Decentralized Systems

Bitcoin

Huge expectations…

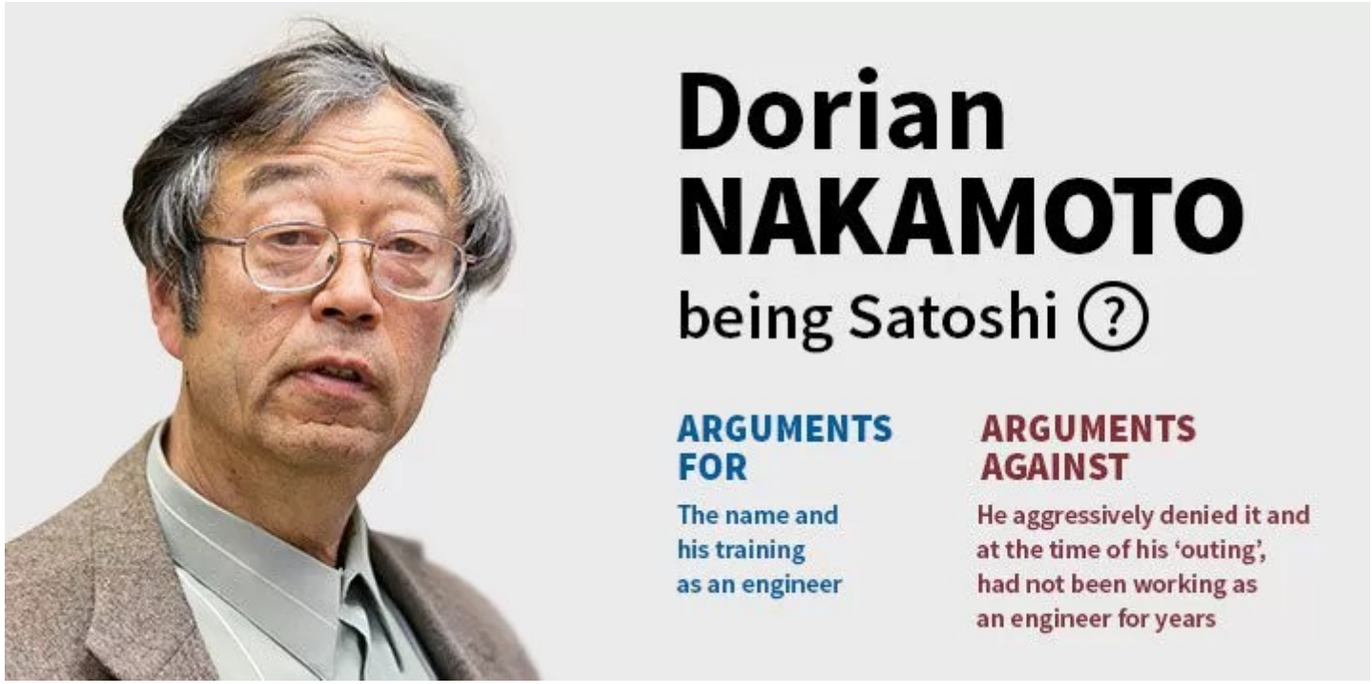
As the Internet transformed and commoditized how society communicates, blockchain will transform and commoditize how society agrees, trusts, and exchange values. How did this begin?

2008 White Paper

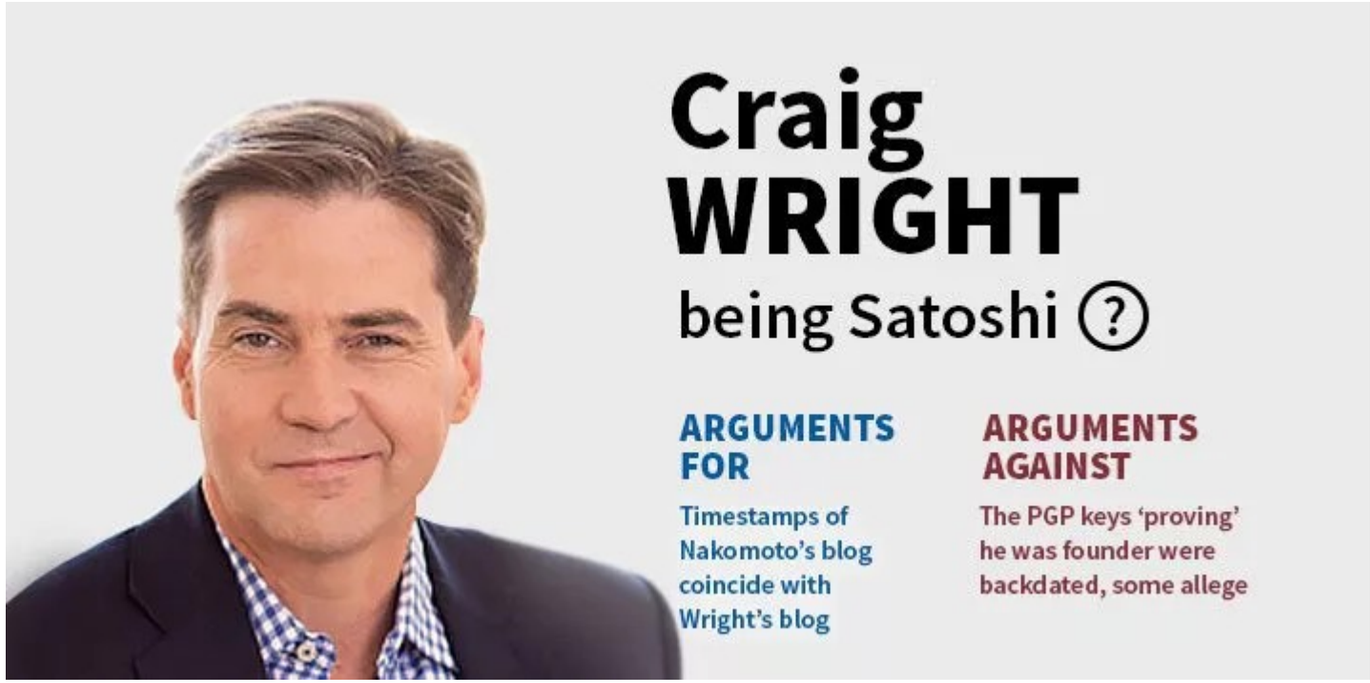
Bitcoin: A Peer-to-Peer Electronic Cash System

By: Satoshi Nakamoto

Who is Satoshi?



Who is Satoshi?



Who is Satoshi?

https://cointelegraph.com/news/from-dorian-nakamoto-to-elon-musk-the-incomplete-list-of-people-speculated-to-be-satoshi-nakamoto

Who is Satoshi?

Main page
Contents
Current events
Random article
About Wikipedia
Contact us
Donate

Contribute

Help
Learn to edit
Community portal
Recent changes
Upload file

Tools

What links here
Related changes
Special pages
Permanent link
Page information
Cite this page
Wikidata item

Print/export

Hal Finney (computer scientist)

From Wikipedia, the free encyclopedia

Harold Thomas Finney II (May 4, 1956 – August 28, 2014) was a developer for PGP Corporation, and was the second developer hired after Phil Zimmermann. In his early career, he was credited as lead developer on several console games. He also was an early bitcoin contributor and received the first bitcoin transaction from bitcoin's creator Satoshi Nakamoto.[1]

Contents [hide]

1 Early life and education
2 Career
3 Bitcoin
4 Personal life
5 Death
6 References
7 External links

Early life and education [edit]

Hal Finney

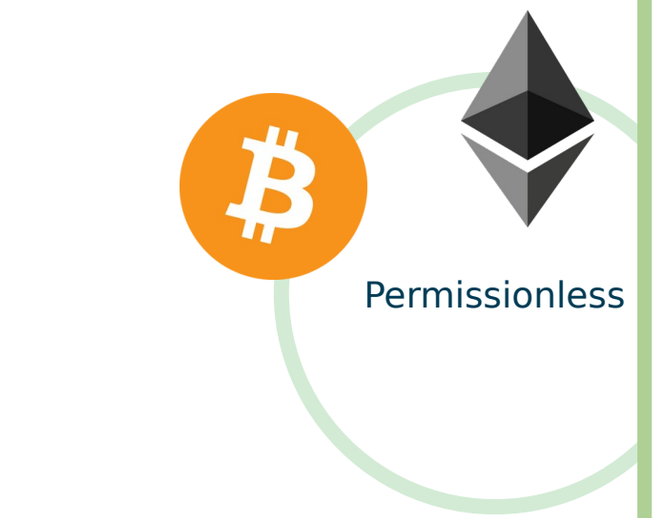
Born	Harold Thomas Finney II May 4, 1956

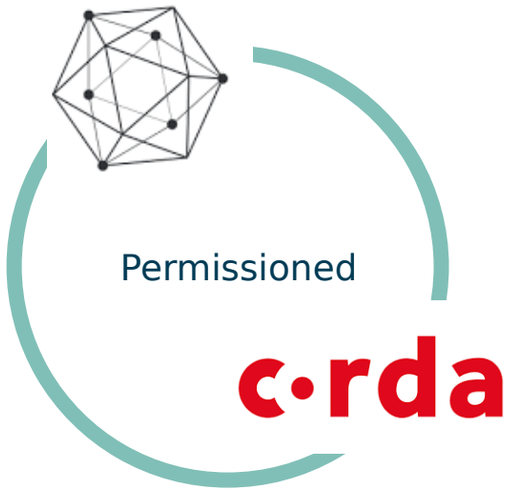
What is Blockchain?

- Introduced in 2008/2009 by **Satoshi Nakamoto**
- **Replicated linked list of blocks** that are **secured through cryptography**
- This list is **maintained by a large number of nodes** to tolerate malicious behaviors of a small group of nodes
- It is also called **Distributed Ledger**

Different types of blockchains

Different types of blockchains



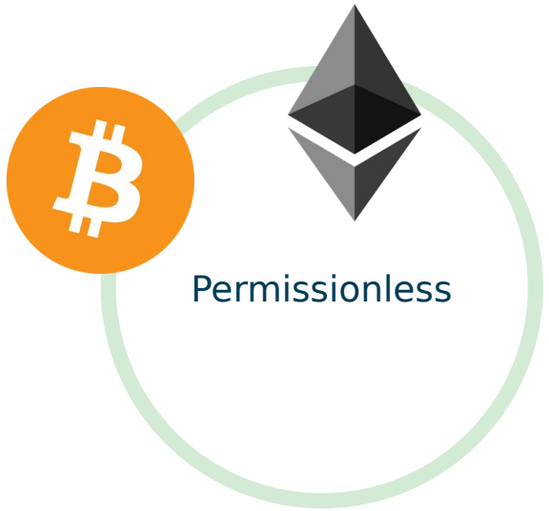


Open and decentralized
Any peer (node) can join and leave the network as writer or reader at any time

No central “entity”

The content (e.g., the transactions) is readable to any peer

Different types of blockchains



Only authorize a limited set of readers and writers

A central entity decides and attributes rights to peers

Can be public or private

What is Bitcoin?

- Bitcoin (BTC) is the most famous and used cryptocurrency
- **First** open-source cryptocurrency
- More **gold** than coins

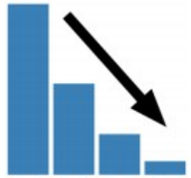
What is Bitcoin?

- Bitcoin is a worldwide payment system that **does not require intermediaries or banks**

Peer-to-peer transactions

No need for third parties

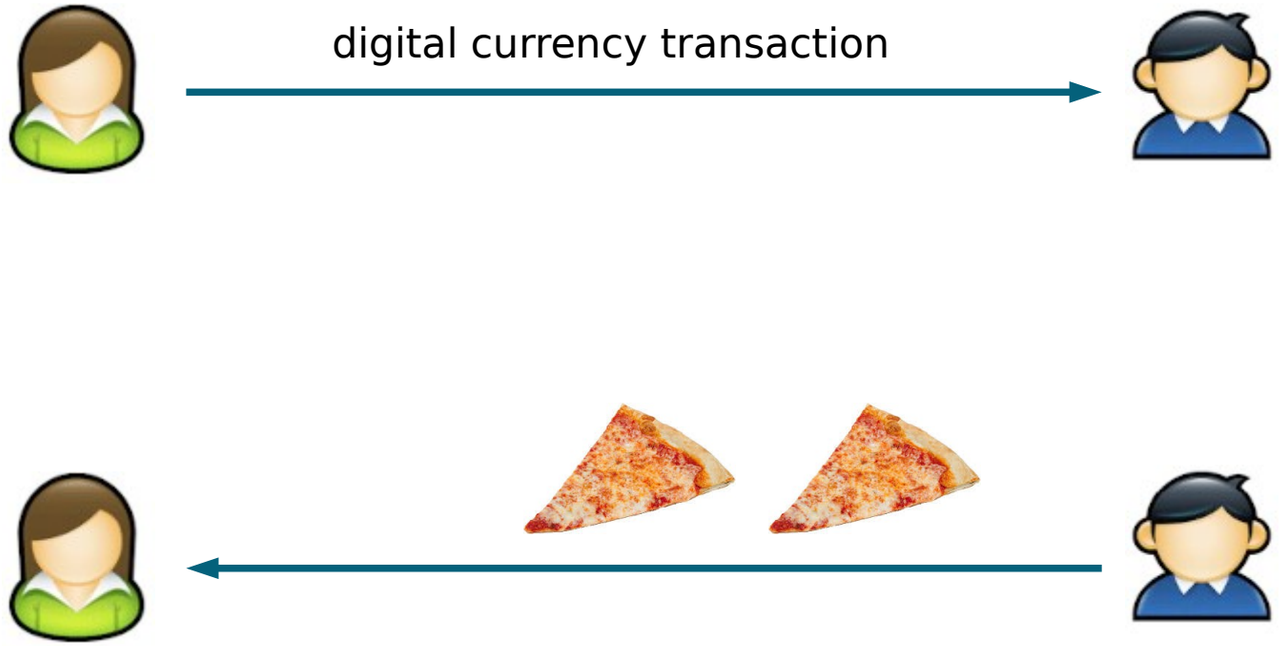
Worldwide payments



Low processing fees

Pizza for Bitcoin

- (May 2010, 2 pizzas paid 10.000 BTC)



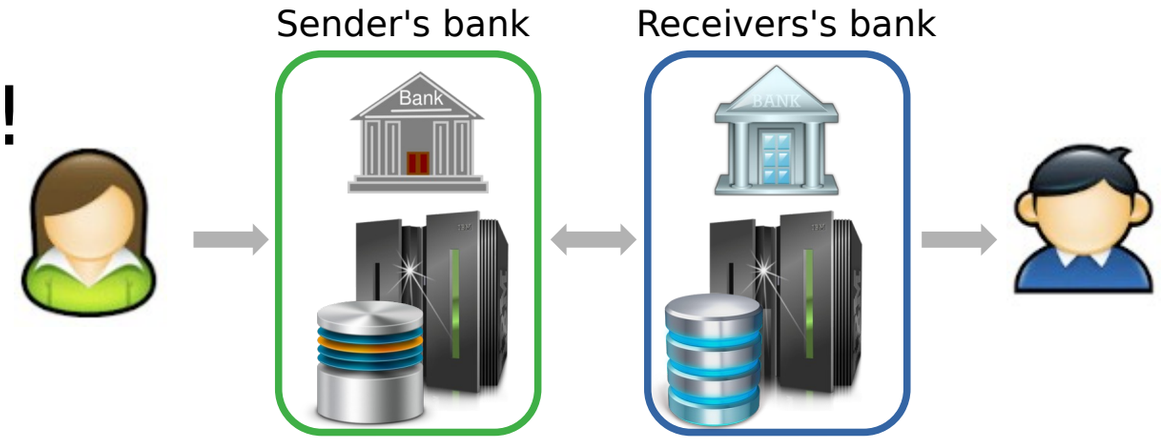
https://www.youtube.com/watch?v=A_3kHupuuxQ

Pizza for Bitcoin, risks

- Payment requires the use of hardware/software devices
  - bartender and customer are **not able to visually check the monetary transaction**
- In addition, currency virtualization introduces the problem of **double spending**
  - the owner of a digital money could make a copy of it and try to spend it two (or more) times

