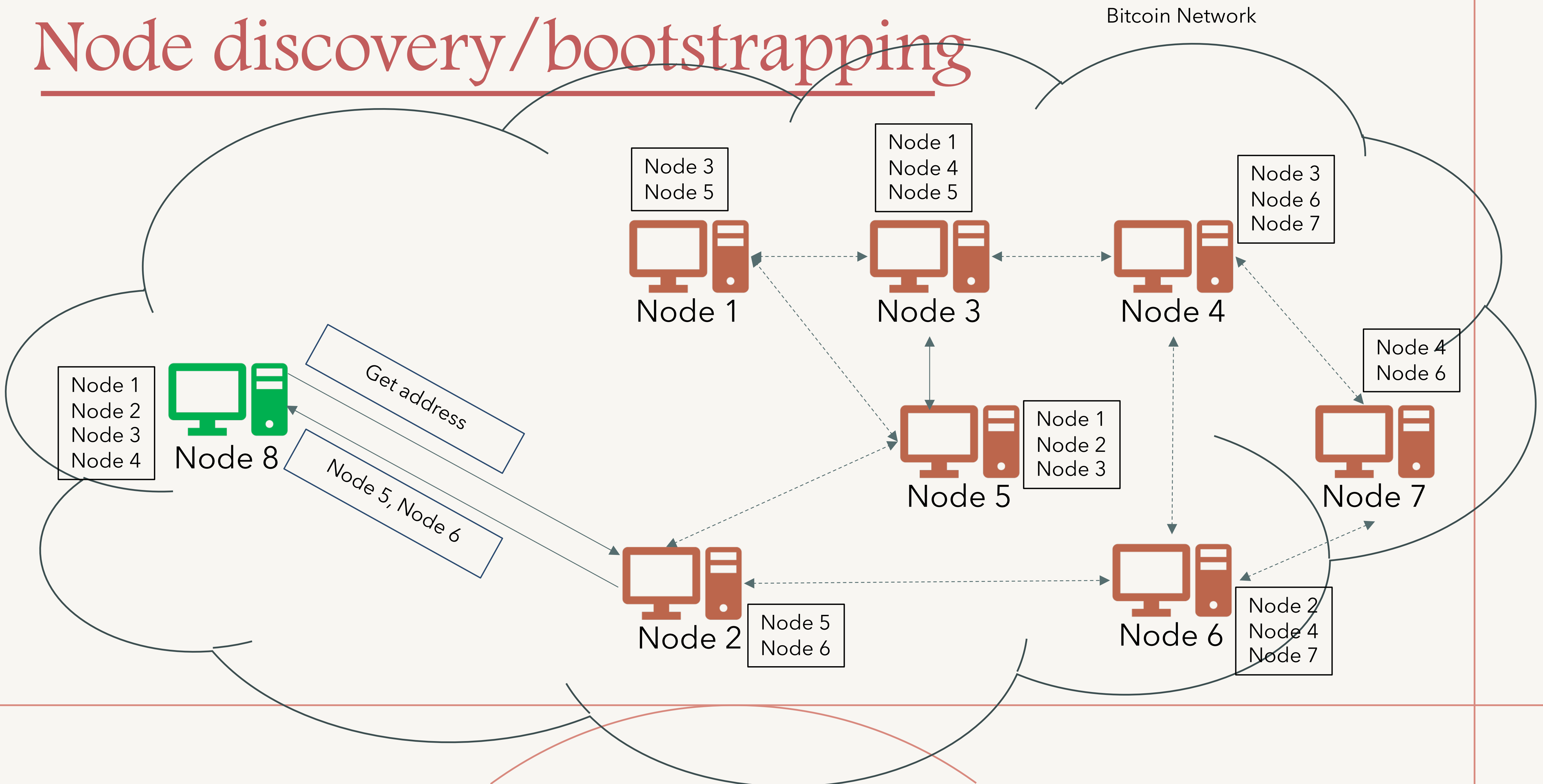


These DNS seeds (servers) provide a static list of IP addresses of stable bitcoin nodes

