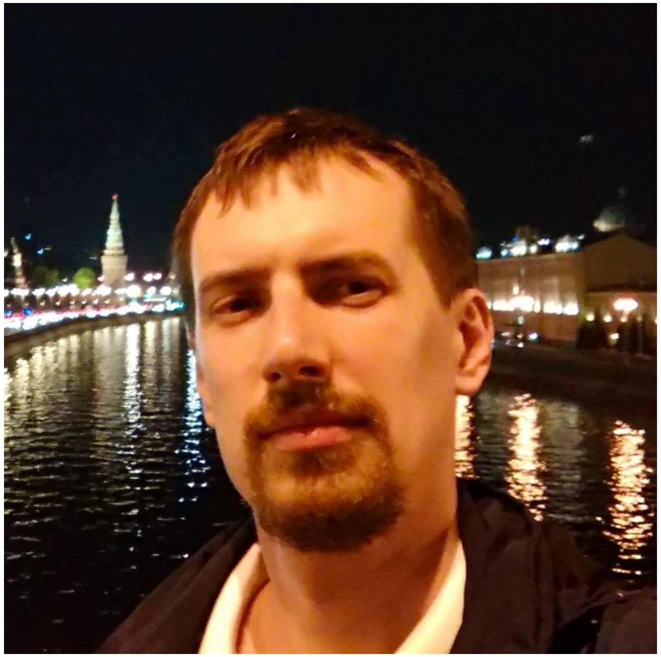


Sergey Gennadievich Sinitsa

Professor, Vice Dean for Science,
Faculty of Computer Technology and Applied Math
Kuban State University, Russia.

Initlab CTO, 15+ years web technology experience.
https://initlab.de/

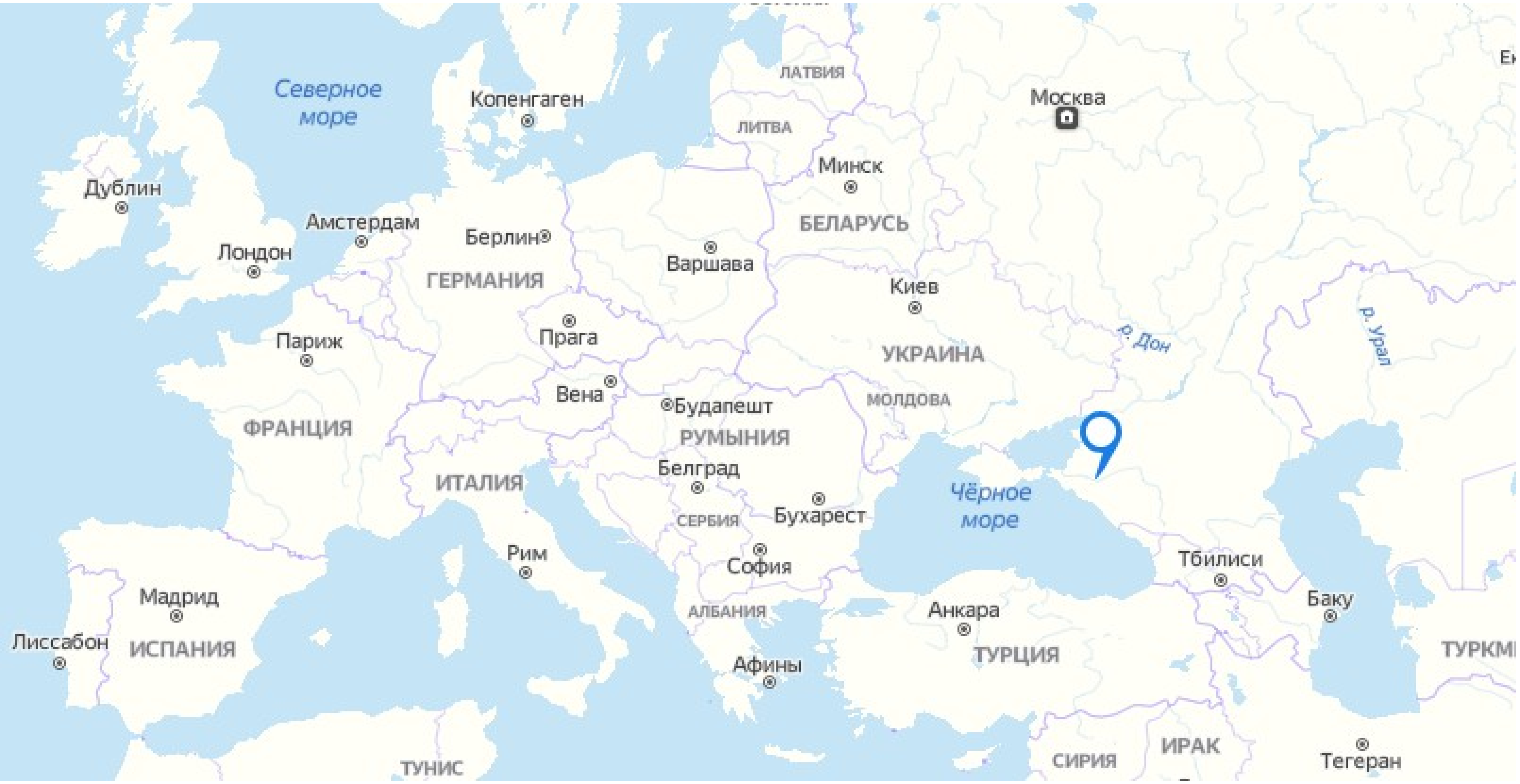
sin@kubsu.ru,

In collaboration with bitzlato.com blockchain experts.