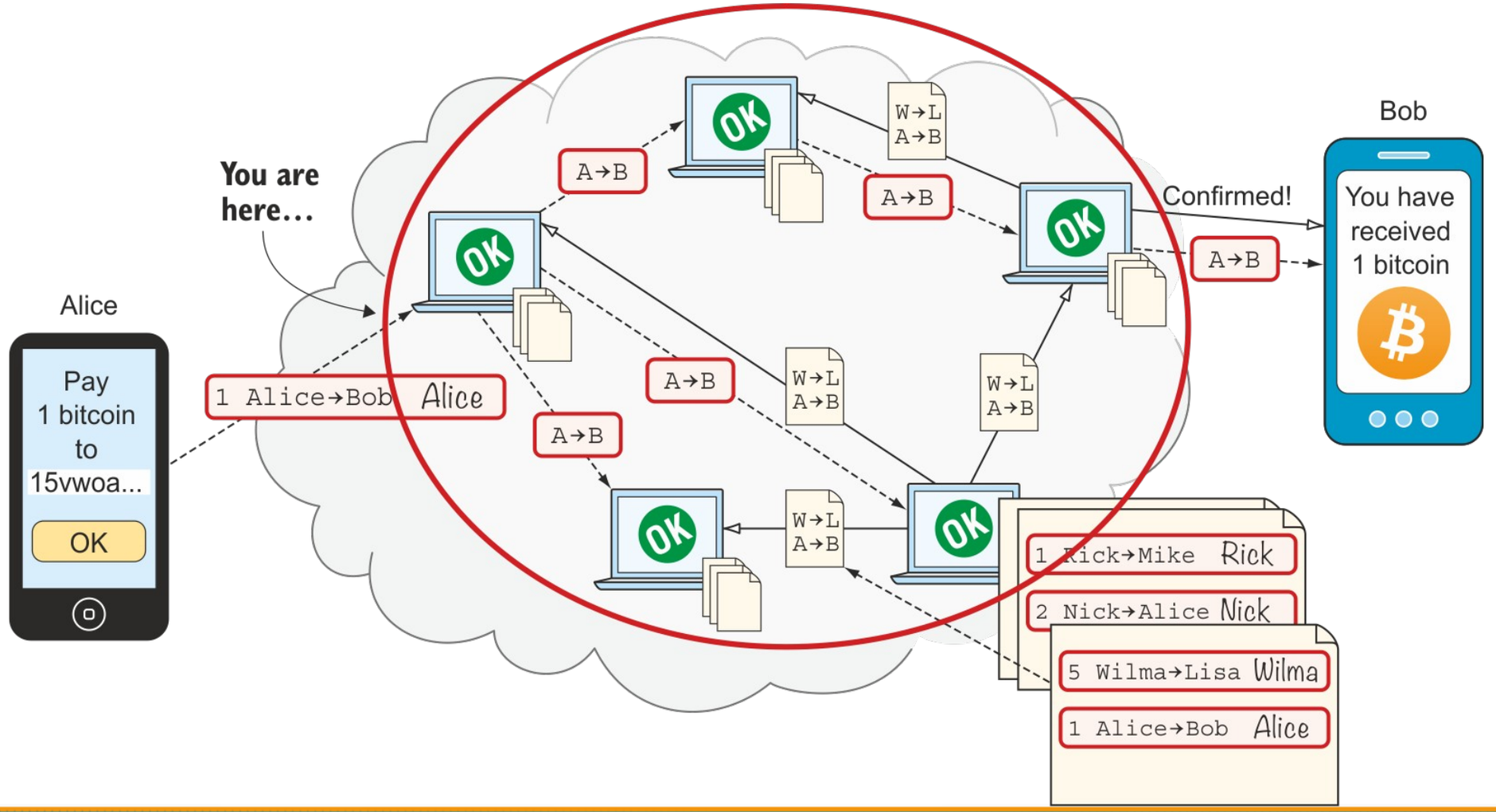
In case of bank transfer

- The transaction is managed by **a few trusted intermediaries**
  - Banks
  - Financial agencies
  - Lenders
- We need to trust banks!



Bitcoin ecosystem

- No intermediaries, but **nodes** in a **P2P network**



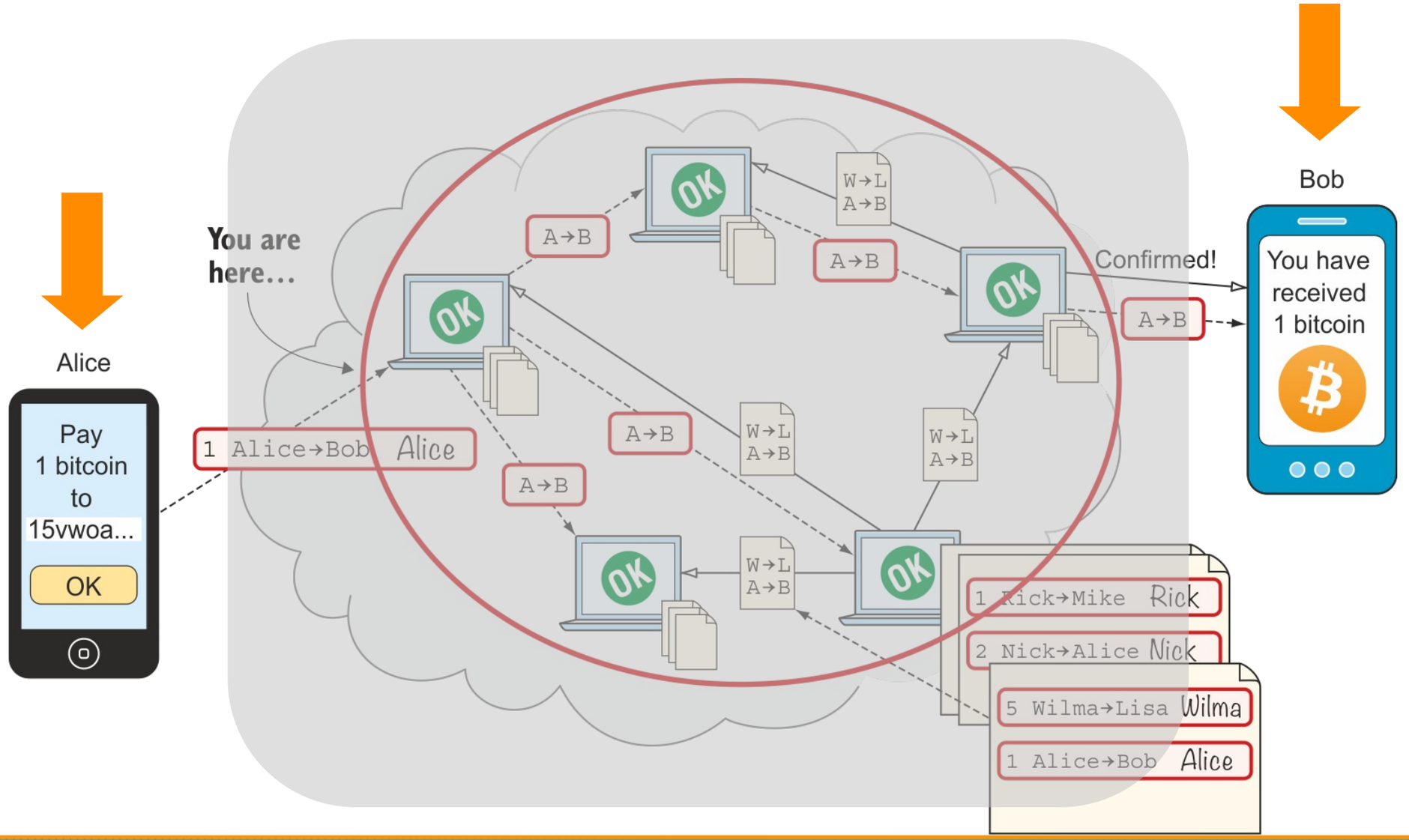
Nodes (or peers)

- Peers are **heterogeneous**, presenting difference in hardware and software
- Many peers **store data about the blockchain**
  - **Full blockchain**
  - **Pruned blockchain** (up-to-date version of the blockchain up to a few days, plus metadata about all known blocks and unspent transactions)
  - **Simplified payments verification** (SPV) clients (up-to-date version of blockchain headers, usually deployed in mobile devices)

Nodes (or peers)

- Peers can be classified for their **functionalities**
  - Wallet
  - Miner
  - Validation and relaying
  - Others: DNS, block explorer, exchange service,..
- A peer may perform more functionalities at the same time

Wallet


Alice
Pay
1 bitcoin
to
15vwoa...
OK
You are
here...
1 Alice→Bob Alice
A→B
A→B
A→B
A→B
W→L
A→B
W→L
A→B
W→L
A→B
W→L
A→B
OK
OK
OK
OK
OK
Confirmed!
A→B
Bob
You have
received
1 bitcoin
1 Rick→Mike Rick
2 Nick→Alice Nick
5 Wilma→Lisa Wilma
1 Alice→Bob Alice

Wallet

- To become a node of type wallet we need
  - an Internet connection
  - a Bitcoin wallet (software program or hardware)
- A Bitcoin wallet can be compared to the **user's bank account** and it is thus necessary for sending and receiving Bitcoins
- It is identified by **two mathematically-connected keys**
  - a private key
  - a public key

Private key

- The **secret key (Sk)** is a **256-bit** randomly generated number which must meet a specific format

Private key

- The **secret key (Sk)** is a **256-bit** randomly generated number which must meet a specific format

64 characters in the range 0-9 or A-F

Private key

- The **secret key (Sk)** is just a **big random number** in **hexadecimal format**

ef235aacf90d9f4aadd8c92e4b2562e1
d9eb97f0df9ba3b508258739cb013db2

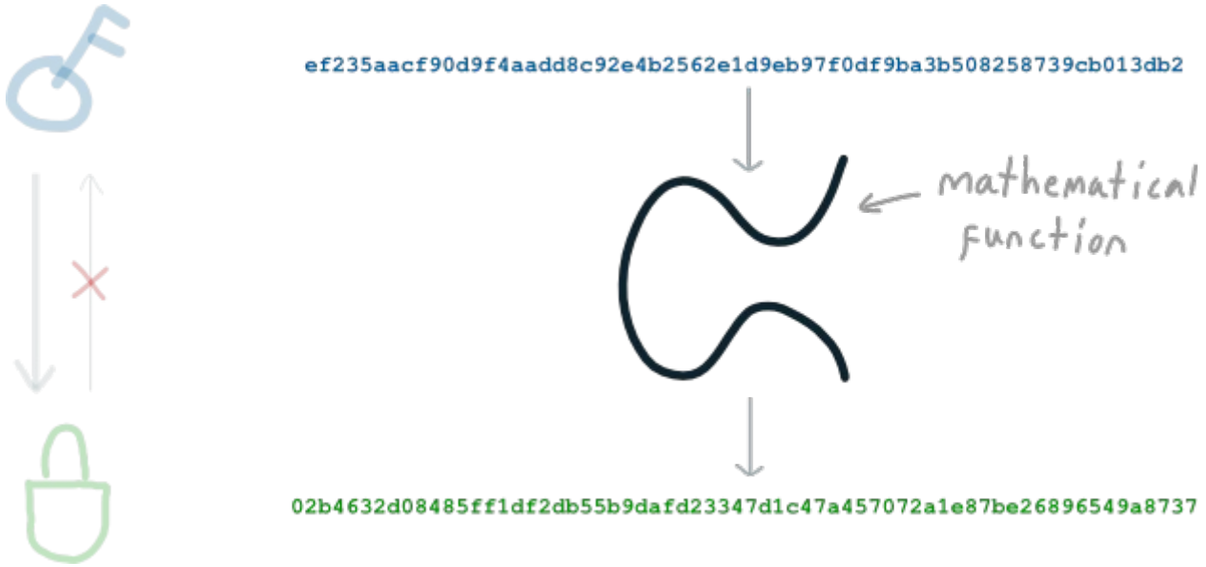
- It is used to **authorize Bitcoin transfers** and it can be thought of as a **password** or the **secret PIN** of a card associated with the bank account

Public key

- The **public key (Pk)**
  - is **calculated from the private key**, by means of an algorithm established by the Bitcoin protocol
  - is a **public data** and it can be thought of as the public information of a bank account
- The private key should not be easily recoverable from the public key
- Pk is used for **transaction verification** and **validity check**

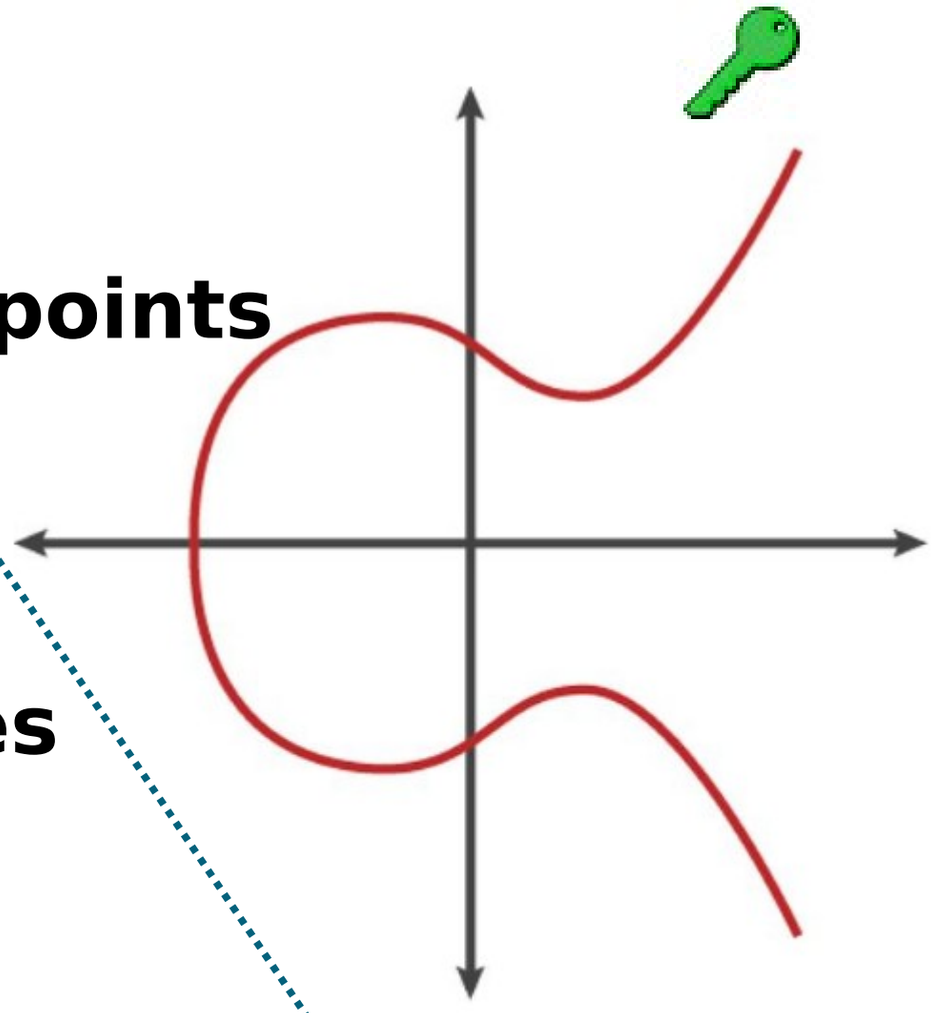
Public key

- The Bitcoin protocol uses **elliptic curve cryptography** (ECC) to compute keys for digital signatures (ECDSA)