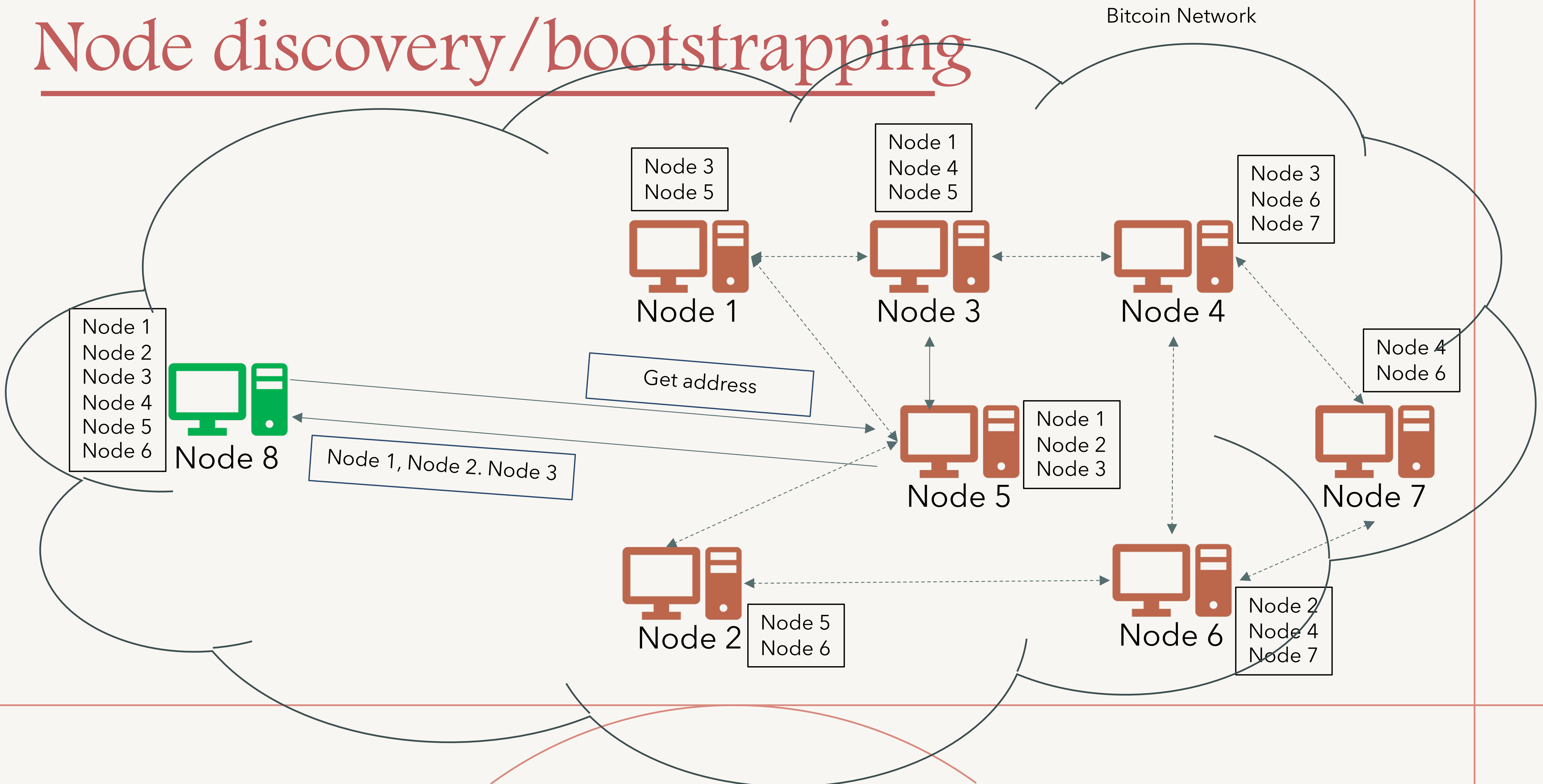
Node discovery/bootstrapping



Node discovery/bootstrapping



Node discovery/bootstrapping



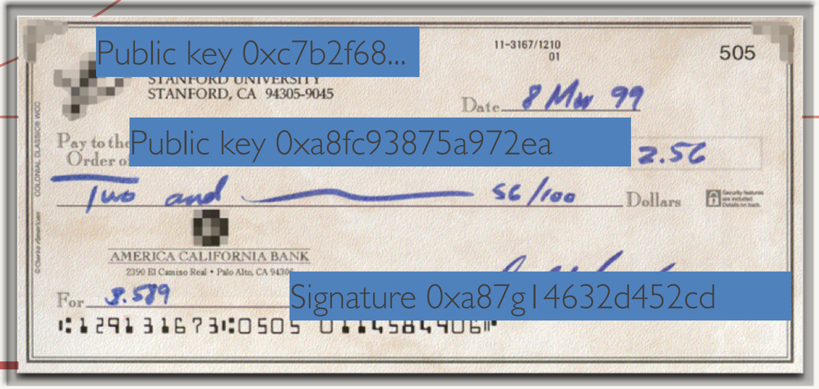
Bitcoin blockchain

- There are three different things to understand
  - Transaction
  - Block
  - Blockchain

Transaction

- Transactions are the most important part of the bitcoin system
- Everything else in bitcoin is designed to ensure that transactions can be created, propagated on the network, validated, and finally added to the global ledger of transactions (the blockchain)
- Transactions are data structures that encode the transfer of values between participants in the bitcoin system
    - A transaction transfers money from somebody to somebody else
- Each transaction is a public entry in bitcoin's blockchain, the global double-entry bookkeeping ledger
    - an entry that affects at least two different accounts
- All transactions are public
- Everybody can see all transactions in a blockchain explorer

Transaction



Public key 0xc7b2f68...