Kuban State University

- 20k students / year
- 16 faculties, 100 years anniversary
- 2000+ staff
- https://kubsu.ru/en/

Krasnodar


ЛАТВИЯ
Северное море
Копенгаген
Москва
ЛИТВА
Минск
Дублин
Амстердам
Берлин
БЕЛАРУСЬ
Лондон
Варшава
ГЕРМАНИЯ
Киев
р. Дон
р. Урал
Париж
Прага
УКРАИНА
Вена
Будапешт
МОЛДОВА
ФРАНЦИЯ
РУМЫНИЯ
ИТАЛИЯ
Белград
Чёрное море
Бухарест
СЕРБИЯ
Рим
София
Тбилиси
Мадрид
АЛБАНИЯ
Анкара
Баку
Лиссабон
ИСПАНИЯ
ТУРЦИЯ
ТУРКМ
Афины
ИРАК
СИРИЯ
Тегеран
ТУНИС

Blockchain tools and technologies

My class gives theoretical base and practical skills to architect solutions and develop ecommerce web applications interacting with blockchains. You will learn how to apply web and blockchain technologies, tools, services and libraries to process cryptocurrency transactions programmatically in a secure way and create fully autonomous P2P Ecommerce solutions, Financial and Cryptocurrency Exchange Services etc.

Learning outcomes:

- Understand how blockchain technology works and how to apply it for P2P Ecommerce.
- Understand key features of three blockchain generations: Bitcoin, Ethereum and Telegram Open Network (TON).
- Practice with official Bitcoin app, Copay Wallet and Blockchain explorer to send and receive coins.
- Learn how to setup Bitcoin node and work with it via command line and web-service to accept Bitcoin P2P payments, perform transactions and work with multisignature wallets programmatically in your NodeJS web application.
- Learn how to setup Ethereum node, develop and deploy Ethereum smart contracts in Solidity language and work with Ethereum blockchain programmatically using Web3 API from your NodeJS application.
- Work with P2P cryptocurrency exchange services to get coin conversion rates programmatically.

Class programm

Prerequesties: JavaScript, web services and Linux CLI experience or ability to learn it quickly.

Laptop or PC with Ubuntu or Debian Linux and fast Internet connection is highly recommended. Webcam and Mic!

Lectures:

1. Introduction, Bitcoin architecture
2. Addresses, transactions, tools and API
3. Bitcoin node, networking
4. Ethereum architecture, smart contracts
5. ERC20, ERC721, Ethereum tools and API
6. Services and Exchange, DAO, SSI, European and other national initiatives
7. TON Architecture, Blockchain Future

Labs:

1. Copay, Faucet, Transactions, Blockchain explorer, Multisig Wallet
2. Bitcoin JS + External services for posting transactions. Multisignature Transaction (team work)
3. Bitcoin official wallet, Bitcoin CLI, Bitcoin Node + JSON RPC + Node JS
4. Remix IDE, Metamask Smart Contract deployment (team work)
5. Ethereum Node, Geth dev CLI, programmatic Smart Contract deployment, Web 3 API + Node JS
6. Ethereum Smart Contracts, ERC20 Tokens, Wallet
7. TON FunC smart contracts

Project (team work)

Homework and team work:
80 hours for labs and project!

Blockchain project

1. Work in teams 2-4, students per team.

2. Propose an Idea, discuss with me. Your project should apply blockchain technology to solve real world problems (may be not ecommerce or cryptocurrency related).

3. Choose your own stack of technologies. Describe the purpose and architecture. Design an API, data structures, write smart contract code and code examples to communicate with your smart contract.

4. Use GitLab for collaboration on code and documentation.

5. Write a paper in a team describing your results.

6. Give an expert talk till the exam date. Split your presentation in a team to describe individual contributions.

How to pass a course successfully

1) Finish the project and give an expert talk.

2) Finish half of labs tasks or more.

3) Visit half of lectures and labs or more.

Grading:

7 labs for 10 points each + 30 points for project = 100 points.

60 is a planned minimum to pass the course.

The grading from 60 to 100 is done proportionally.

Books

Mastering Bitcoin (Second Edition, Second Print): Programming the Open Blockchain:
https://github.com/bitcoinbook/bitcoinbook

Joseph Holbrook. Architecting Enterprise Blockchain Solutions

Chris Dannen. Introducing Ethereum and Solidity: Foundations of Cryptocurrency…

Official documentation of Bitcoin, Etherium etc...

Bitcoin

Satoshi Nakamoto original paper, 2008 https://bitcoin.org/bitcoin.pdf

C++ official client was developed in one year, 3 january 2009 first block mined.

It took 40 years to develop necessary cryptography and e-money theory. First working solution to decentralized double spend problem.

Remarkable CS achievement.

Jinnie from a Bottle.

Bitcoin Cash fork in 2017.