ECC

- An elliptic curve is the **set of points** described by the equation
- Depending on the values of a and b, elliptic curves may assume **different shapes**



$$4a^3 + 27b^2 \neq 0$$

Condition on a and b

$$y^2 = x^3 + ax + b$$

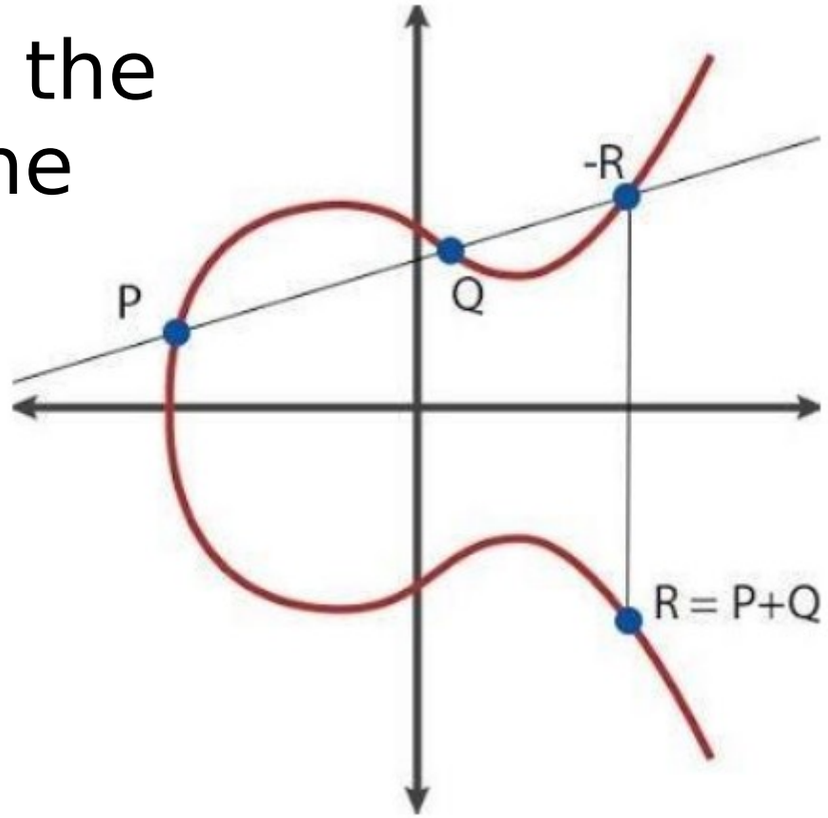
Equation of a generic elliptic curve

ECC

- Elliptic curves are **symmetric on the x-axis**
- Form a **group**, e.g., the elements of the group are all the points on the curve
- The **inverse** of a point is the one symmetric on the x-axis

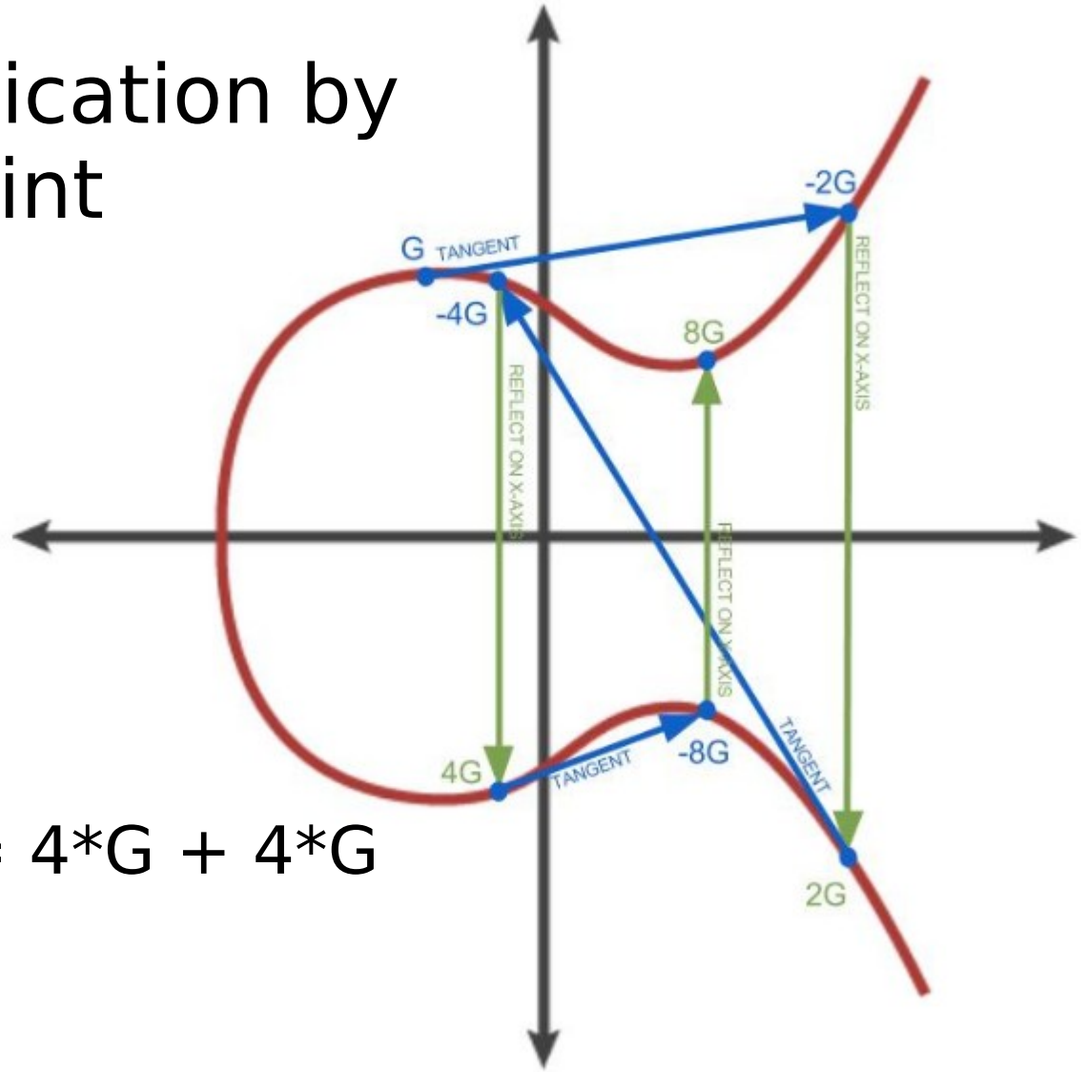
ECC: sum

- The sum of two points on the curve is a third point of the curve
- P + Q = ?
  - Find -R
  - Reflect on the x axis



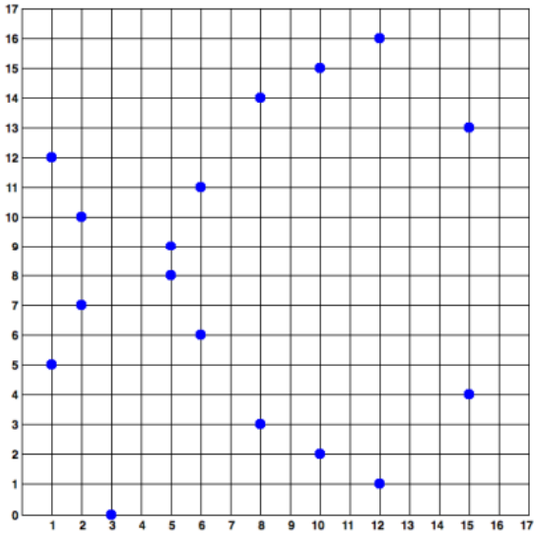
ECC: scalar multiplication

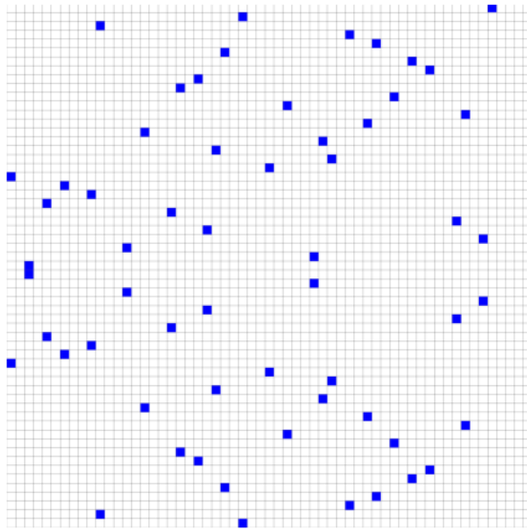
- Given a point G, its multiplication by a scalar value returns a point on the curve
- 2*G = G + G
  - Take the **tangent** in G
  - Reflect on x axis
- Same for 4*G= 2*G + 2*G, 8*G= 4*G + 4*G



ECC: scalar multiplication

- Given a known **base point G**, it can be multiplied by a private key **Sk** to find the corresponding public key **Pk**
- **Modular arithmetic**
  - **Pk =** (**Sk** * **G**) mod **p** (with p prime number)





https://www.youtube.com/watch?v=qCafMW4OG7s

Bitcoin Secp256k1

- Specific elliptic curve: $y^2 = (x^3 + 7) \bmod p$
  - **a** = 0, **b** = 7
  - **p** = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFC2F

    $= 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$
  - **G** = (0x79be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f81798, 0x483ada7726a3c4655da4fbfc0e1108a8fd17b448a68554199c47d08f5b166e6c)
  - **Sk** is 256 bit
- How secure is 256 bit security? https://www.youtube.com/watch?v=S9JGmA5_unY

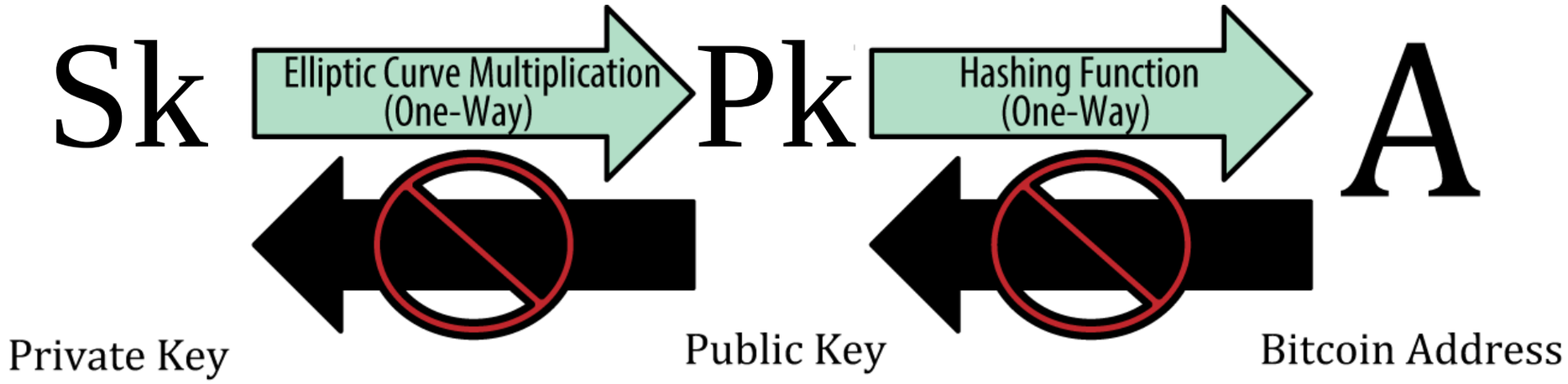
Bitcoin address

- The **Bitcoin address A** is a sequence of digits and letters computed from the public key* with 2 hash functions
  A = Base58Check(**RIPEMD160**(**SHA256**(Pk)))
  - it is encoded in **Base58**, to avoid I,l,O,0
    https://tools.ietf.org/id/draft-msporny-base58-01.html
  - it starts with **1** (legacy), **3** (multi-signature transactions) or **bc1** (SegWit transactions, using Bench32 encoding)
- It serves as **identifier of the wallet**
  - it can be seen as the **IBAN**, needs to **be shared** for receiving payments

* things are more complex

Wallet

- Any Bitcoin wallet is associated and identified with (at least) **three numbers**



Source: https://www.oreilly.com/library/view/mastering-bitcoin-2nd/9781491954379/ch04.html

Wallet

- If the **private key Sk** of a Bitcoin wallet **is lost**, it cannot be recovered and the crypto associated with that private key are lost as well!

- And this happens…

https://news.bitcoin.com/analyst-1500-bitcoins-lost-every-day-less-than-14-million-coins-will-ever-circulate/

Wallet

- Stores keys, not Bitcoin!
  - Bitcoins are transactions on the blockchain
- It can be
  - **Custodial**, e.g., provided by third parties (exchanges) which have control over private keys (and therefore, over crypto)
  - **Non custodial**, e.g., the owner is the only one with access to their private keys, and therefore, has complete control over their assets

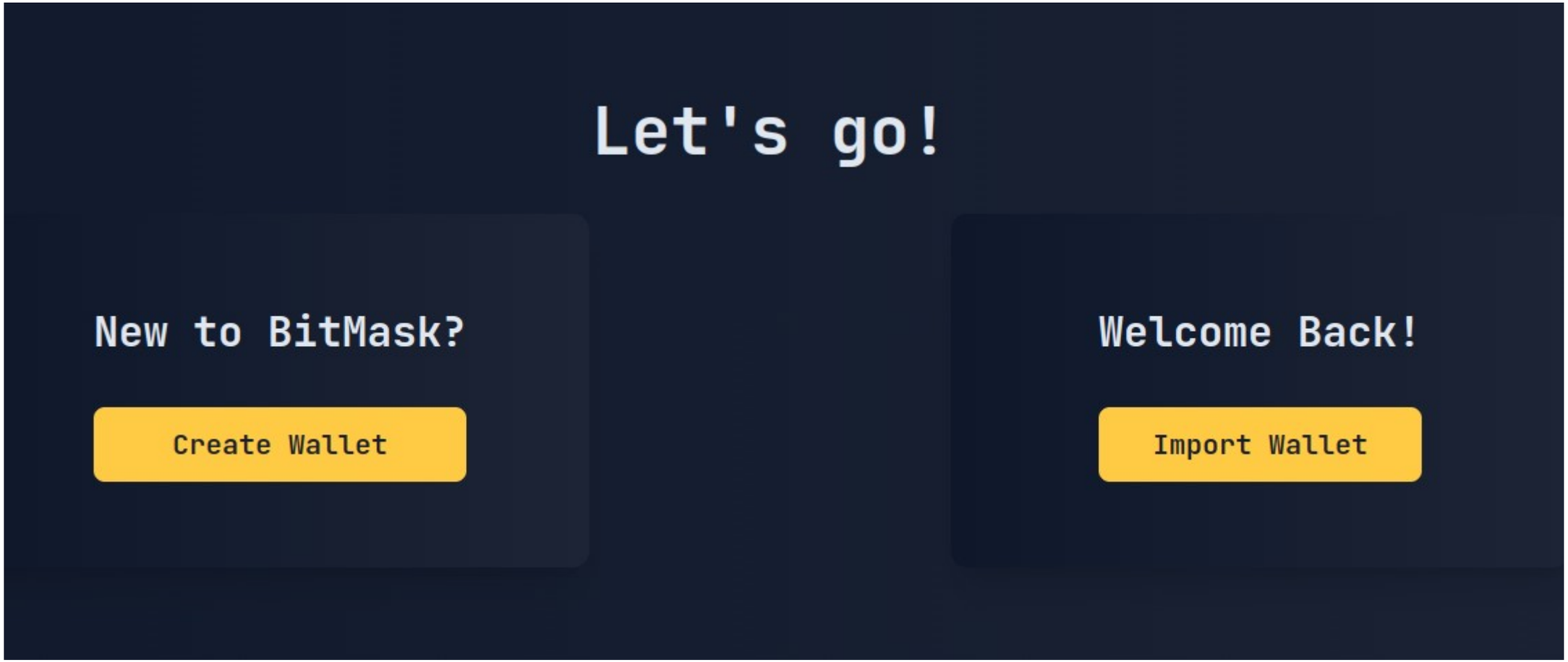
Pros	Cons
Less responsibility held by users	Private keys are controlled by third party
Simple and easy to use for beginners	Custodial wallets are vulnerable to hackers
Can reset password to regain access to digital assets	KYC and AML verification for account creation
	Less advanced features available for experienced crypto users

Custodial

Pros	Cons
You control your keys	Impossible to recover digital assets if users lose private keys and/or recovery phrases
Fast and easy to create new wallets	More technical knowhow needed to use advanced features
Funds won't be impacted in cases of exchange hacks	
No KYC or AML process necessary for creating/storing	
More advanced functions and features available than custodial services	