STANFORD UNIVERSITY
STANFORD, CA 94305-9045

11-3167/1210
01

505

Date 8 Mw 99

Pay to the Order of Public key 0xa8fc93875a972ea 2.56

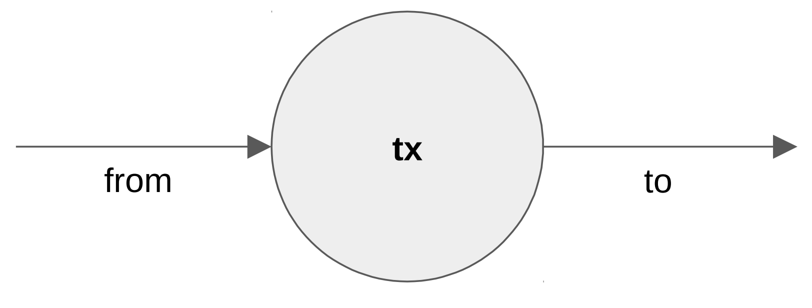
Two and 56/100 Dollars

AMERICA CALIFORNIA BANK
2390 El Camino Real • Palo Alto, CA 9430

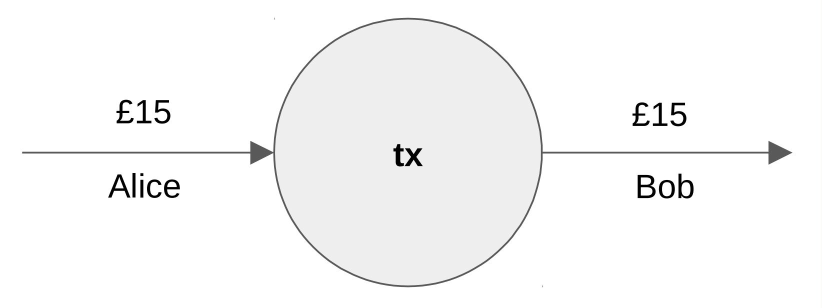
For 8.589

Signature 0xa87g14632d452cd

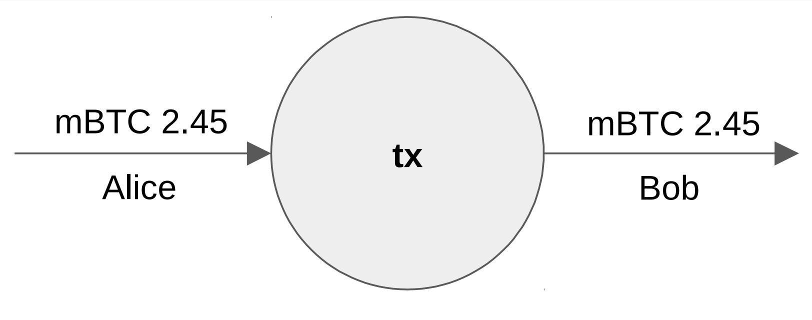
⑆1291316 73⑆0505 0114584906⑈



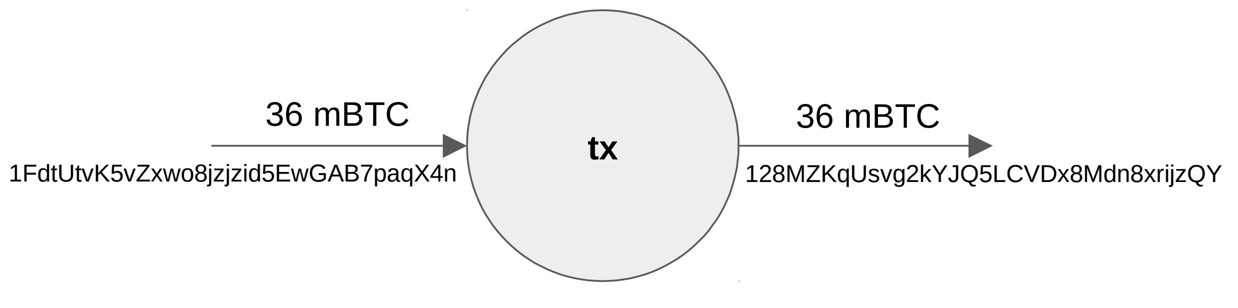
Abstract format of any transaction



A traditional transaction

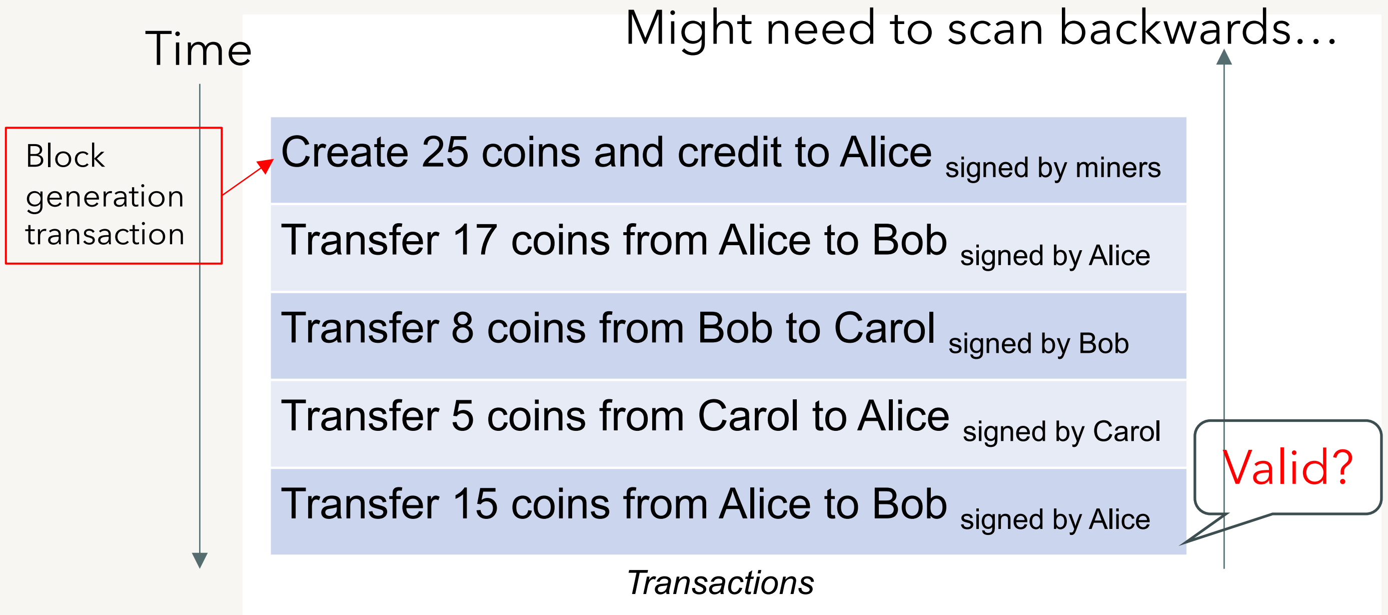


Abstract format of a bitcoin transaction


36 mBTC
1FdtUtvK5vZxwo8jzjzid5EwGAB7paqX4n
tx
36 mBTC
128MZKqUsvg2kYJQ5LCVDx8Mdn8xrijzQY

A bitcoin transaction

Account/transaction based Ledger?



Create 25 coins and credit to Alice signed by miners

Transfer 17 coins from Alice to Bob signed by Alice

Transfer 8 coins from Bob to Carol signed by Bob

Transfer 5 coins from Carol to Alice signed by Carol

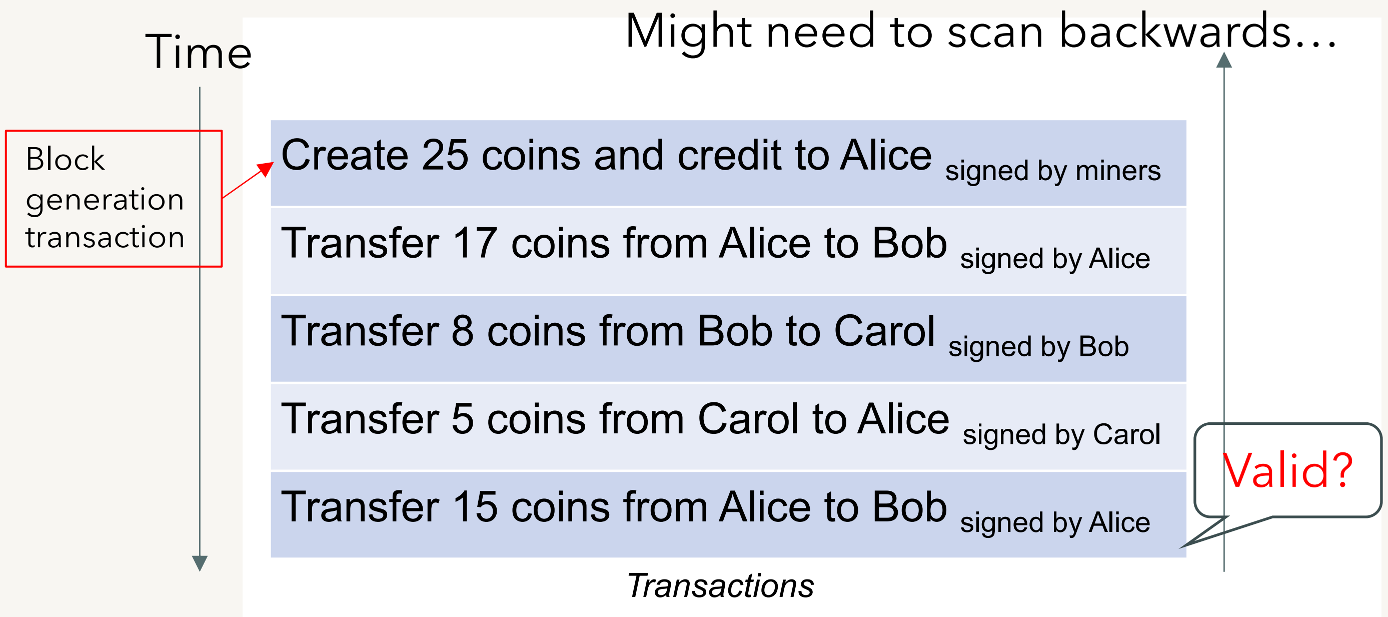
Transfer 15 coins from Alice to Bob signed by Alice