Blockchain

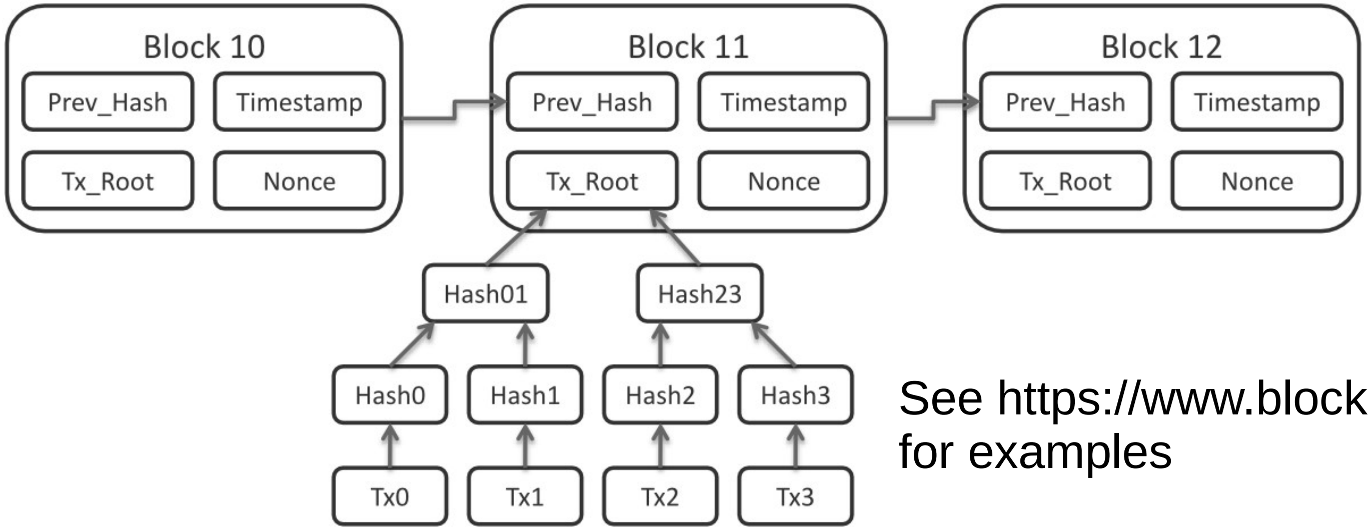
The idea of blocks chain.

Each block is a set of transactions.

Miners do “work” calculating SHA-256 hashes to accept blocks.

Hash should be less then target difficulty. Target difficulty adapts to mine ~6 blocks per hour for bitcoin

Miners exchange block information.



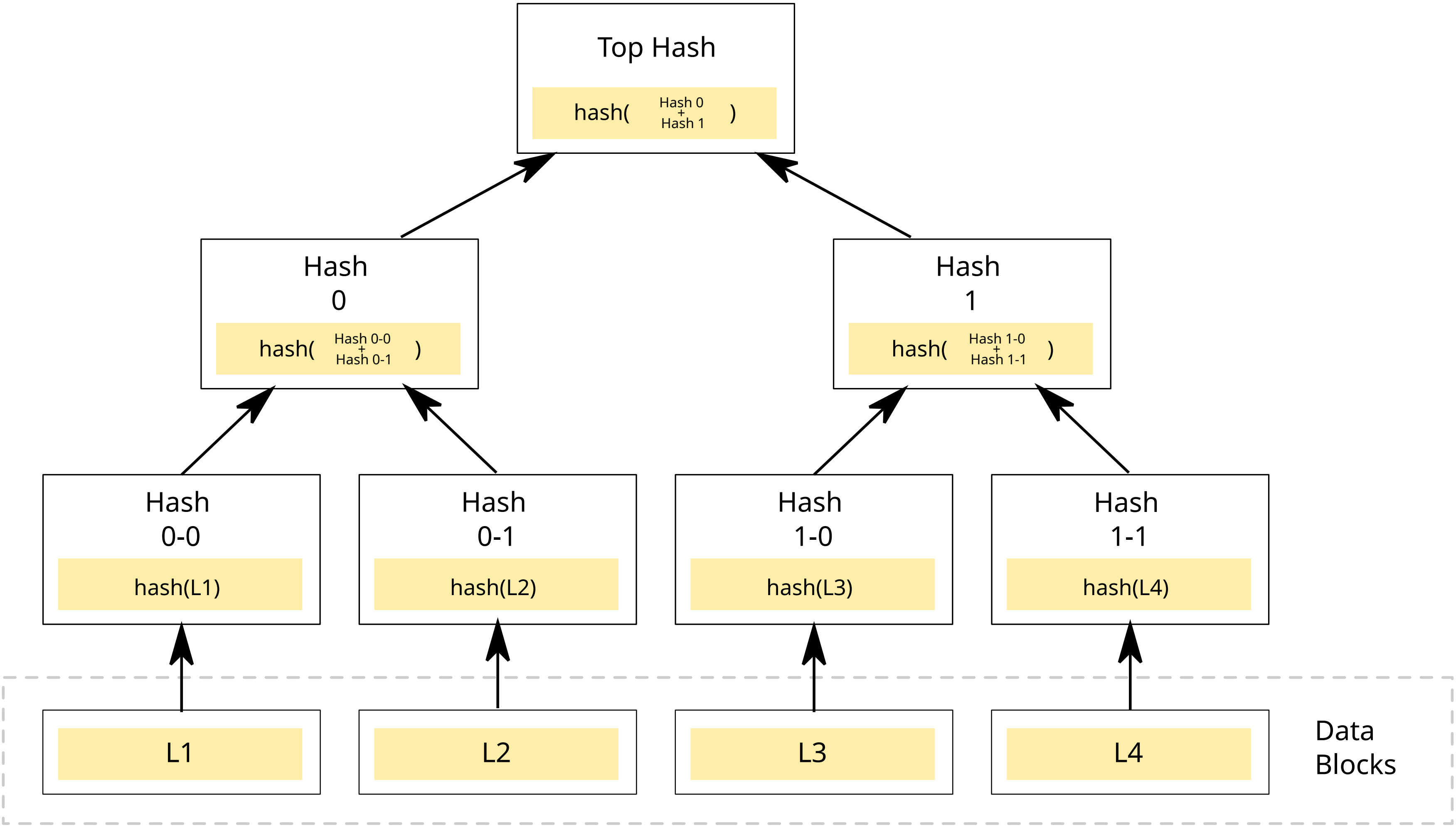
See https://www.blockchain.com/en/explorer for examples