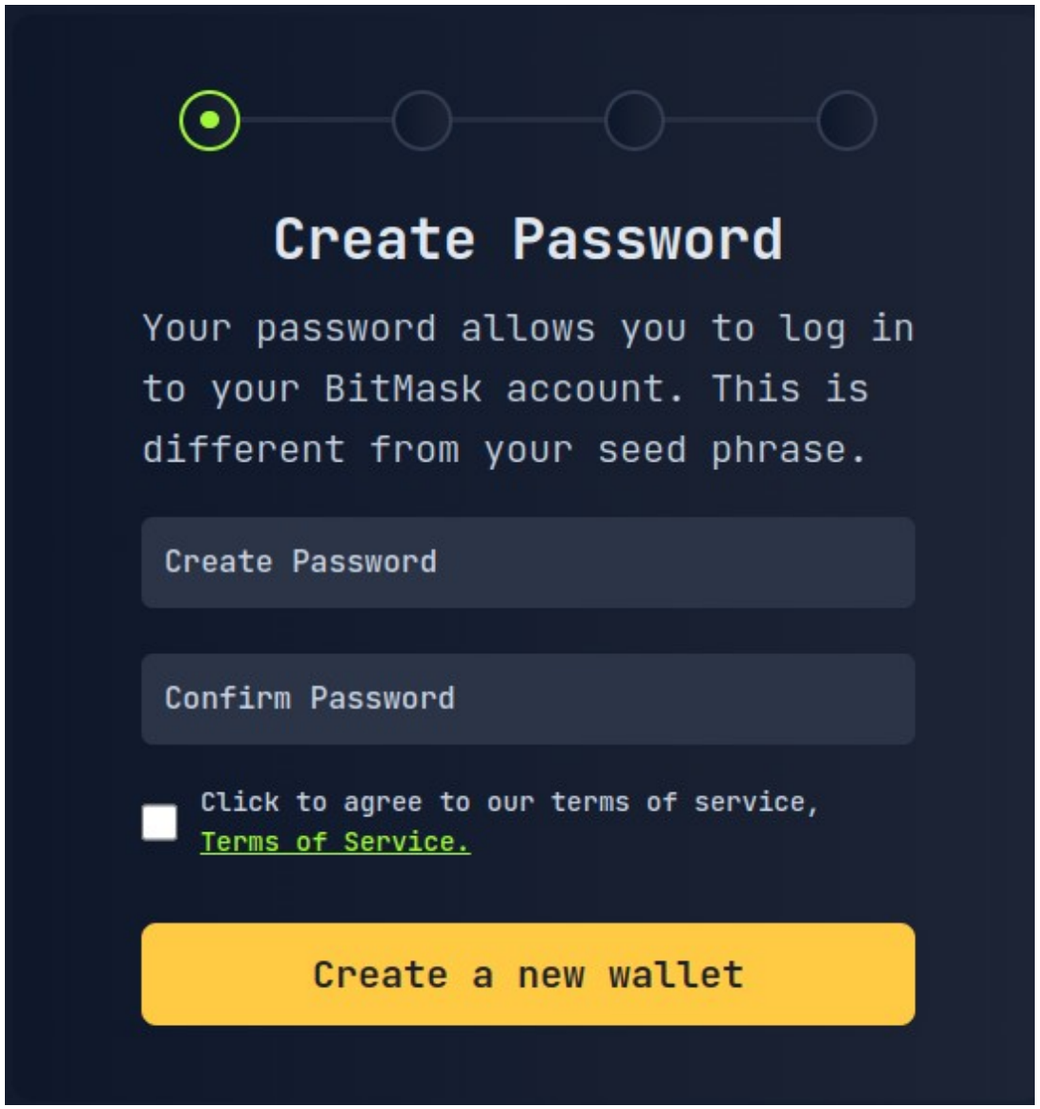
Non custodial

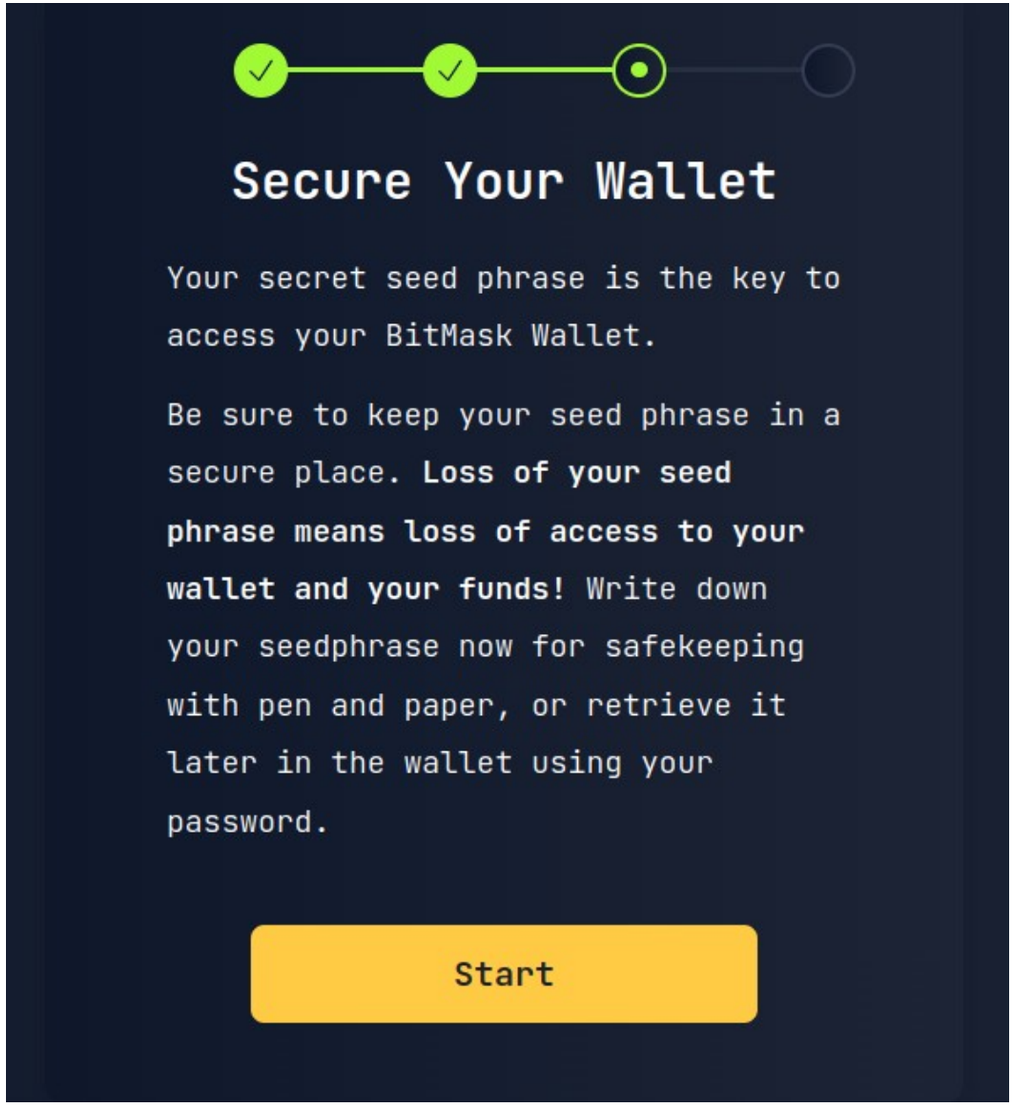
BitMask



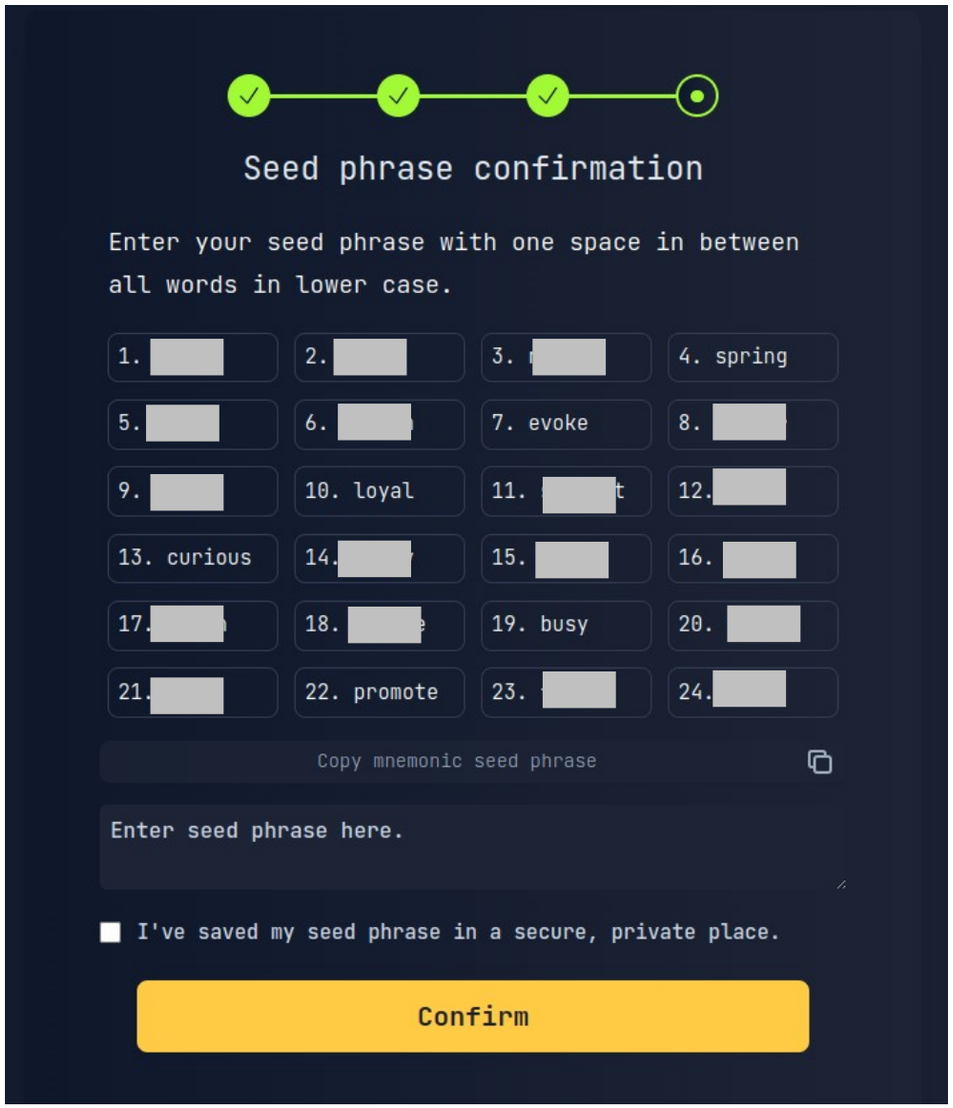
Note: We will use **MetaMask** (Ethereum wallet)

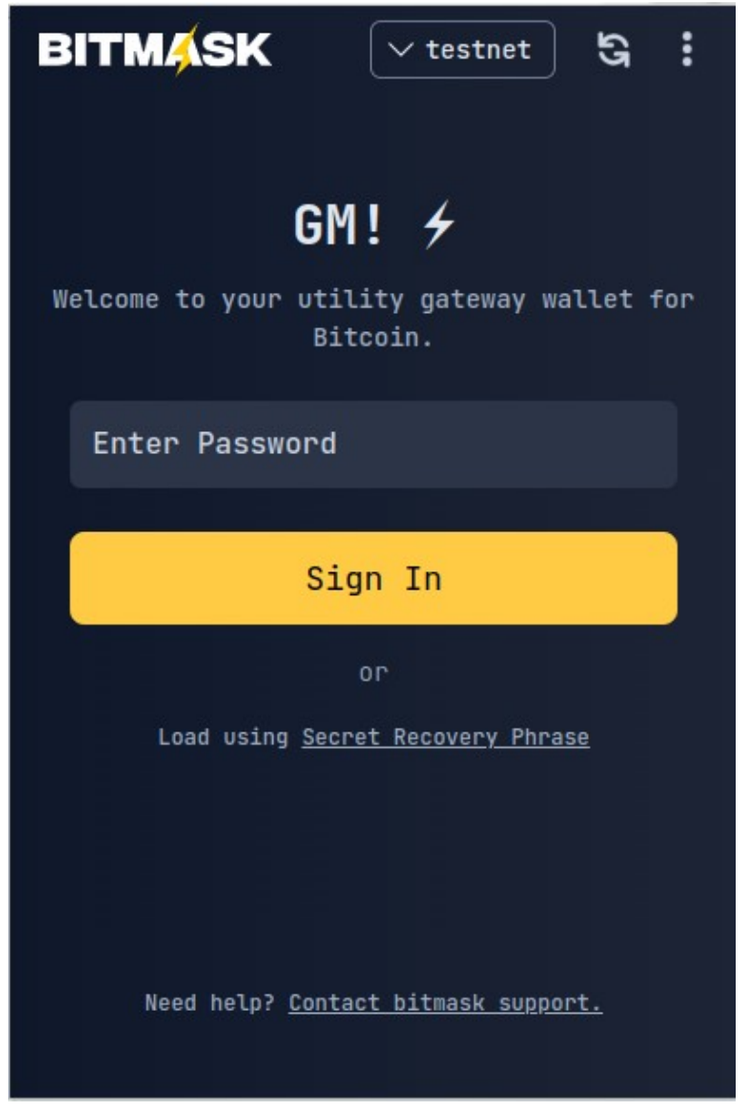
BitMask


Create Password
Your password allows you to log in to your BitMask account. This is different from your seed phrase.
Create Password
Confirm Password
Click to agree to our terms of service,
Terms of Service.
Create a new wallet


Secure Your Wallet
Your secret seed phrase is the key to access your BitMask Wallet.
Be sure to keep your seed phrase in a secure place. Loss of your seed phrase means loss of access to your wallet and your funds! Write down your seedphrase now for safekeeping with pen and paper, or retrieve it later in the wallet using your password.
Start

BitMask

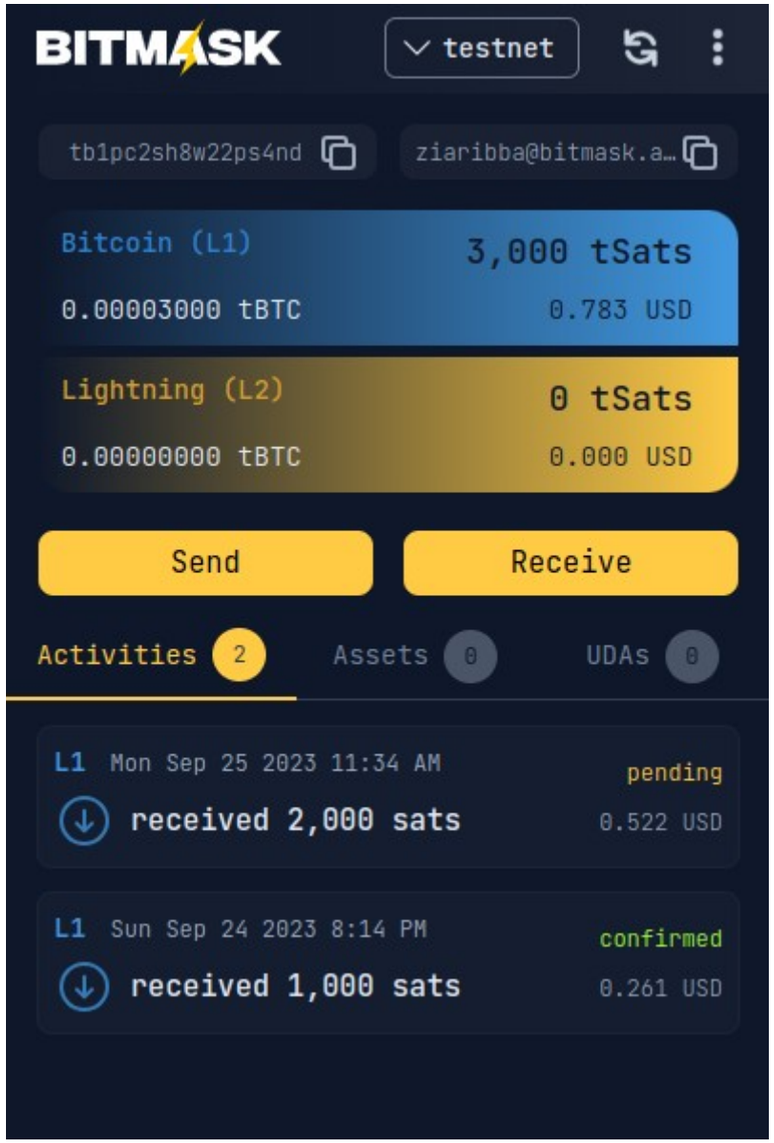

Seed phrase confirmation
Enter your seed phrase with one space in between all words in lower case.
1.
2.
3.
4. spring
5.
6.
7. evoke
8.
9.
10. loyal
11.
12.
13. curious
14.
15.
16.
17.
18.
19. busy
20.
21.
22. promote
23.
24.
Copy mnemonic seed phrase
Enter seed phrase here.
I've saved my seed phrase in a secure, private place.
Confirm


BITMASK
testnet
GM!
Welcome to your utility gateway wallet for Bitcoin.
Enter Password
Sign In
or
Load using Secret Recovery Phrase
Need help? Contact bitmask support.