Transactions

- *Intuitively*: At first, we consider Bitcoin to use an account-based ledger. However, an account-based approach takes a lot of effort to track the balances of every account
- In an account-based ledger, transactions can transfer arbitrary amounts of coins between accounts
- Transactions lead to a “world-state” of accounts and account balances

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Account/transaction based Ledger?



Might need to scan backwards…

Time

Block generation transaction

Create 25 coins and credit to Alice signed by miners

Transfer 17 coins from Alice to Bob signed by Alice

Transfer 8 coins from Bob to Carol signed by Bob

Transfer 5 coins from Carol to Alice signed by Carol

Transfer 15 coins from Alice to Bob signed by Alice

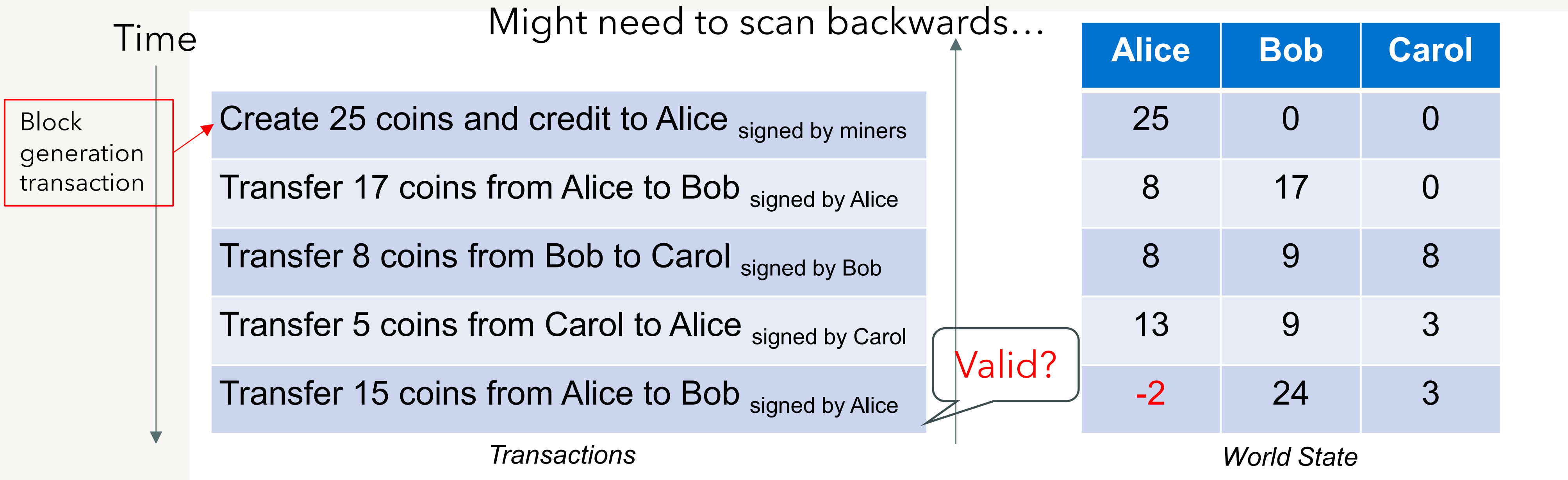
Valid?

Transactions

- To validate a certain transaction, you might need to track very old transactions
- Since you do not know which old transaction, you need track each of the previous transactions one by one until you find the desired ones

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Account/transaction based Ledger?



Create 25 coins and credit to Alice signed by miners
Transfer 17 coins from Alice to Bob signed by Alice
Transfer 8 coins from Bob to Carol signed by Bob
Transfer 5 coins from Carol to Alice signed by Carol
Transfer 15 coins from Alice to Bob signed by Alice

Transactions

Alice	Bob	Carol
25	0	0
8	17	0
8	9	8
13	9	3
-2	24	3

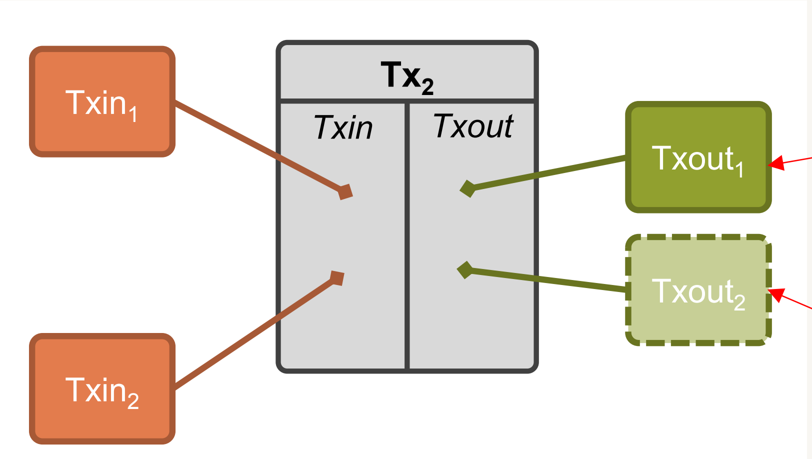
World State

- One option is to maintain the world state in a separate database
  - Remember, in the world state you store the account balance
- That would require to maintain two separate databases: world state and transactions

Account/transaction based Ledger?