Read https://en.bitcoin.it/wiki/Protocol_rules and https://en.bitcoin.it/wiki/Protocol_documentation for data structures and protocol details.

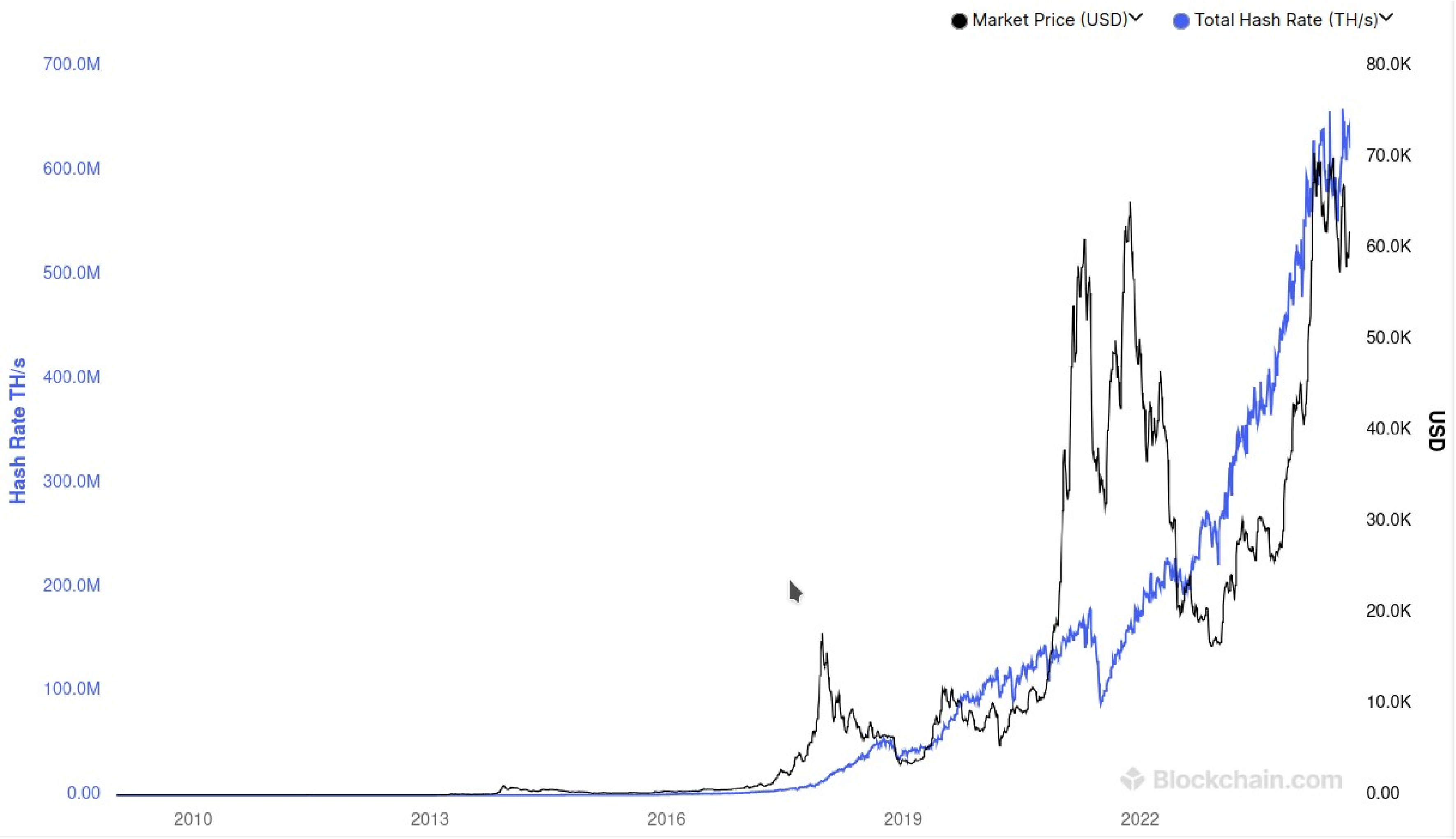
Blockchain Merkle Hash tree

Validation nodes do not store all transactions but only block headers
with Top Merkle Hash of all block transactions.