Faucet

- Website or application to get fake cryptocurrency for testnet https://bitcoinfaucet.uo1.net/

- Sometimes they award registered users with small amounts of cryptocurrency in exchange for completing a simple task such as viewing an ad or participating in a short survey



Seed phrase

- Created using a specific algorithm (BIP 39) which converts a **randomly generated number** (**entropy**) into a **sequence of words** from a predetermined word list
    - **128 bits** of entropy → **12 words**
    - **256 bits** of entropy → **24 words**
- Seed phrases enable users to regenerate all private keys associated with their wallets

COMPLETE LIST OF BIP-39 SEED RECOVERY PHRASE WORDS FOR WALLETS

abandon ability able about above absent absorb abstract absurd abuse access accident account accuse achieve acid acoustic acquire across act action actor actress actual adapt add addict address adjust admit adult advance advice aerobic affair afford afraid again age agent agree ahead aim air airport aisle alarm album alcohol alert alien all alley allow almost alone alpha already also alter always amateur amazing among amount amused analyst anchor ancient anger angle angry animal ankle announce annual another answer antenna antique anxiety any apart apology appear apple approve april arch arctic area arena argue arm armed armor army around arrange arrest arrive arrow art artefact artist artwork ask aspect assault asset assist assume asthma athlete atom attack attend attitude attract auction audit august aunt author auto autumn average avocado avoid awake aware away awesome awful awkward axis baby bachelor bacon badge bag balance balcony ball bamboo banana banner bar barely bargain barrel base basic basket battle beach bean beauty because become beef before begin behave behind believe below belt bench benefit best betray better between beyond bicycle bid bike bind biology bird birth bitter black blade blame blanket blast bleak bless blind blood blossom blouse blue blur blush board boat body boil bomb bone bonus book boost border boring borrow boss bottom bounce box boy bracket brain brand brass brave bread breeze brick bridge brief bright bring brisk broccoli broken bronze broom brother brown brush bubble buddy budget buffalo build bulb bulk bullet bundle bunker burden burger burst bus business busy butter buyer buzz cabbage cabin cable cactus cage cake call calm camera camp can canal cancel candy cannon canoe canvas canyon capable capital captain car carbon card cargo carpet carry cart case cash casino castle casual cat catalog catch category cattle caught cause caution cave ceiling celery cement census century cereal certain chair chalk champion change chaos chapter charge chase chat cheap check cheese chef cherry chest chicken chief child chimney choice choose chronic chuckle chunk churn cigar cinnamon circle citizen city civil claim clap clarify claw clay clean clerk clever click client cliff climb clinic clip clock clog close cloth cloud clown club clump cluster clutch coach coast coconut code coffee coil coin collect color column combine come comfort comic common company concert conduct confirm congress connect consider control convince cook cool copper copy coral core corn correct cost cotton couch country couple course cousin cover coyote crack cradle craft cram crane crash crater crawl crazy cream credit creek crew cricket crime crisp critic crop cross crouch crowd crucial cruel cruise crumble crunch crush cry crystal cube culture cup cupboard curious current curtain curve cushion custom cute cycle dad damage damp dance danger daring dash daughter dawn day deal debate debris decade december decide decline decorate decrease deer defense define defy degree delay deliver demand demise denial dentist deny depart depend deposit depth deputy derive describe desert design desk despair destroy detail detect develop device devote diagram dial diamond diary dice diesel diet differ digital dignity dilemma dinner dinosaur direct dirt disagree discover disease dish dismiss disorder display distance divert divide divorce dizzy doctor document dog doll dolphin domain donate donkey donor door dose double dove draft dragon drama drastic draw dream dress drift drill drink drip drive drop drum dry duck dumb dune during dust dutch duty dwarf dynamic eager eagle early earn earth easily east easy echo ecology economy edge edit educate effort egg eight either elbow elder electric elegant element elephant elevator elite else embark embody embrace emerge emotion employ empower empty enable enact end endless endorse enemy energy enforce engage engine enhance enjoy enlist enough enrich enroll ensure enter entire entry envelope episode equal equip era erase erode erosion error erupt escape

Abandon is 0001.

Ability is 0002.

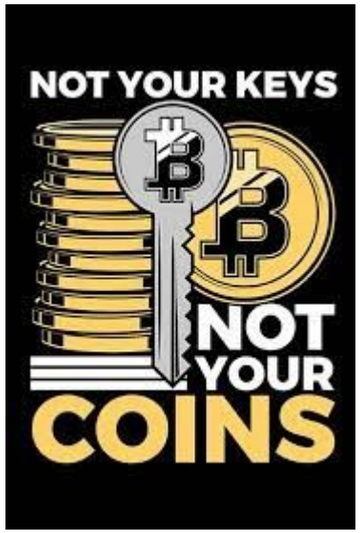
Able is 0003 etc. etc.

Seed phrase

- **Start** from the **entropy**, e.g., a sequence of random bits (128 or 256)
- **Compute hash** = SHA256(entropy)
- **Take** the first n = len(entropy) / 32 bits from **hash** (128/32=4, 256/32=8)
- **Append** these bits at the end of the entropy (checksum) (128+4=132, 256+8=264)
- **Split** bits into groups of 11 and convert them into integers (132/11=12 words, 264/11=24 words)
- **Find** the corresponding words in the word list

Seed phrase

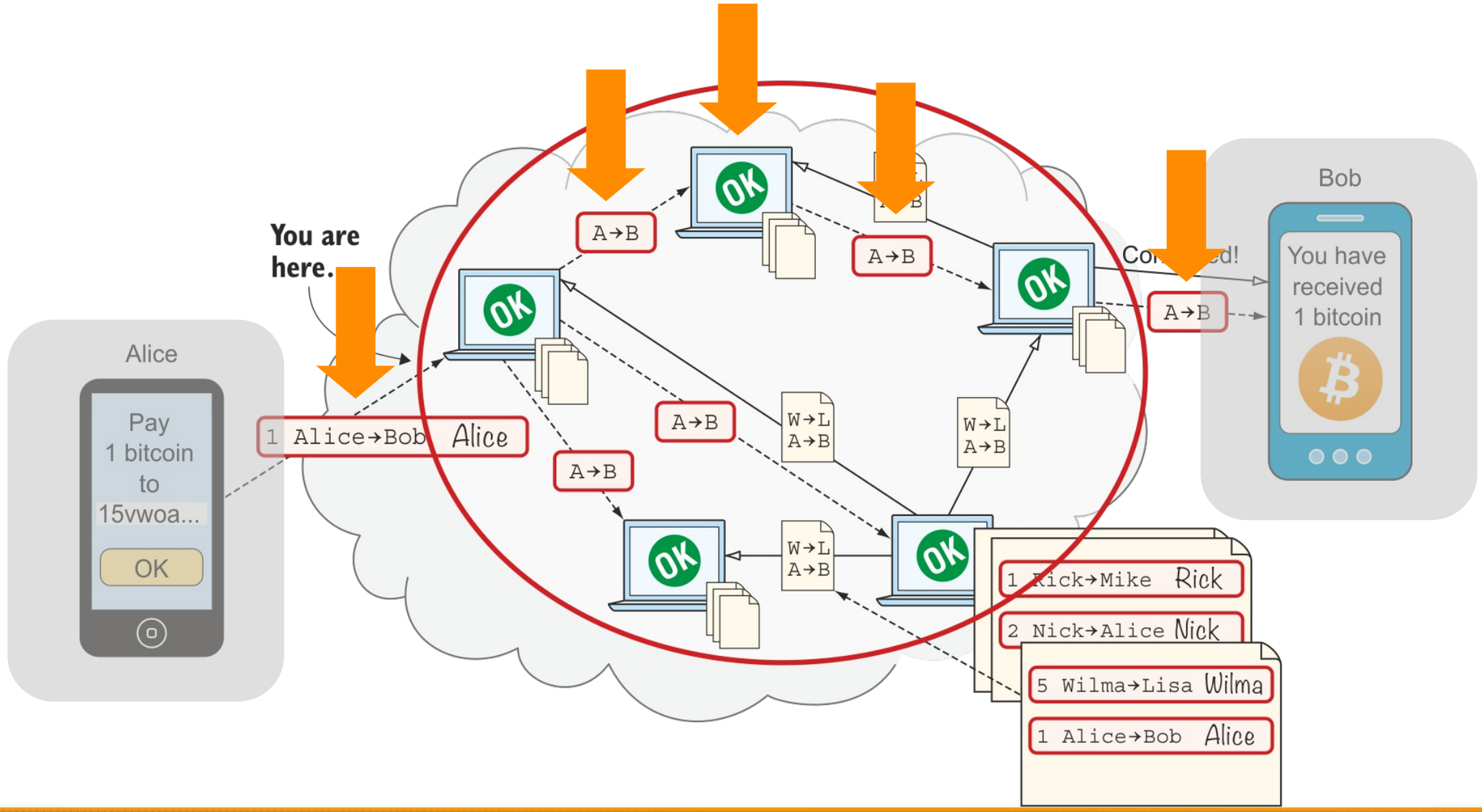
- The process is **reversible**, if you loose your password and cannot access your private key(s) you can restore the wallet starting from the seed phrase
- The seed phrase is used to recompute the starting random number (the **master private key**) which generates the rest of the other private keys



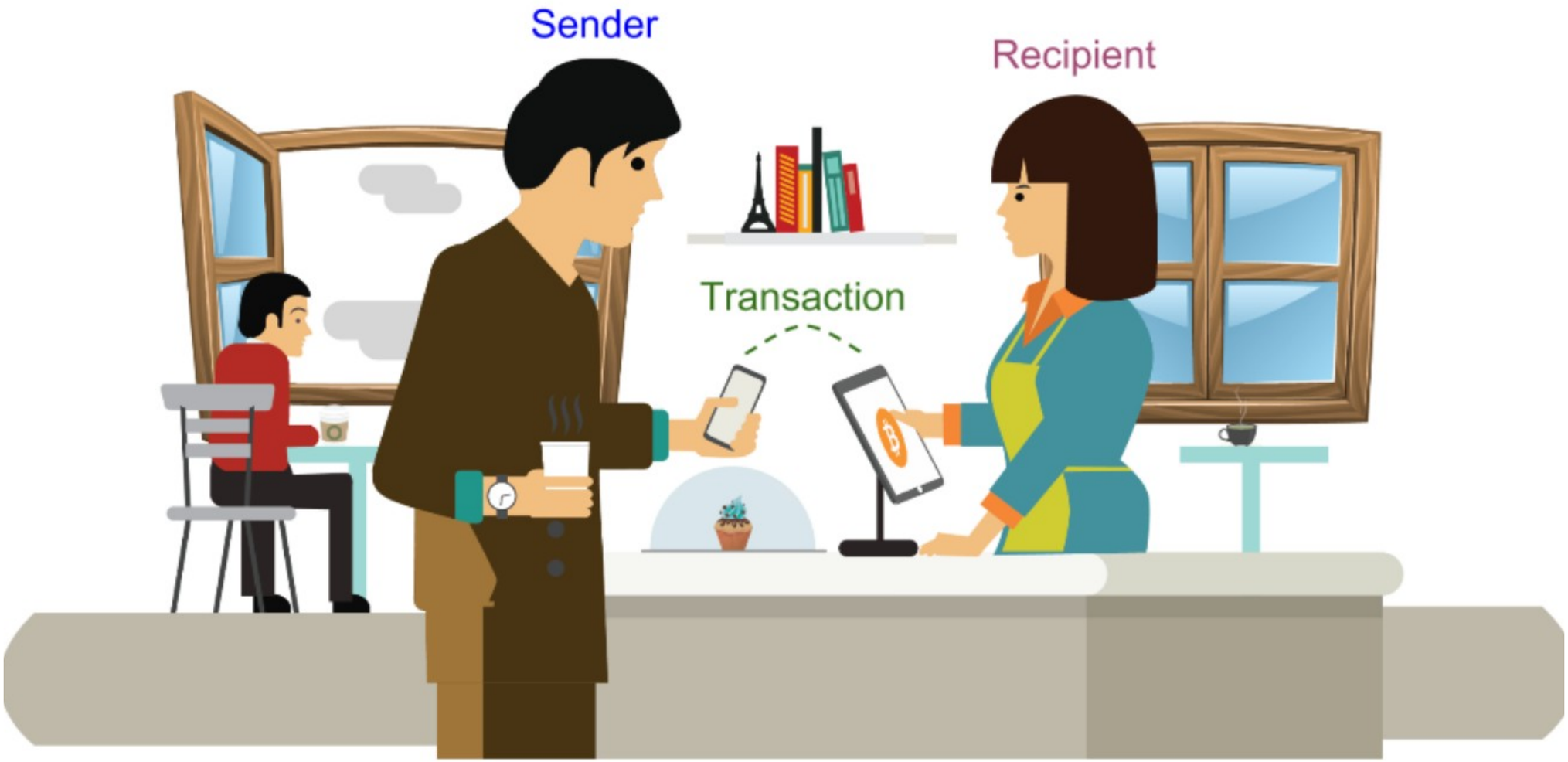
DIY (Do It Yourself)

- Install  METAMASK

Broadcast transactions



Bitcoin transaction



Bitcoin transaction

- **Payments** in the Bitcoin system are **transactions** that represent **Bitcoins movements** from source addresses (**input**) to destination addresses (**output**)
- The protocol forces input addresses to spend the exact amount of a previously received transaction
- At any given moment, an **output** may be
  - **Already spent**
  - **Unspent** transaction output (**UTXO**)

Bitcoin transaction: metadata

- Information about the transaction itself
  - creation date
  - dimension
  - transaction identification number calculated using a hash function
  - segwit flag (for segregated witness, added later)
  - See https://www.blockchain.com/explorer