- Bitcoin’s solution: a transaction-based ledger
- By using a transaction-based ledger, Bitcoin enables wallet owners to define conditional transactions using Bitcoin Script

Transaction Based Ledger



Output not consumed, not used as inputs in other transactions These are known as **UTXO (Unspent Transaction Output)**

Output consumed, used as inputs in other transactions. These are known as **STXO (Spent Transaction Output**)

- Transactions (Tx) have a number of inputs and a number of outputs
    - Inputs (Txin): Former outputs, that are being consumed
    - Outputs (Txout): New outputs transferring the value
- In transactions (coinbase transaction) where new coins are created, no Txin is used (no coins are consumed)
- Each transaction has a unique identifier (TxID). Each output has a unique identifier within a transaction
- We refer to them (in this example) as *#TX[#txout]*, e.g., 1[1], which is the second Txout of the second transaction

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Transaction Based Ledger

This is known as a coinbase transaction

- Create 25 coins and credit to Alice signed by miners
- Transfer 17 coins from Alice to Bob signed by Alice
- Transfer 8 coins from Bob to Carol signed by Bob
- Transfer 5 coins from Carol to Alice signed by Carol
- Transfer 15 coins from Alice to Bob signed by Alice

Transactions

0	Txin: Ø Txout: 25.0 -> Alice signed by the miner
1	Txin: 0[0] Txout: 17.0 → Bob, 8.0 → Alice signed by Alice
2	Txin: 1[0] Txout: Txout: 8.0 → Carol, 9.0 → Bob signed by Bob
3	Txin: 2[0] Txout: 5.0 → Alice, 3.0 → Carol signed by Carol
4	Txin: 1[1], 3[0] Txout: 15.0 → Bob, ? → Alice signed by Alice

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Transaction Based Ledger

Create 25 coins and credit to Alice signed by miners

Transfer 17 coins from Alice to Bob signed by Alice

Transfer 8 coins from Bob to Carol signed by Bob

Transfer 5 coins from Carol to Alice signed by Carol

Transfer 15 coins from Alice to Bob signed by Alice

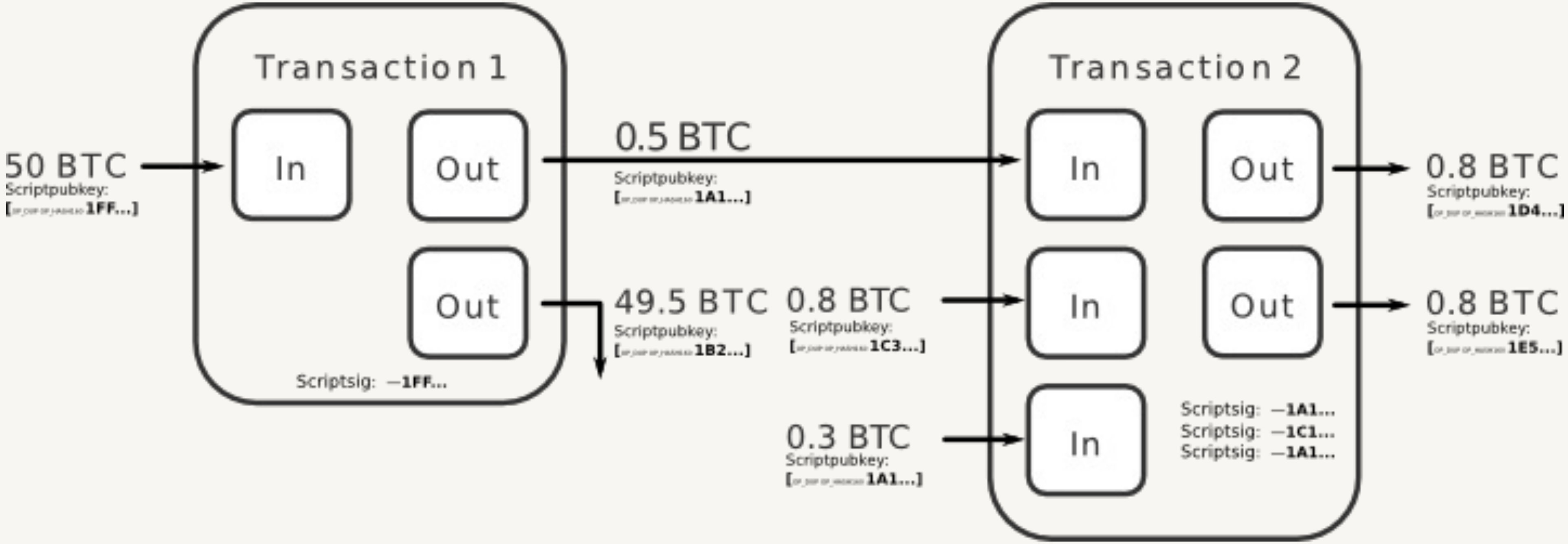
Transactions

0	Txin: Ø Txout: 25.0 -> Alice signed by the miner
1	Txin: 0[0] Txout: 17.0 → Bob, 8.0 → Alice signed by Alice
2	Txin: 1[0] Txout: Txout: 8.0 → Carol, 9.0 → Bob signed by Bob
3	Txin: 2[0] Txout: 5.0 → Alice, 3.0 → Carol signed by Carol
4	Txin: 1[1], 2[0] Txout: 15.0 → Bob, ? → Alice signed by Alice

Joined payment

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Connecting different transactions