These nodes ask Merkle Path hashes from Full Nodes to validate transactions.



Bitcoin hash rate

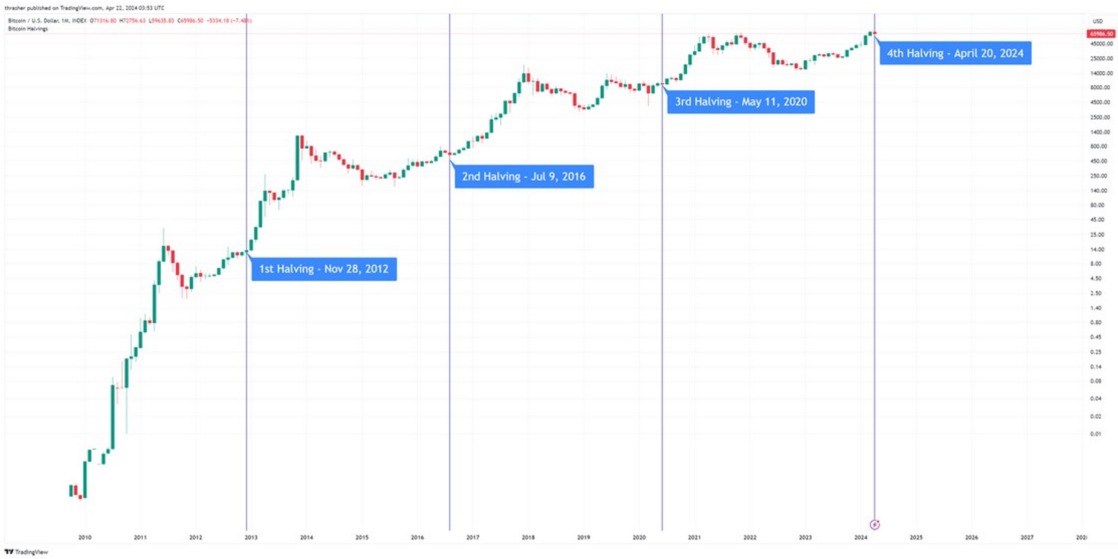


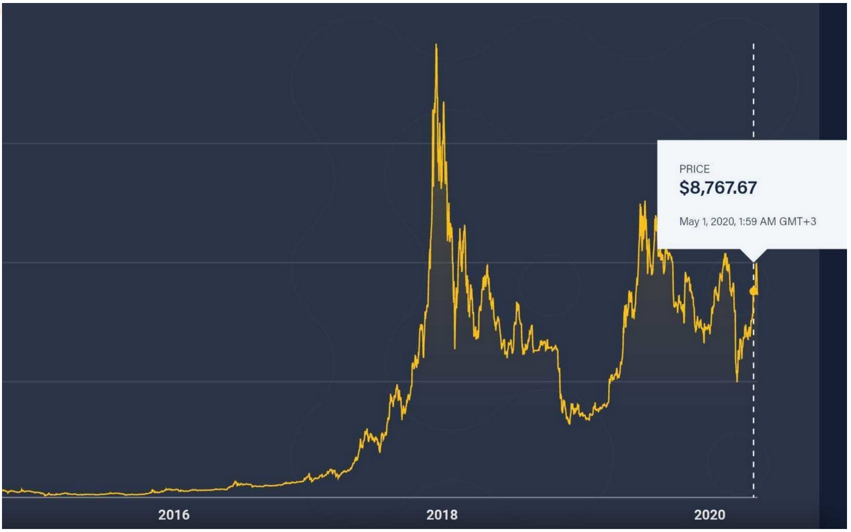
Target difficulty is so high that miners use GPUs, ASICs and organize mining pools. How pools work?

See https://www.blockchain.com/charts/hash-rate

Bitcoin Halving

- 21,000,000 Bitcoins to ever be produced (19kk+ mined so far, 1,250,950 Bitcoins left to mine)
- Target of 10-minute block intervals
- Halving event occurring every 210,000 blocks (approximately every 4 years)
- Block reward which starts at 50 and halves continually every halving event until it reaches 0 (approximately by year 2140)
- The first halving event occurred on the 28th of November, 2012 (UTC) at block height 210,000 (25 BTC per block)
- The second halving event occurred on the 9th of July, 2016 (UTC) at block height 420,000 (12.5 BTC per block)
- The third halving event occurred on the 11th of May, 2020 (UTC) at block height 630,000 (6.5 BTC per block)
- The fourth halving event occurred on the 20th of April, 2024 (UTC) at block height 840,000 (3.25 BTC per block)





Source https://www.bitcoinblockhalf.com/

Blockchain ISO defines basic terms

June 2020:
ISO 22739:2020
Blockchain and distributed ledger technologies
https://www.iso.org/obp/ui/ru/#iso:std:iso:22739:ed-1:v1:en:e