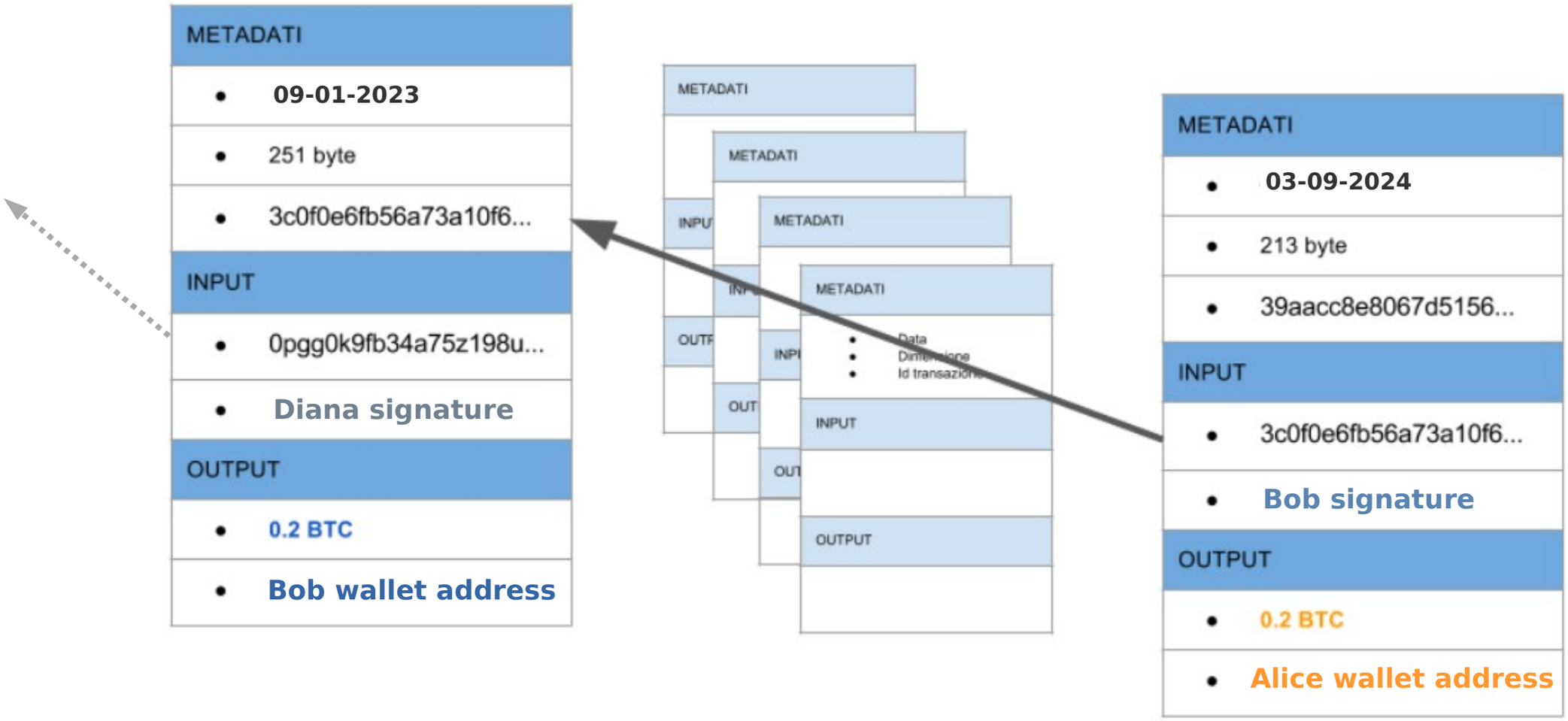
Bitcoin transaction: input

- Consists of
  - a transaction **sender**, for example Bob
  - the transaction **identification number** of a **previous transaction** with **Bob as receiver** (UTXO)
  - Bob’s **digital signature** (by means of his private key)
  - A **script** to prove Bob owns the money

Bitcoin transaction: output

- Consists of
  - the **recipient**'s Bitcoin address, for example Alice's one
  - the **number of Bitcoin/Satoshi** to be transferred to Alice
  - a **lock script**, which means that the coins can only be used as inputs in future transactions by people who can unlock them

From Bob to Alice

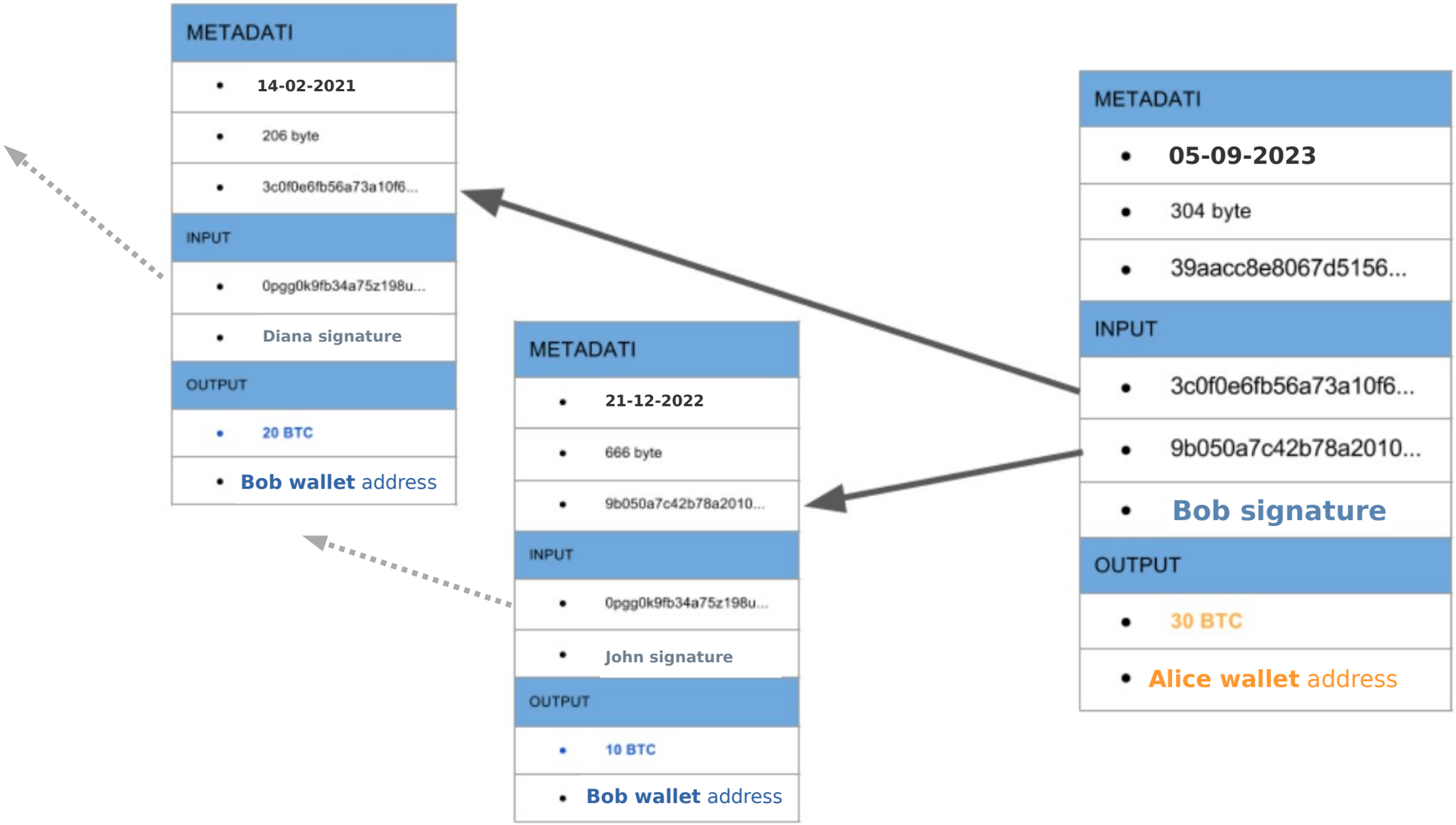


Bob can spend UTXO coming from his past history

Multiple input

- Suppose **Bob wants to pay** the amount of **30 BTC** to Alice, but he is not the recipient of any single transaction of 30 BTC
- Bob can insert into the input of the new transaction **one or more transaction IDs** he has previously received
  - one of **10 BTC**
  - one of **20 BTC**
- In the **input of a transaction**, the sender can insert a **list of transactions previously received**

Multiple input

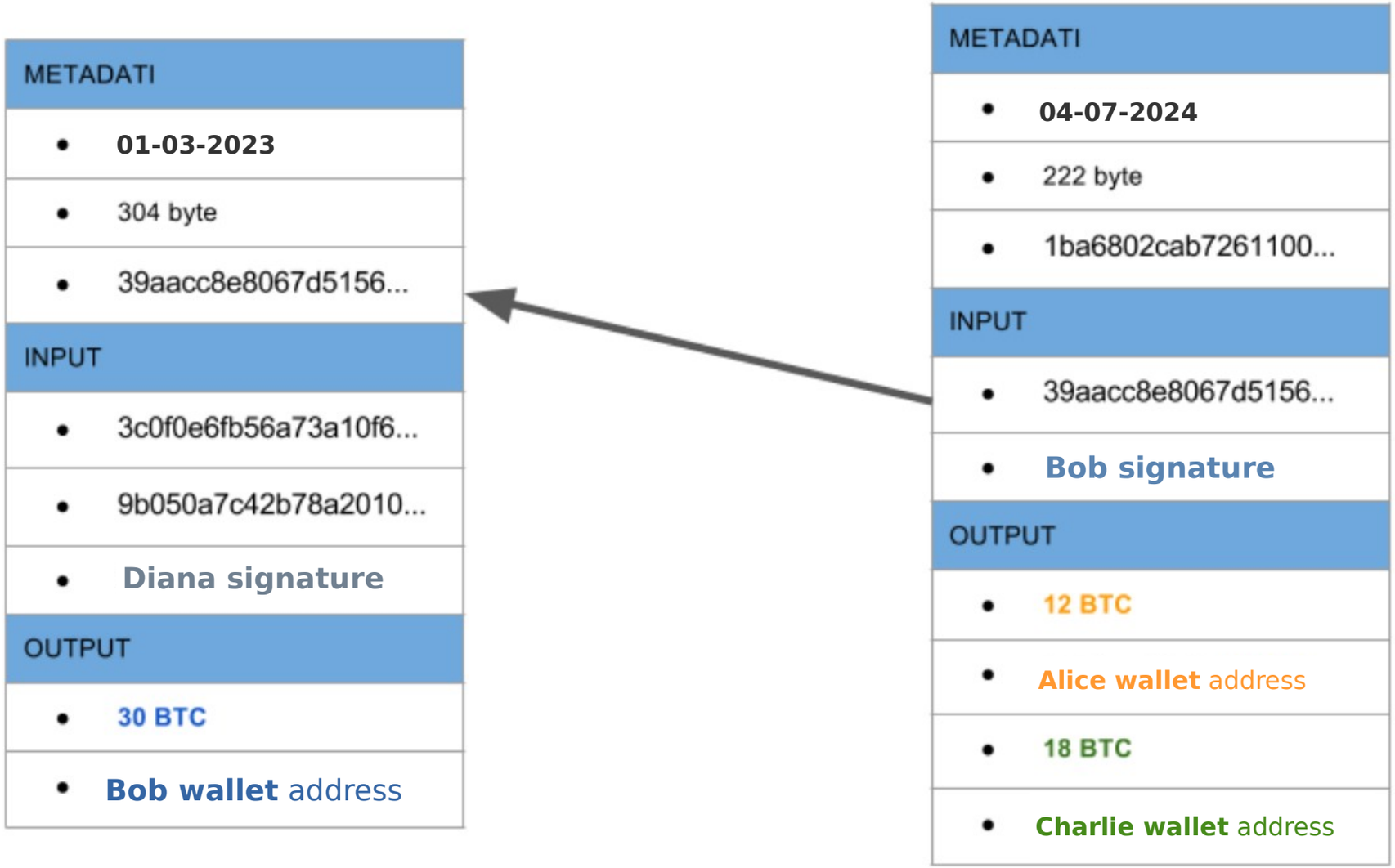


Multiple output

- Suppose **Bob wants to pay**
  - **12 BTC** to Alice
  - **18 BTC** to Charlie

  and he wants to pay using a transaction of **30 BTC** he has previously received
- Bob can insert into the output of the new transaction both Alice's address, to which he sends 12 BTC, and Charlie's, to which he sends 18 BTC
- In the **output of a transaction**, the sender can insert a **list of addresses** of the payment recipients

Multiple output



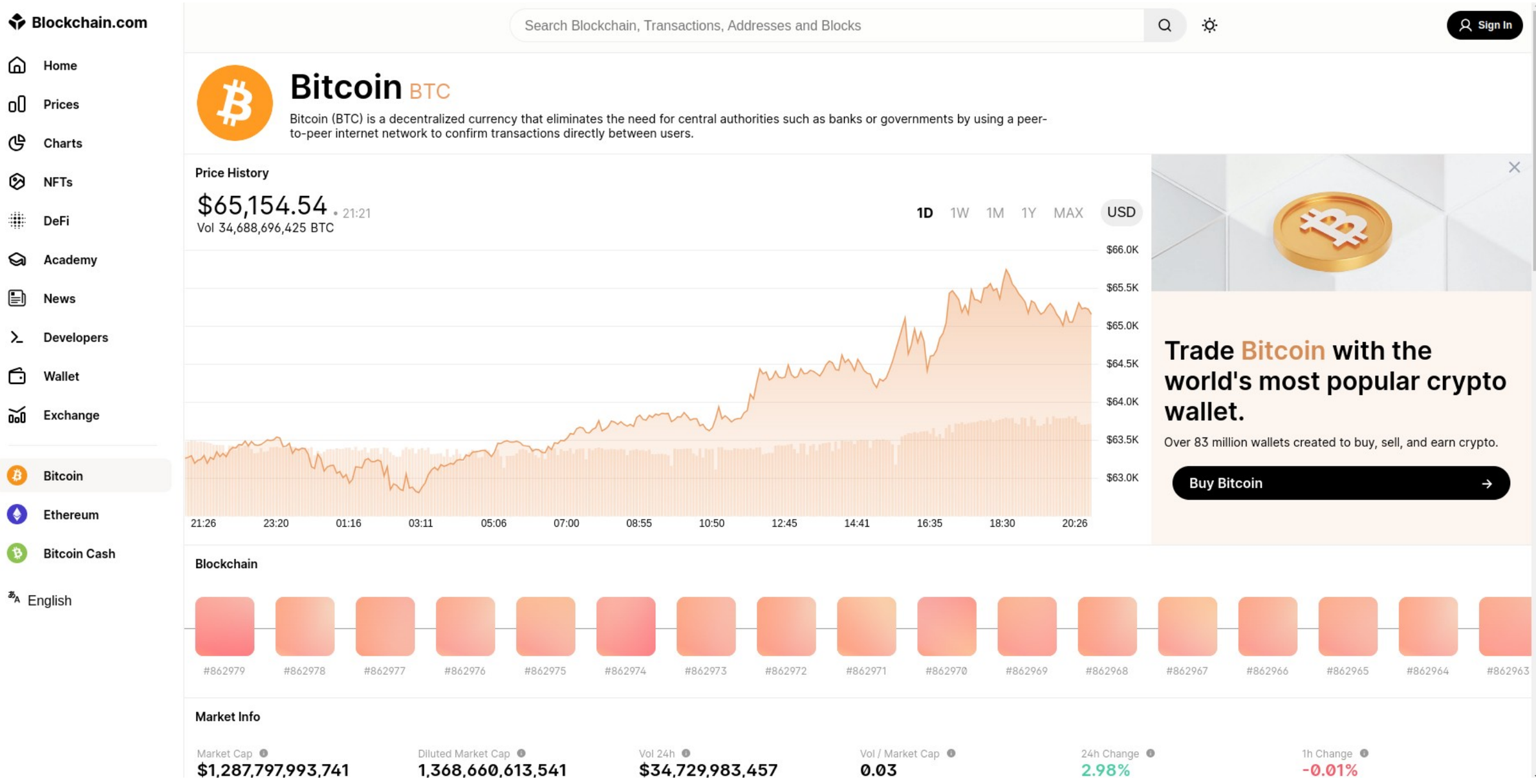
Change

- **Charlie can be Bob** himself and in this case we have a **change** of 18 BTC going back to Bob
- Bob can use the same address or a different one (as often suggested)
- Each user can have **multiple Bitcoin addresses**