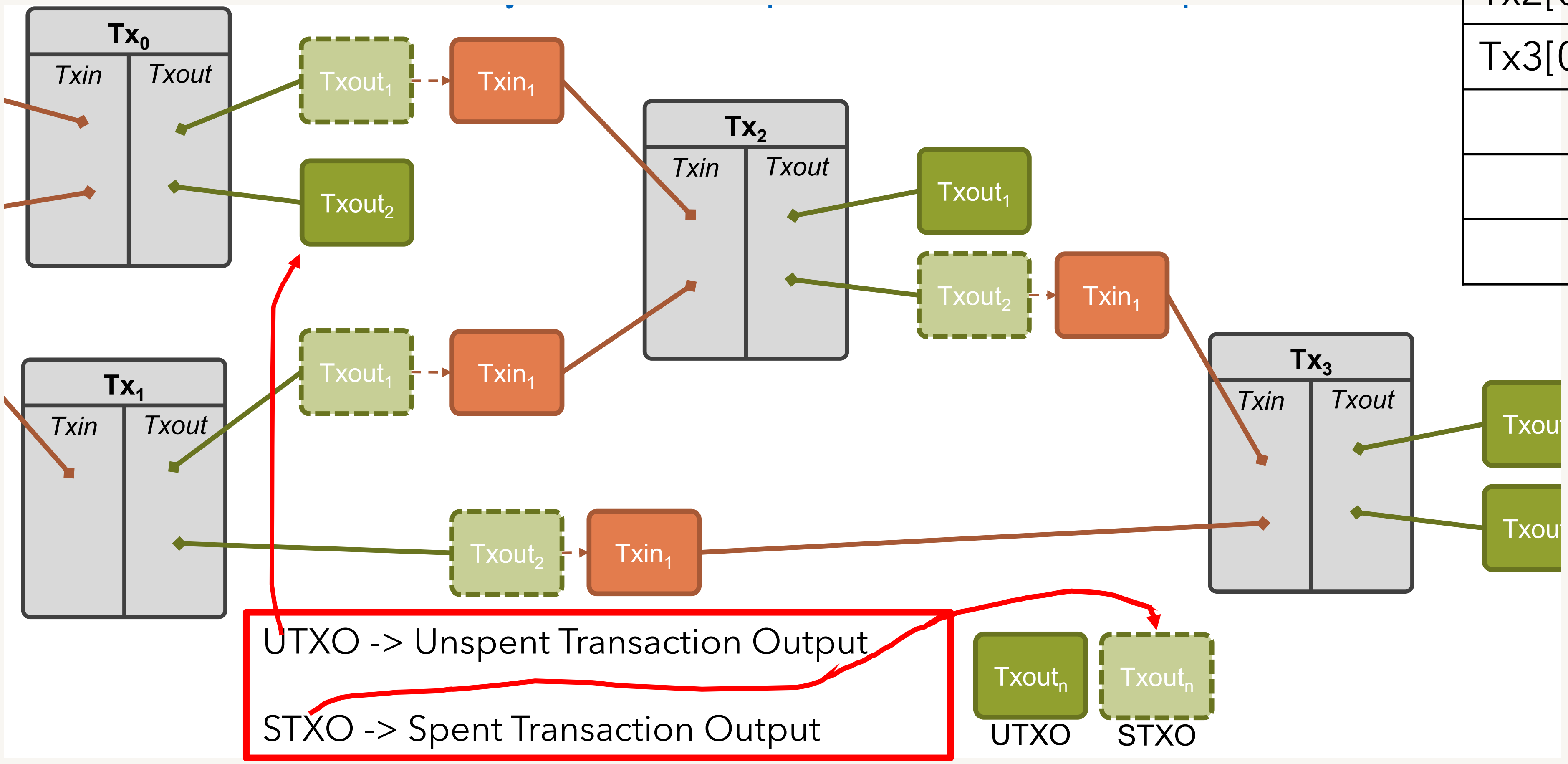


Source: http://s3.amazonaws.com/main-newsbtc-images/2015/11/27120846/Coinjoin.png

Connecting different transactions

All UTXOs are maintained in an UTXO Table

Tx0[1]
Tx2[0]
Tx3[0]
…
…
…



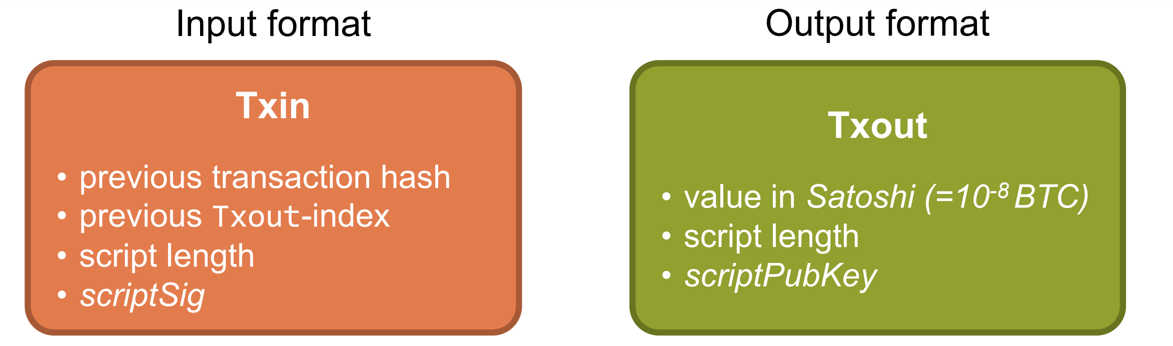
UTXO -> Unspent Transaction Output

STXO -> Spent Transaction Output

Bitcoin Basics- Gallersdörfer, U., Holl, P., & Matthes, F. (2020). "Blockchain-based Systems Engineering". Lecture Slides. TU Munich.

Transaction

- Each transaction has a list of inputs and outputs
- All inputs reference an existing unspent output or a coinbase transaction
- Inputs and outputs contain scripts (scriptSig, scriptPubKey) for verification and other metadata
- lock_time: is the time at which a particular transaction can be added to the blockchain, 0 means now