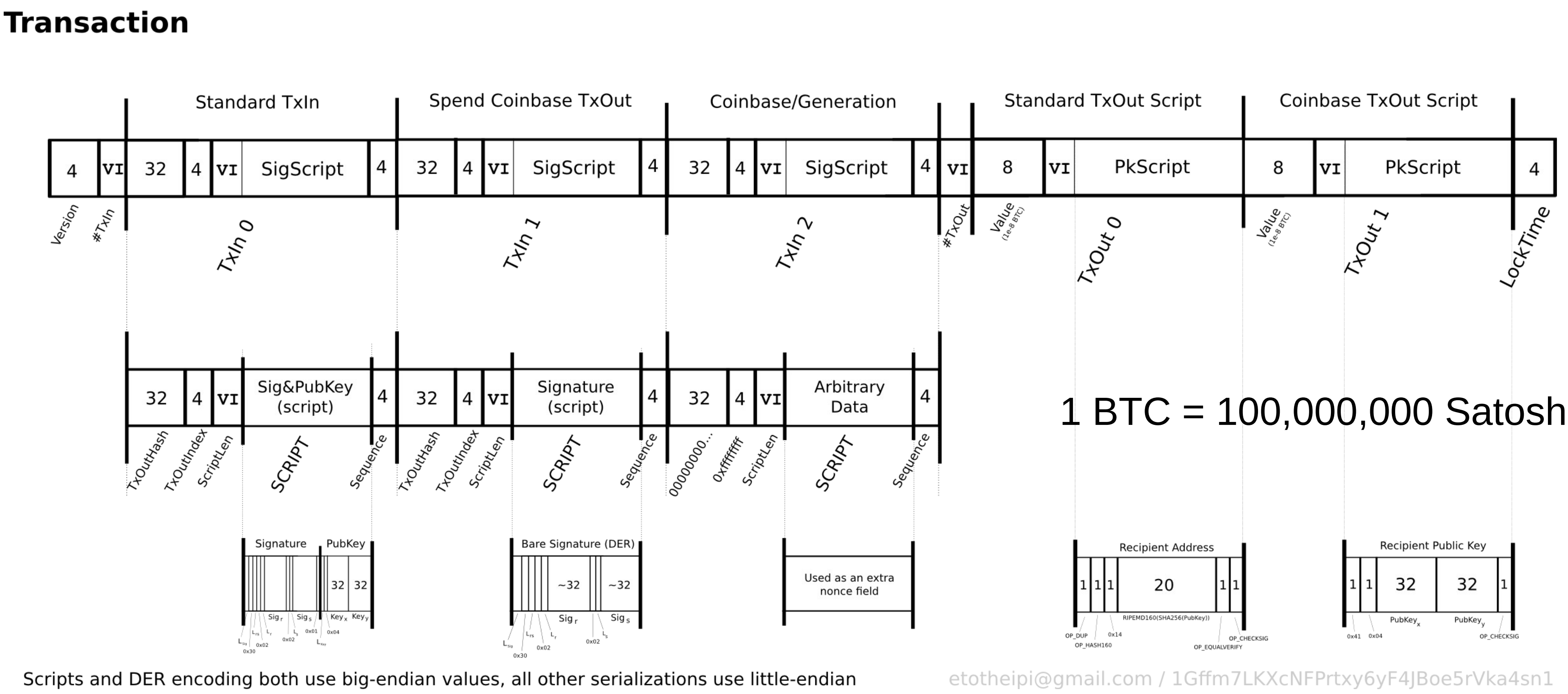
blockchain
distributed ledger (3.22) with confirmed blocks (3.9) organized in an append-only, sequential chain using cryptographic links (3.16)

distributed ledger
ledger (3.43) that is shared across a set of DLT nodes (3.27) and synchronized between the DLT nodes using a consensus mechanism (3.12)

ledger
information store that keeps records (3.67) of transactions (3.77) that are intended to be final, definitive and immutable (3.40)

Blockchain Transactions

A transaction is a transfer of Bitcoin value that is broadcast to the network and collected into blocks. A transaction typically references previous transaction outputs as new transaction inputs and dedicates all input Bitcoin values to new outputs. Transactions are not encrypted, so it is possible to browse and view every transaction ever collected into a block. Once transactions are buried under enough confirmations they can be considered irreversible.



1 BTC = 100,000,000 Satoshi

See https://en.bitcoin.it/Transaction for full details

Blockchain Scripts

See https://en.bitcoin.it/wiki/Script for full details

A script is essentially a list of instructions recorded with each transaction that describe how the next person wanting to spend the Bitcoins being transferred can gain access to them. The script for a typical Bitcoin transfer to destination Bitcoin address D simply encumbers future spending of the bitcoins with two things: the spender must provide