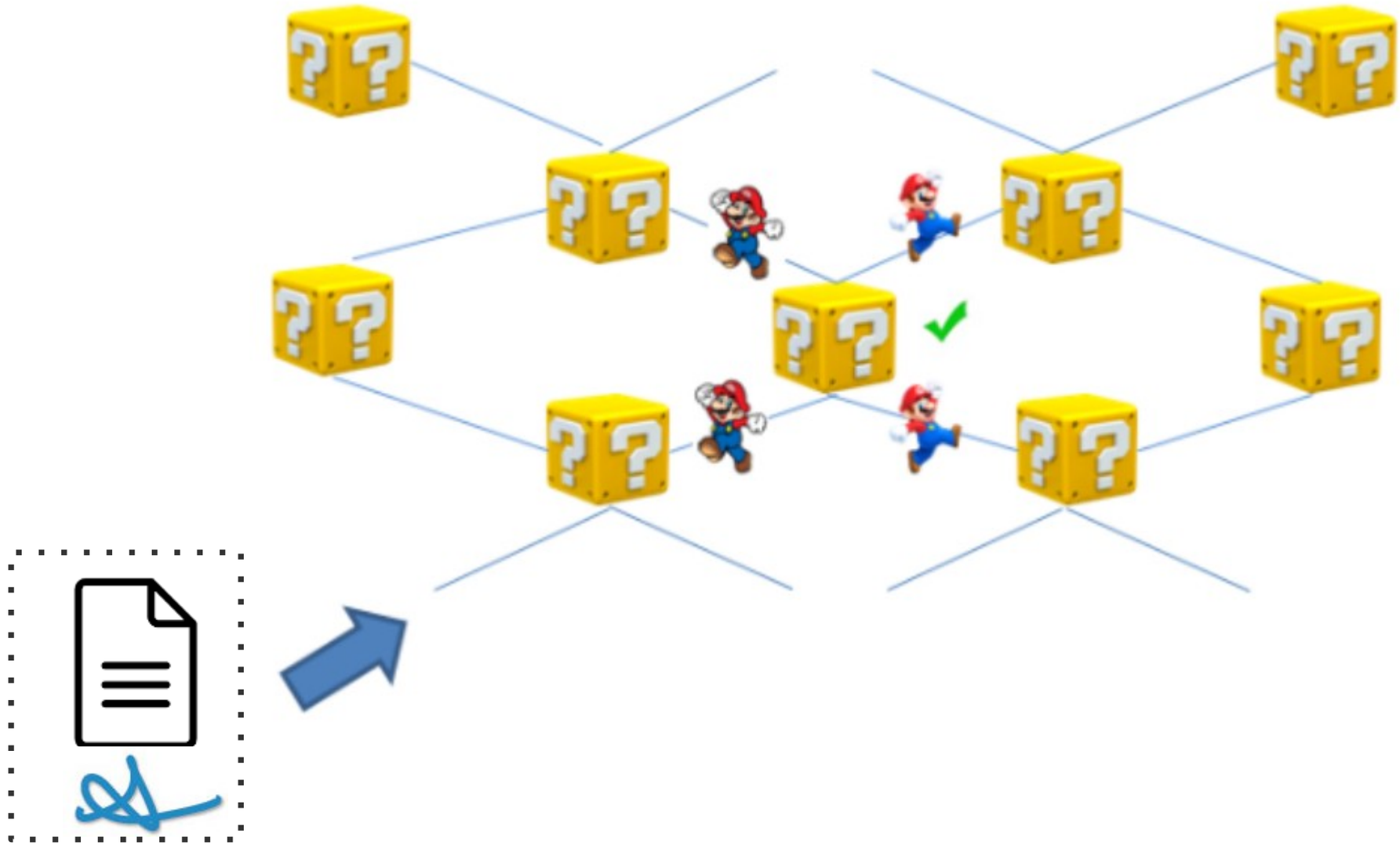
Transaction fee

- The **transaction fee** does not depend on the amount spent but on the **weight in terms of KB**
- The **number of UTXOs** used to compose a transaction is relevant to determine the fee of a transaction
- Therefore, to pay a lower fee it is better to **use a UTXO close to the amount to spend** rather than using several

Live explorer



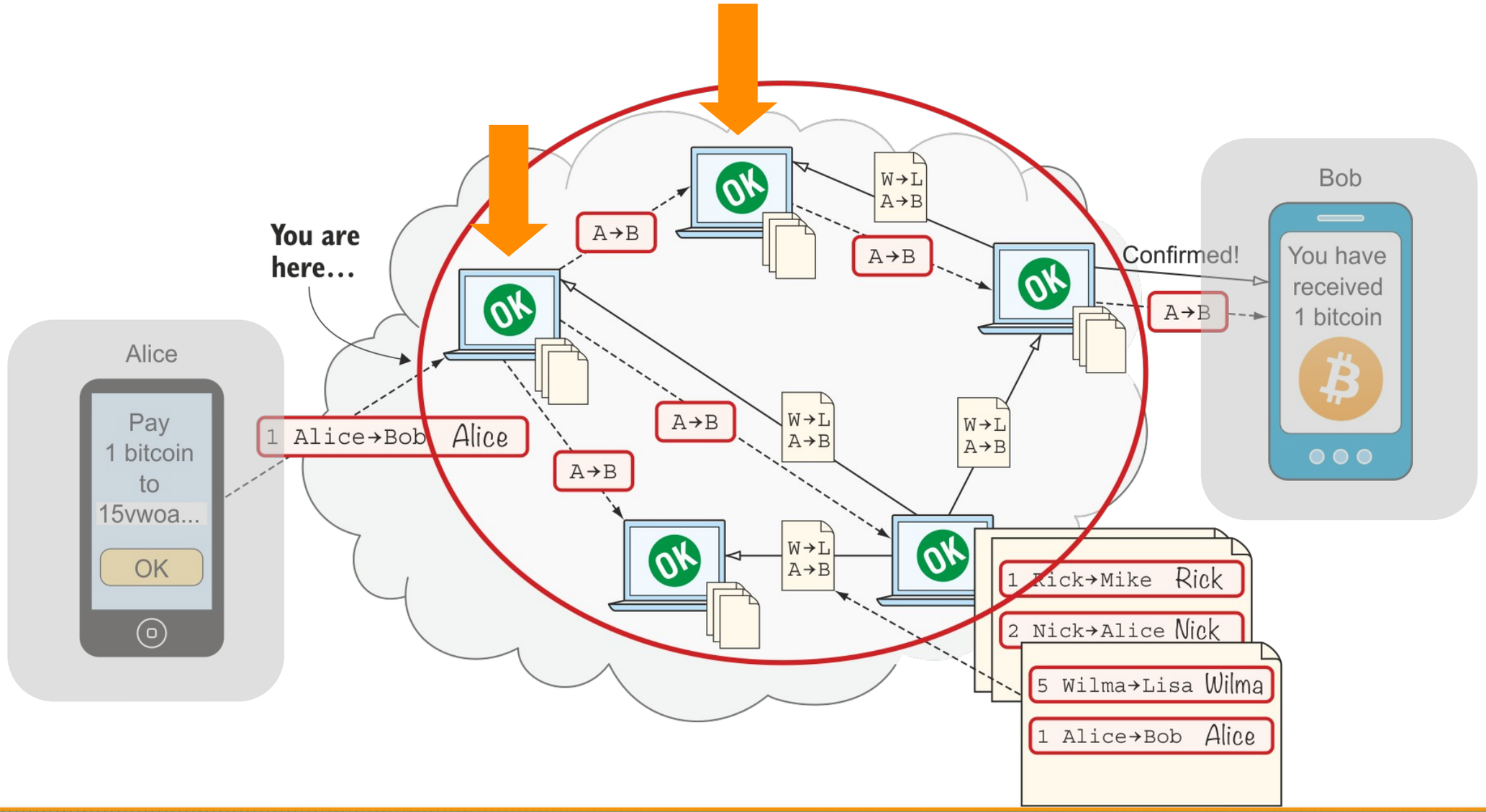
Transaction transmission



Transaction transmission

- All **transactions** broadcast in the Bitcoin network **wait to be confirmed** in the **mempool**
- A high mempool size indicates more network traffic which will result in longer average confirmation time and higher priority fees

Storing transactions



Mining

- Mining is the process of **adding new blocks** of valid transactions to the blockchain
- Miners are also responsible for the **creation of all new Bitcoins** and are a fundamental part of the Bitcoin ecosystem
- Most Bitcoin users do not mine

Nice explanation here: https://learnmeabitcoin.com/beginners/mining

Solo miner

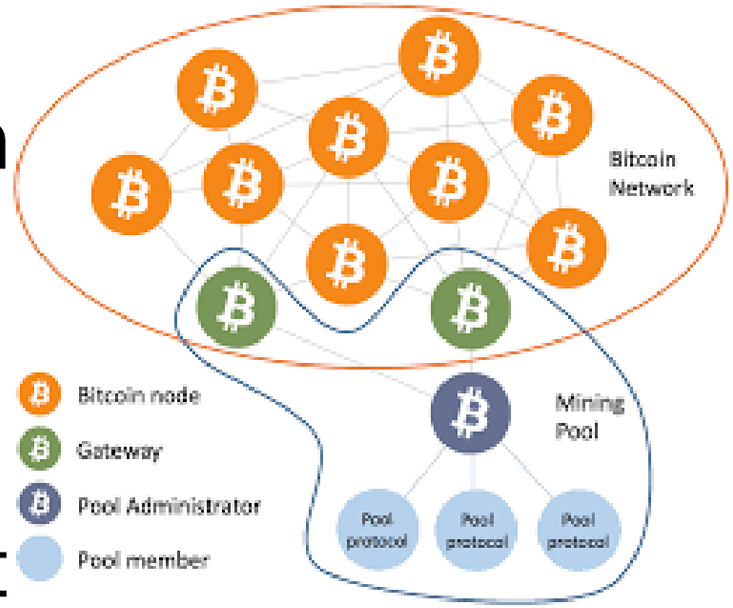
- Past: individual CPU, GPU, home-made hardware configurations
- They had a full copy of the blockchain to validate transactions
- They also needed a wallet to manage their mining reward

Solo miner

- Today: mining is split in two tasks
  - Block creation is done by peers that store the full blockchain
  - Hashing is done with dedicated hardware (ASIC), to speed up the computation

Mining pools

- Today: mining is not individual anymore and we have **mining pools**
- Organizations which compute hash to get a **reward**
  (see https://btc.com/stats/pool)



- Few mining pools concentrate a lot **power**
  → still real decentralization?

Six steps to be a miner

- **Listen for transactions** and **check them** by
  - checking that Bitcoins have not been spent before (no double-spending)
  - checking the signatures
- Maintain the blockchain and **listen for new blocks**
- **Prepare** a new block
  - group transactions into a new block (e.g., a file) that extends the latest known block

Six steps to be a miner

- **Reach the mining target** (see Proof-of-Work)
- **Hope** your block is accepted
  - e.g., other miners start mining on top of it
- If yes, **profit!**

Prof-of-Work

- Mining requires a task that is very **difficult to perform**, but **easy to verify**
- Satoshi proposed a sort of lottery/race/brute-force activity called **Proof-of-Work**
- Uses cryptography, with a **hash function** called **double SHA-256**

Prof-of-Work

- The mathematical challenge faced by miners consists in finding the **NONCE**
- SHA256(SHA256(**NONCE** + **hash previous block** + **transactions**)) <= **0000000000007C96A72BBC4445A5634BD**
- Trial and error procedure
- Try a new NONCE until the TARGET is reached

Prof-of-Work

- The mathematical challenge faced by miners consists in finding the **NONCE**
- SHA256(SHA256(**NONCE** + **hash previous block** + **transactions**)) <= **0000000000007C96A72BBC4445A5634BD**
- Trial and error procedure
- Try a new NONCE until the TARGET is reached

Prof-of-Work

- The average number of “trial and errors” steps needed to meet the target is the **Difficulty**
- **Every two weeks it is adjusted** as follows:
  - **1 block** on average every **10 minutes**
  - **20160 minutes** in **2 weeks** (around 2016 blocks)
  - **T = time required to mine the last 2016 blocks**
  - NewDifficulty = Difficulty * (**20160/T**)
  - If T < 20160, the NewDifficulty increases (and the target decreases)

Prof-of-Work

- The Difficulty is established by the Bitcoin network to moderate the mining speed: **every 10 minutes** a new block should be added to the blockchain

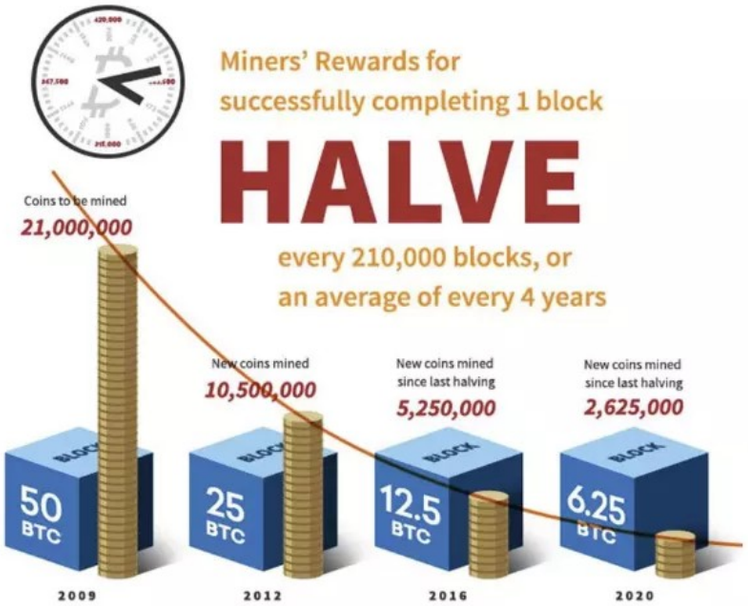
Why 10 minutes?

Reward for miners

- To encourage their work, miners receive incentives (rewards)
  - **fees** of the transactions contained in the block
    - priority to the transactions with highest fees
  - **newly created Bitcoins**
    - Coinbase transaction

Reward for miners

- The Coinbase transaction
  - is the **only way to create new Bitcoins**
  - reward per block was 50 BTC in 2008
  - this value is halved every 4 years
- **April 25, 2024**
  Last Bitcoin halving: the reward dropped to 3.125 Bitcoin per block
  See bock 840000:
  https://www.blockchain.com/explorer/blocks/btc/840000



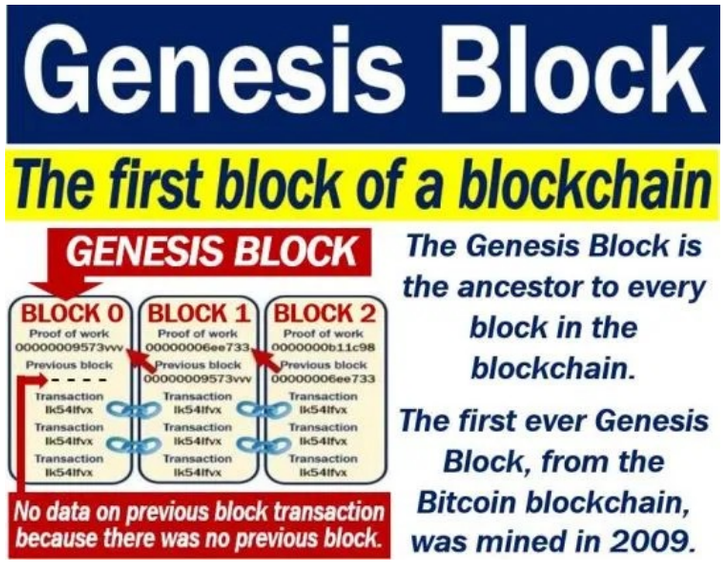
Issuing new Bitcoins

- Bitcoins are finite!
  - the maximum and total amount of Bitcoins that can ever exist is **21 million**
  - there are **less than ???? million** Bitcoins left to be mined
  - Satoshi estimated that the last Bitcoin should be mined sometime in the year 2140

https://www.buybitcoinworldwide.com/how-many-bitcoins-are-there/

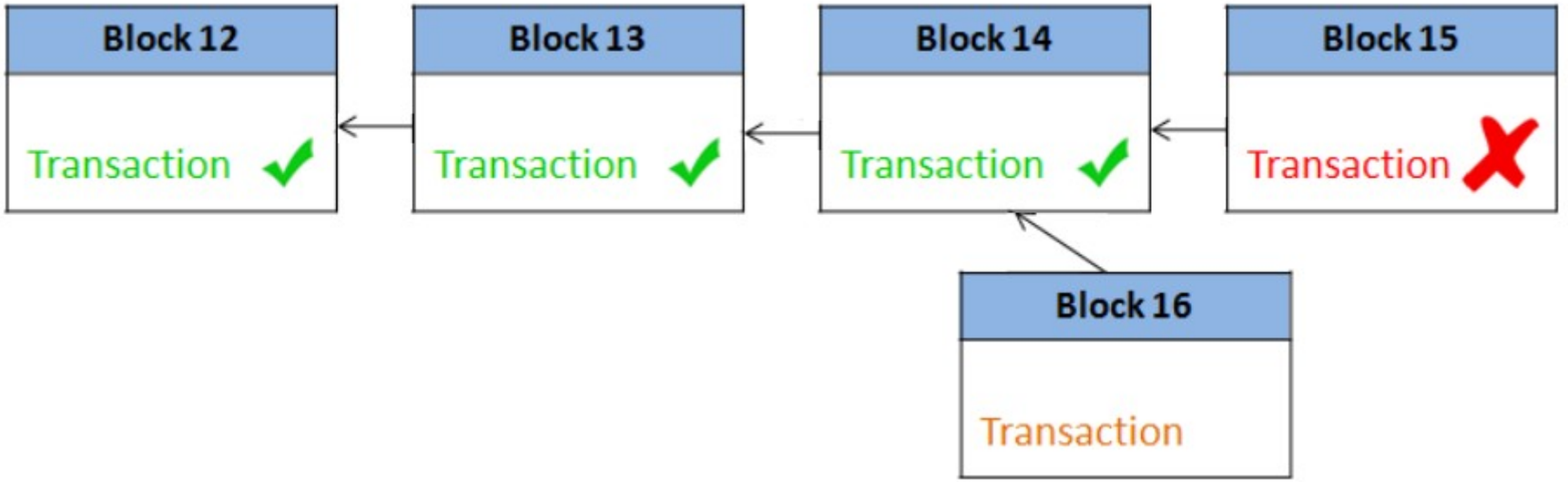
Genesis block

- It was mined over the course of six days by Nakamoto to start the blockchain
- https://www.blockchain.com/btc/block/000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f
- https://ma.ttias.be/retrieving-the-genesis-block-in-bitcoin-with-bitcoin-cli/