Input format

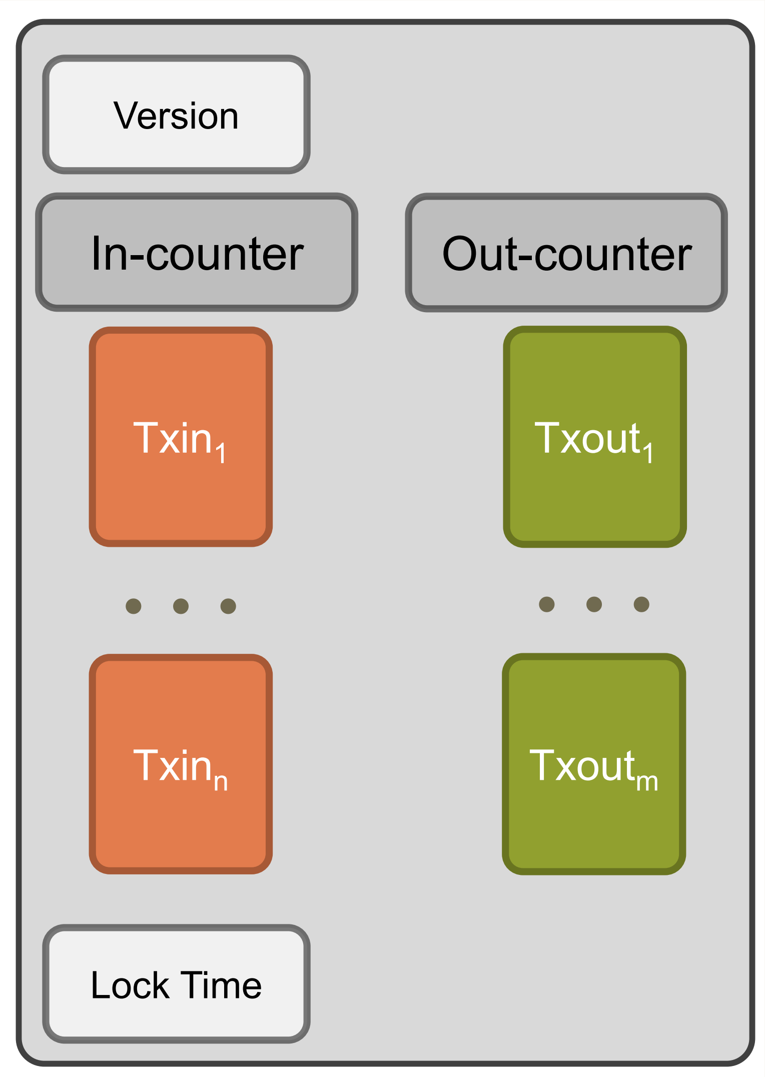
Txin

- previous transaction hash
- previous Txout-index
- script length
- *scriptSig*

Output format

Txout

- value in *Satoshi* (=10^{-8} BTC)
- script length
- *scriptPubKey*



Version

In-counter

Out-counter

Txin_1

Txout_1

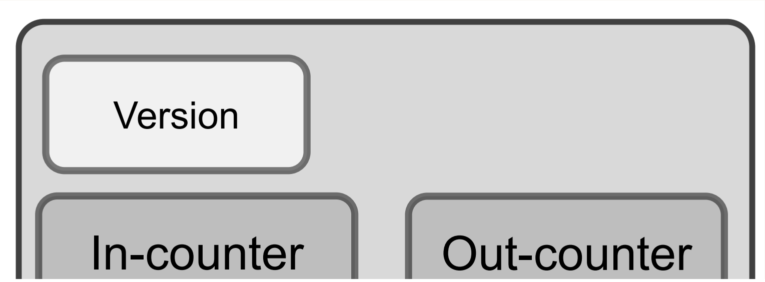
Txin_n

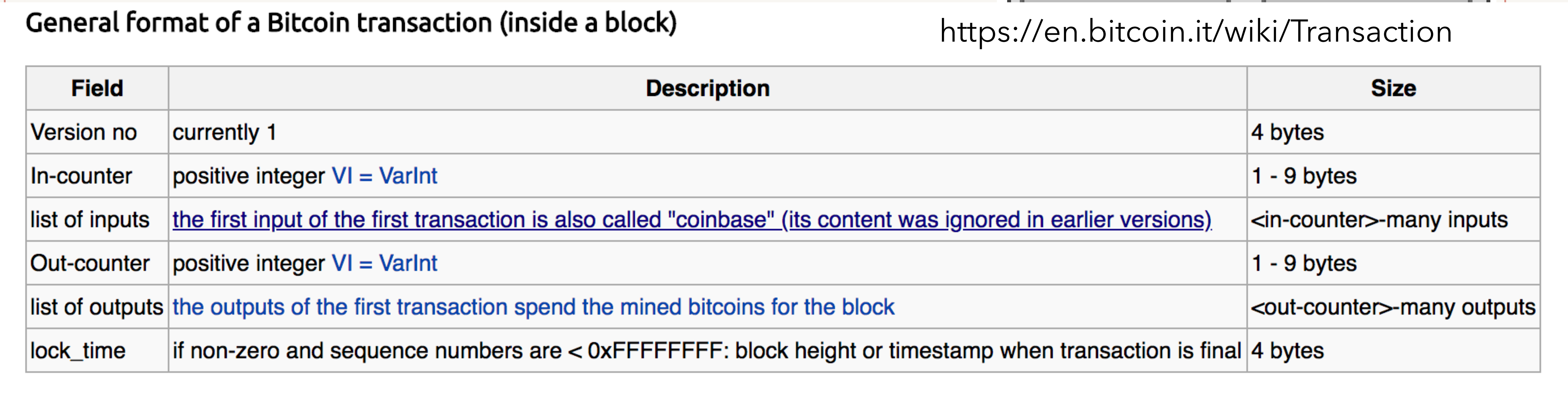
Txout_m

Lock Time

Transaction

- All inputs reference an existing unspent output or a

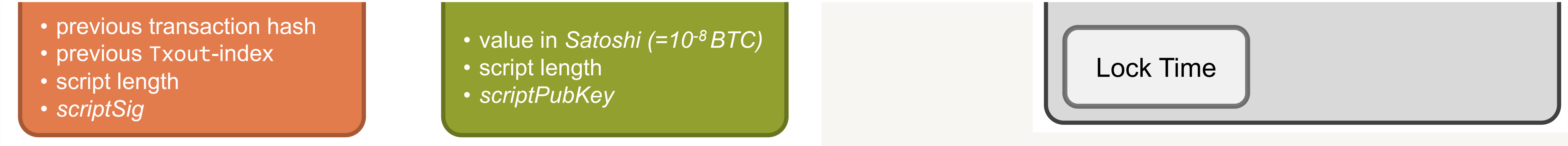




General format of a Bitcoin transaction (inside a block)

https://en.bitcoin.it/wiki/Transaction

Field	Description	Size
Version no	currently 1	4 bytes
In-counter	positive integer VI = VarInt	1 - 9 bytes
list of inputs	the first input of the first transaction is also called "coinbase" (its content was ignored in earlier versions)	<in-counter>-many inputs
Out-counter	positive integer VI = VarInt	1 - 9 bytes
list of outputs	the outputs of the first transaction spend the mined bitcoins for the block	<out-counter>-many outputs
lock_time	if non-zero and sequence numbers are < 0xFFFFFFFF: block height or timestamp when transaction is final	4 bytes