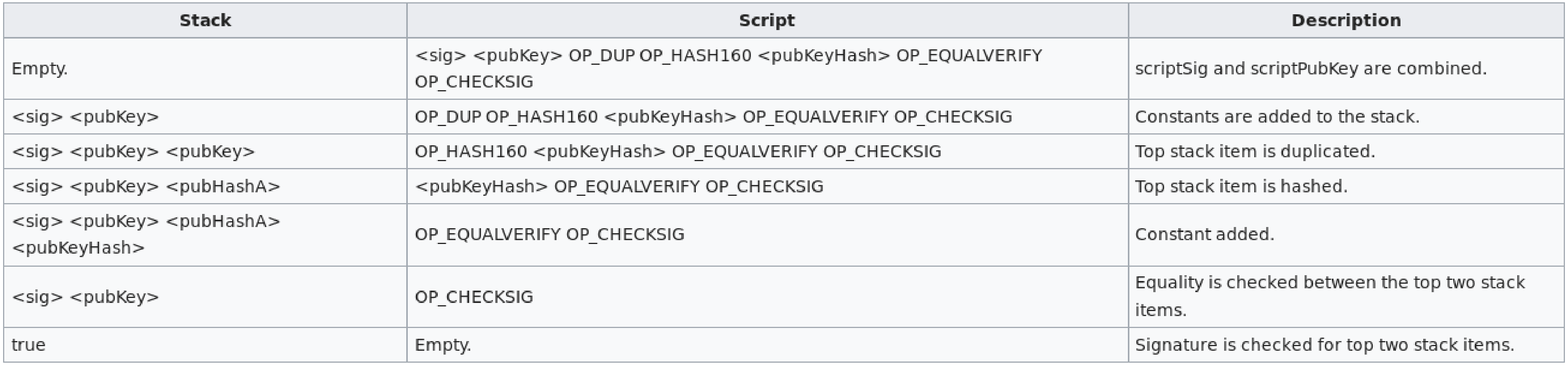
1) a public key that, when hashed, yields destination address D embedded in the script, and

2) a signature to prove ownership of the private key corresponding to the public key just provided.

A transaction is valid if nothing in the combined script triggers failure and the top stack item is True (non-zero) when the script exits.

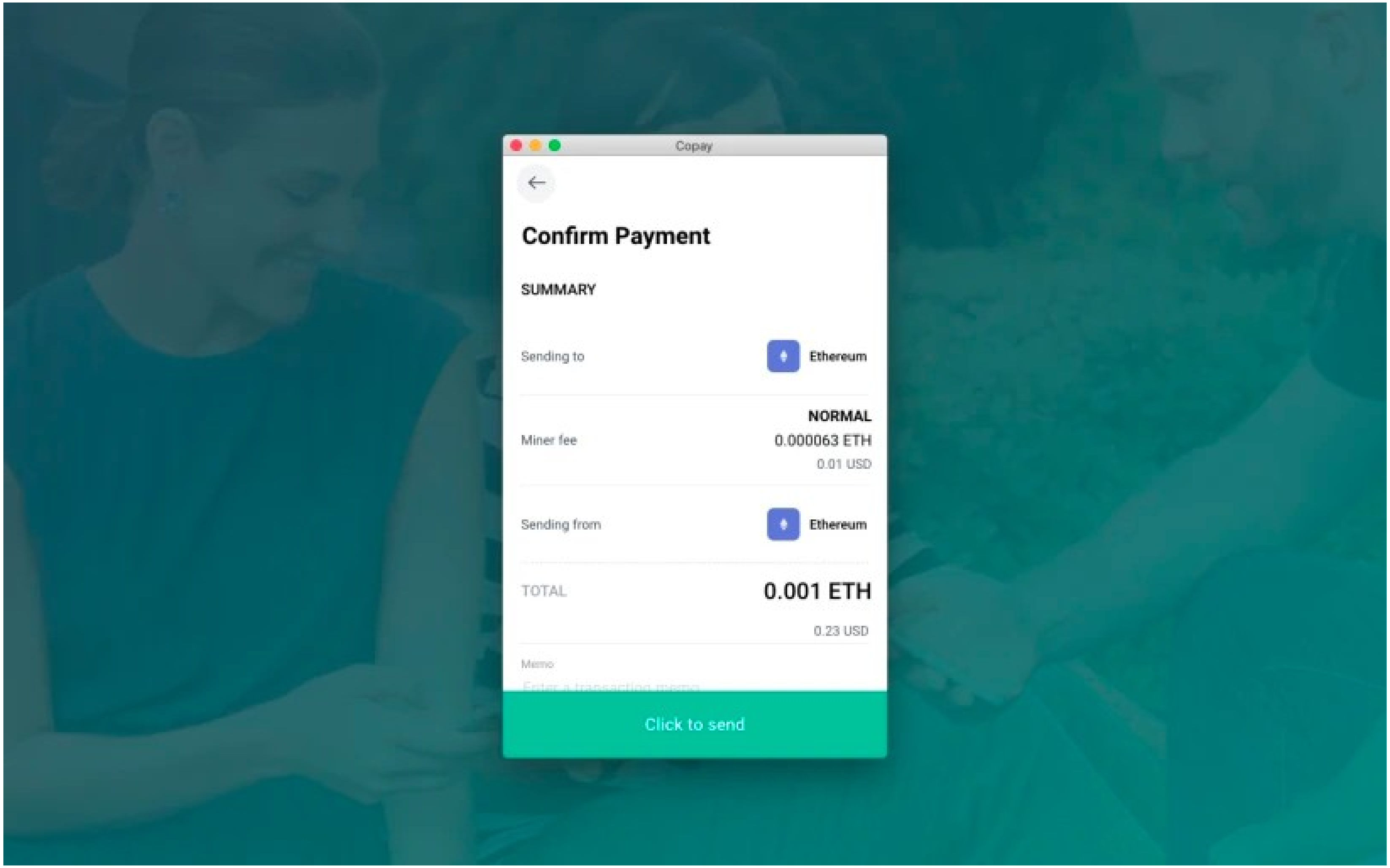
The party that originally sent the Bitcoins now being spent dictates the script operations that will occur last in order to release them for use in another transaction.