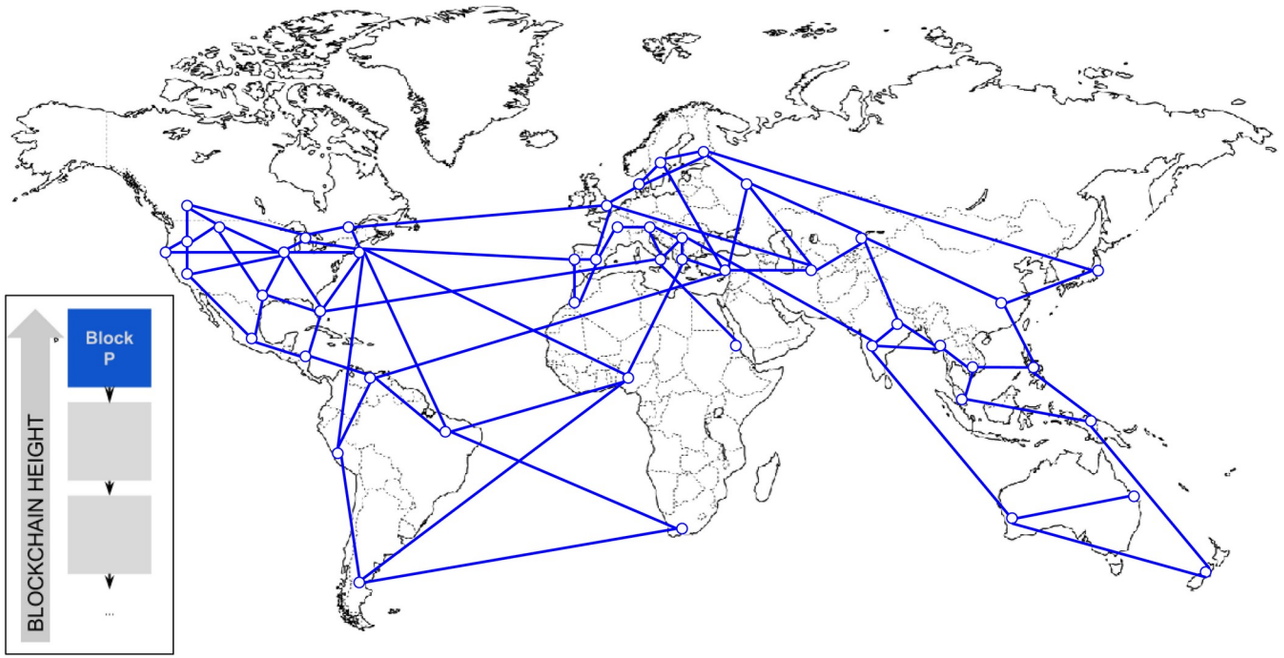


Forks

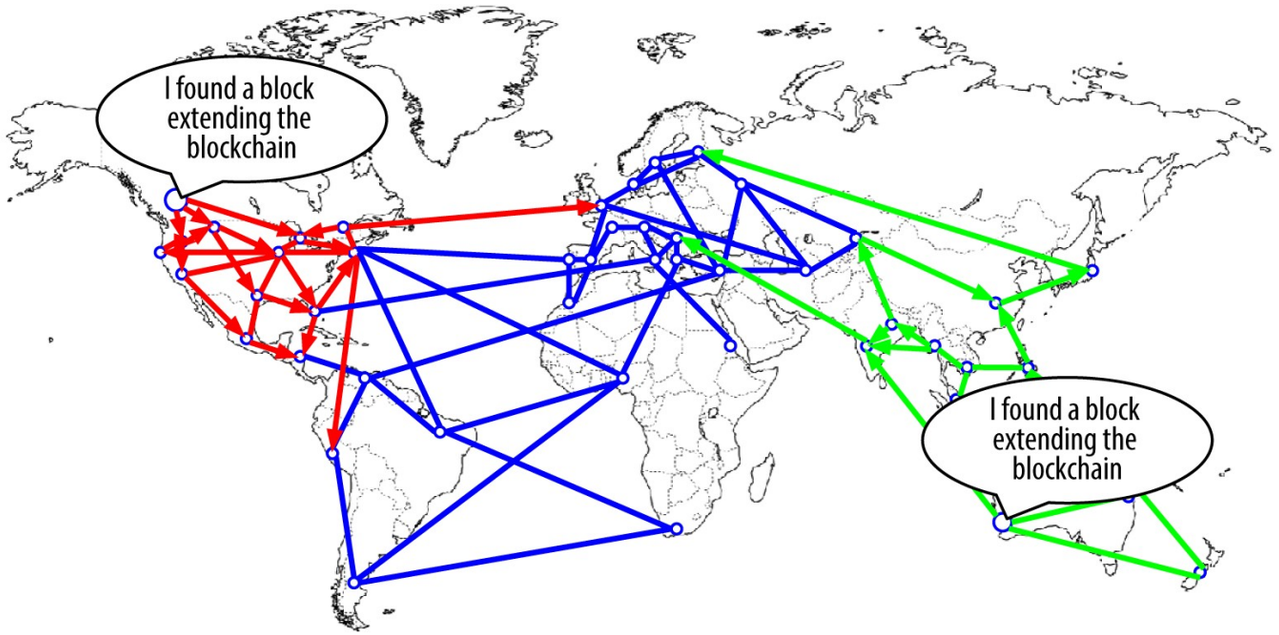
- The miner that has the right to insert a new block into the blockchain, adds it after the last known block
- But we are in a **decentralized** network and...



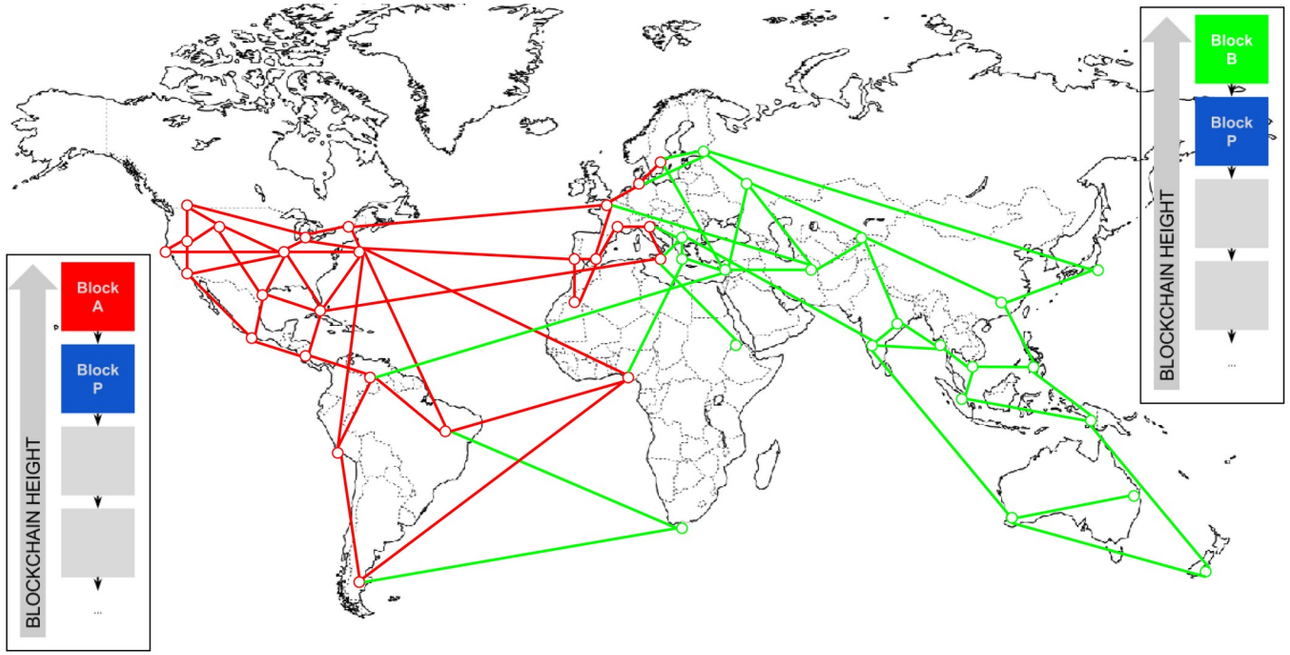
Forks



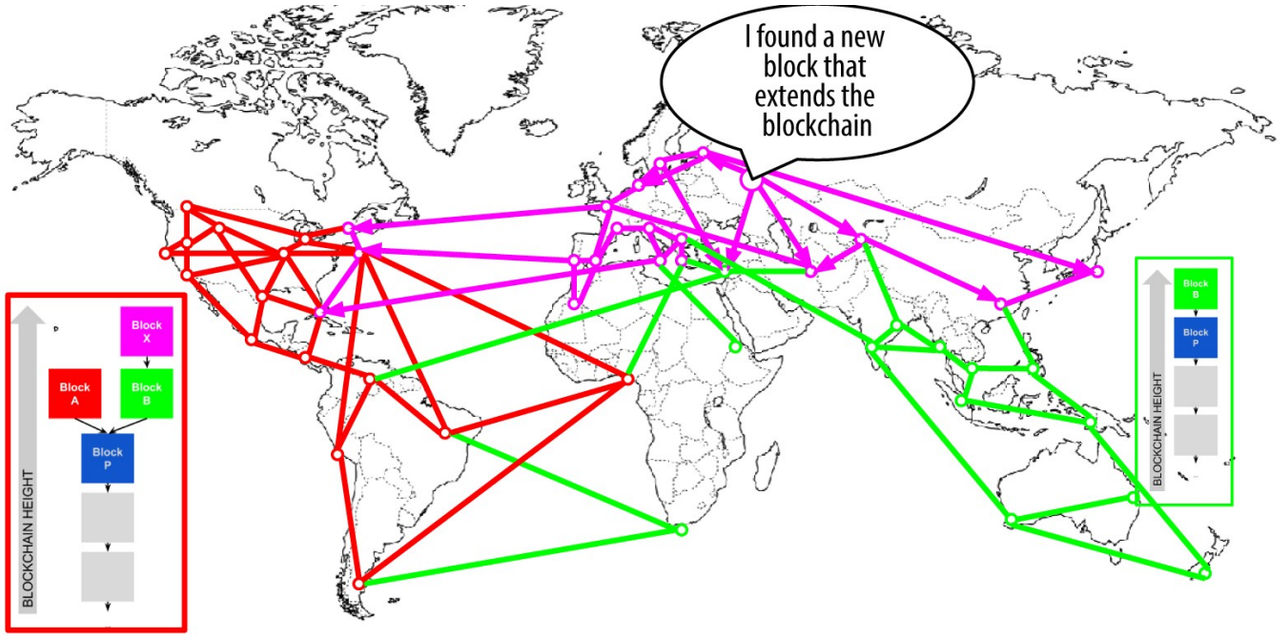
Forks


I found a block extending the blockchain
I found a block extending the blockchain

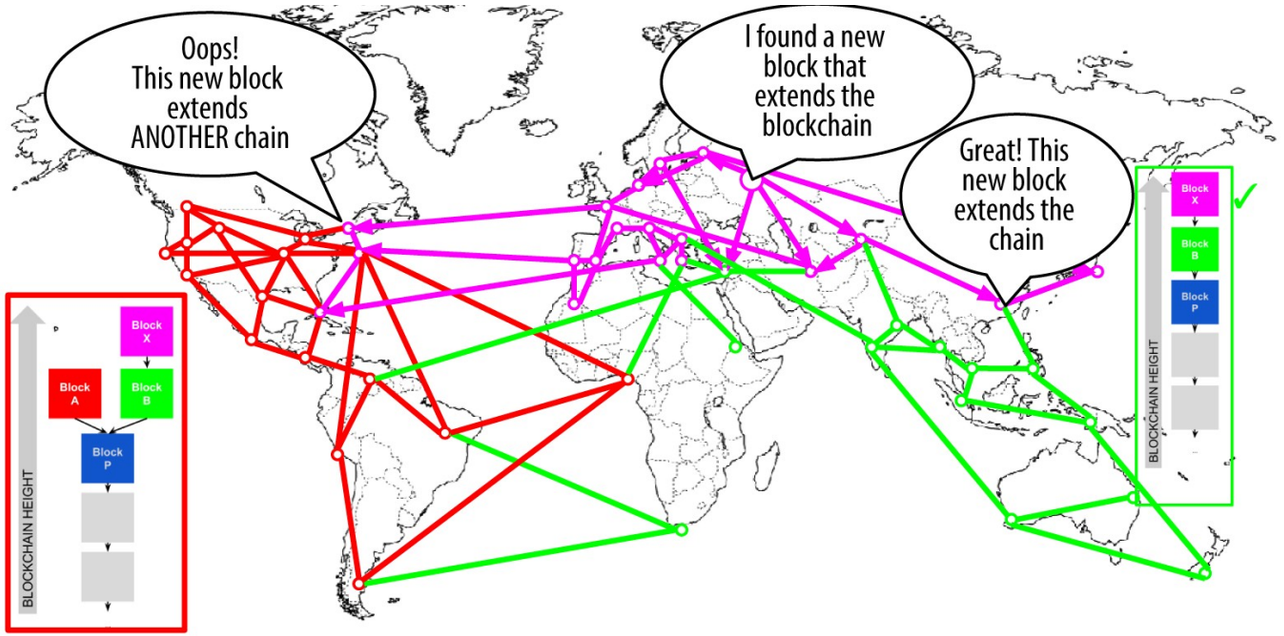
Forks


Block A
Block P
...
BLOCKCHAIN HEIGHT
Block B
Block P
...
BLOCKCHAIN HEIGHT

Forks


I found a new block that extends the blockchain
Block X
Block A
Block B
Block P
BLOCKCHAIN HEIGHT
Block B
Block P
BLOCKCHAIN HEIGHT

Forks


Oops!
This new block
extends
ANOTHER chain
I found a new
block that
extends the
blockchain
Great! This
new block
extends the
chain
Block
X
Block
A
Block
B
Block
P
BLOCKCHAIN HEIGHT
Block
X
Block
B
Block
P
BLOCKCHAIN HEIGHT

The longest chain wins!

- A block is valid only after other blocks are added on top of it
  - in the Bitcoin network usually wait for 6 blocks (around 1 hour) to consider a block really valid
- Blocks in the shorter chain are “canceled” and the unspent transactions return into the mempool

How many confirmations?

- **0** - can still be reversed! Wait for at least one
- 1 - enough for small payments, less than $1,000
- 3 - enough for payments $1,000 - $10,000. Most exchanges require 3 confirmations for deposits
- **6** - for large payments between $10,000 - $1,000,000 Standard for most transactions to be considered secure
- 60 - suggested for large payments greater than $1,000,000. Less is likely fine, but this is to be safe!