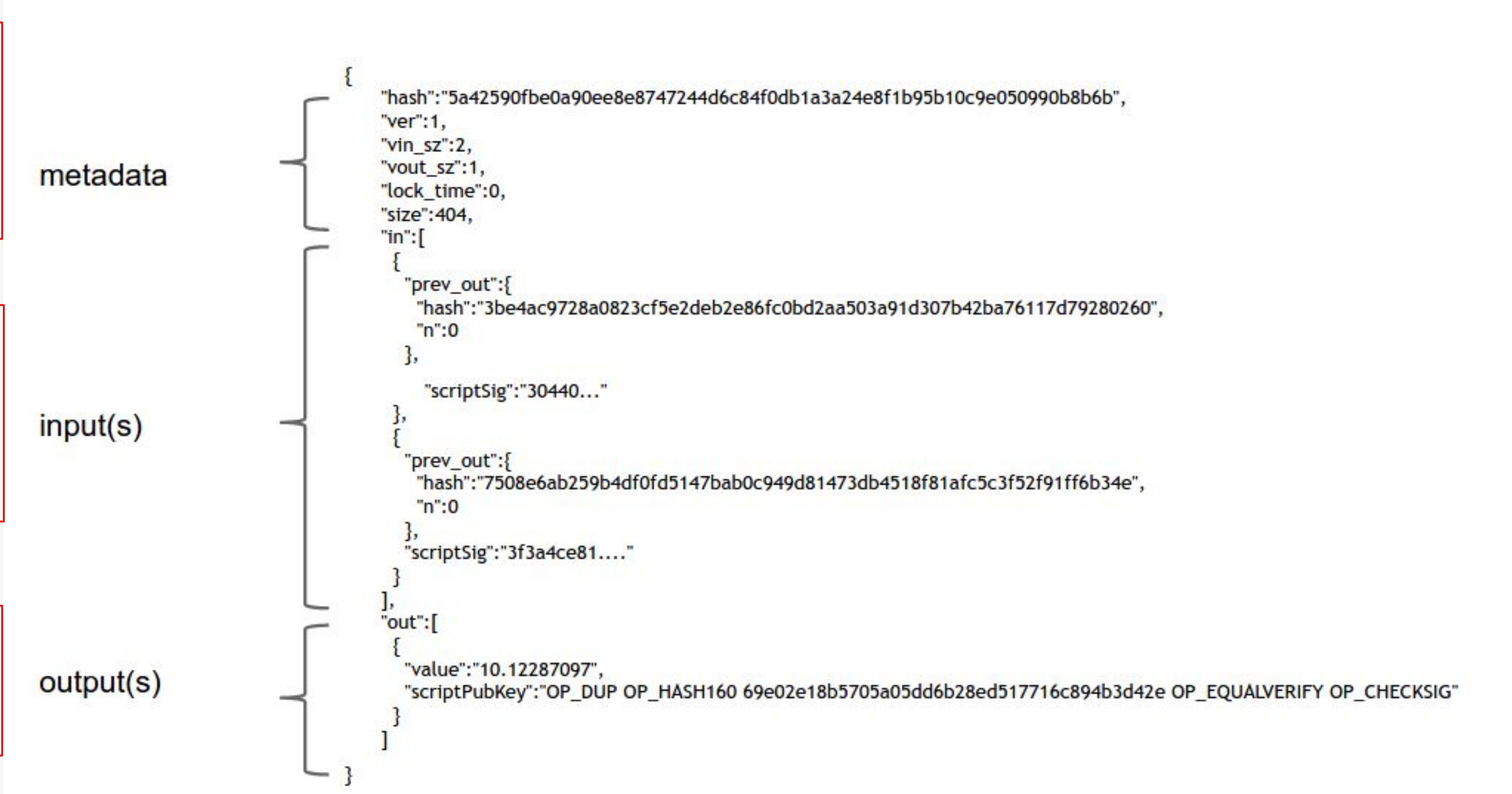


Transaction

It contains the size of the transaction, the number of inputs and outputs, the version and a lock-time. The hash is the transaction ID (TxID) discussed later

An array of all inputs. Each input contains the previous transaction hash (TxID) and the index of Txout. Also a signature *script (scriptSig)* is provided.

An array of all outputs. One output has two fields: the amount of the transferred coins and the scriptPubKey.

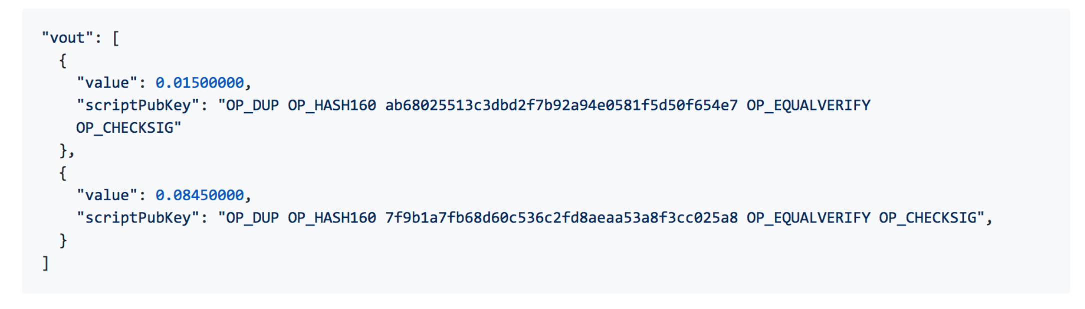

metadata
input(s)
output(s)

```
{
    "hash":"5a42590fbe0a90ee8e8747244d6c84f0db1a3a24e8f1b95b10c9e050990b8b6b",
    "ver":1,
    "vin_sz":2,
    "vout_sz":1,
    "lock_time":0,
    "size":404,
    "in":[
     {
      "prev_out":{
       "hash":"3be4ac9728a0823cf5e2deb2e86fc0bd2aa503a91d307b42ba76117d79280260",
       "n":0
      },

        "scriptSig":"30440..."
     },
     {
      "prev_out":{
       "hash":"7508e6ab259b4df0fd5147bab0c949d81473db4518f81afc5c3f52f91ff6b34e",
       "n":0
      },
      "scriptSig":"3f3a4ce81...."
     }
    ],
    "out":[
     {
      "value":"10.12287097",
      "scriptPubKey":"OP_DUP OP_HASH160 69e02e18b5705a05dd6b28ed517716c894b3d42e OP_EQUALVERIFY OP_CHECKSIG"
     }
    ]
}
```

Transaction output

- An output contains instructions for sending bitcoins
  - Value is the number of Satoshi (1 BTC = 100,000,000 Satoshi) that this output will be worth when claimed
- There can be more than one output, and they share the combined value of the inputs
- If the input is worth 50 BTC but you only want to send 25 BTC,
  - Bitcoin will create two outputs worth 25 BTC: one to the receiver, and one back to you (known as "*change*", though you send it to yourself)
- Any input bitcoins not redeemed in an output is considered a *transaction fee*; whoever generates the block will get it



```
"vout": [
  {
    "value": 0.01500000,
    "scriptPubKey": "OP_DUP OP_HASH160 ab68025513c3dbd2f7b92a94e0581f5d50f654e7 OP_EQUALVERIFY
    OP_CHECKSIG"
  },
  {
    "value": 0.08450000,
    "scriptPubKey": "OP_DUP OP_HASH160 7f9b1a7fb68d60c536c2fd8aeaa53a8f3cc025a8 OP_EQUALVERIFY OP_CHECKSIG",
  }
]
```

https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch06.asciidoc

THANK YOU
QUESTIONS?
memegenerator.net