The party wanting to spend them must provide the input(s) to the previously recorded script that results in the combined script completing execution with a true value on the top of the stack.



Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.
<sig> <pubKey>	OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Constants are added to the stack.
<sig> <pubKey> <pubKey>	OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is duplicated.
<sig> <pubKey> <pubHashA>	<pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is hashed.
<sig> <pubKey> <pubHashA> <pubKeyHash>	OP_EQUALVERIFY OP_CHECKSIG	Constant added.
<sig> <pubKey>	OP_CHECKSIG	Equality is checked between the top two stack items.
true	Empty.	Signature is checked for top two stack items.

Copay Wallet



Copay

Confirm Payment

SUMMARY

Sending to Ethereum

Miner fee NORMAL
0.000063 ETH
0.01 USD

Sending from Ethereum

TOTAL 0.001 ETH
0.23 USD

Memo

Enter a transaction memo

Click to send

Source: https://github.com/bitpay/copay

Binary: https://snapcraft.io/install/copay/debian

Copay Wallet

Secure bitcoin on your own terms with an open source, multisignature wallet from BitPay.

Copay users can hold funds individually or share finances securely with other users with multisignature wallets, which prevent unauthorized payments by requiring multiple approvals. Here are some ways Copay can be used with others:

To save for vacations or joint purchases with friends
To track family spending and allowances
To manage business, club, or organization funds and expenses

Custodial (third party manages your keys) VS Non-Custodial (you own keys)

Copay Features

Support Bitcoin and Bitcoin Cash

Multiple bitcoin wallet creation and management in-app

Integration for buying and selling bitcoin.

Integration for buying Amazon.com gift cards.

Intuitive multisignature security for personal or shared wallets

Device-based security: all private keys are stored locally, not in the cloud

Hierarchical deterministic (HD) address generation and wallet backups

Payment protocol (BIP70-BIP73) support: easily-identifiable payment requests and verifiably secure bitcoin payments

Support for 150+ currency pricing options and unit denomination in BTC or bits

Email and push notifications for payments and transfers

Easy spending proposal flow for shared wallets and group payments

Support for Bitcoin testnet wallets

Customizable wallet naming and background colors

Multiple supported languages, including French, German, Chinese (Simplified), and Spanish

Copay Wallet Debian 10 install

```
sudo apt update
sudo apt install snapd
snap install copay
```

Logout / relogin

```
copay
```

Bitcoin testnet

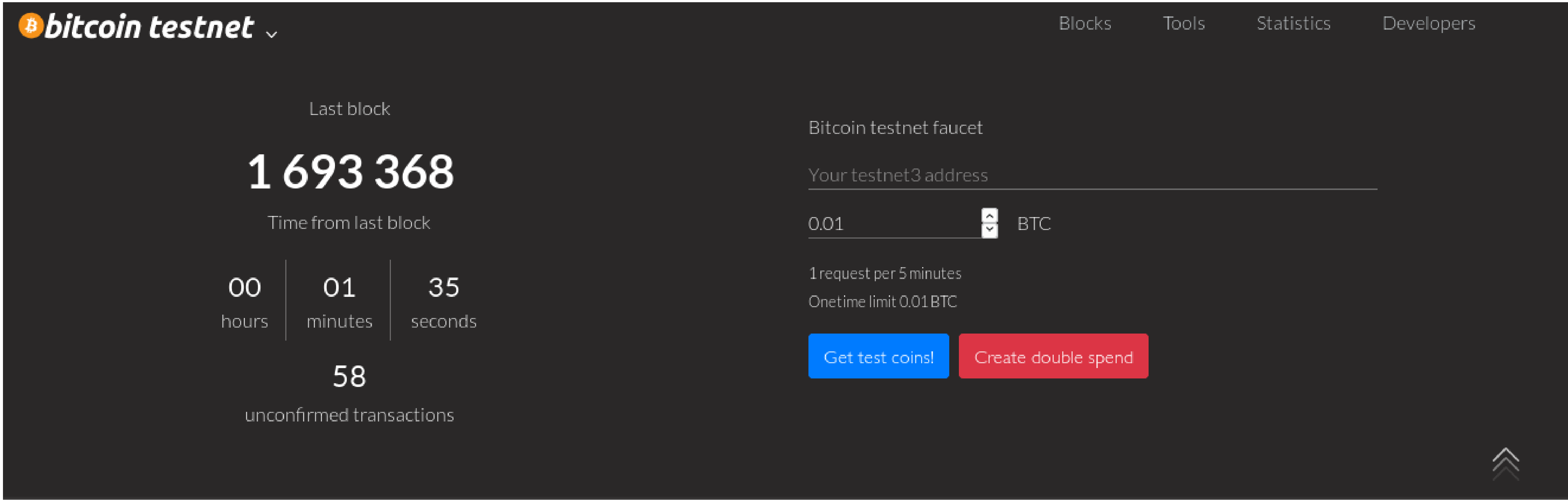
Get test coins:
https://coinfaucet.eu/en/btc-testnet/
https://bitcoinfaucet.uo1.net/
https://tbtc.bitaps.com/

Or just Google Bitcoin Faucet

Get address transaction info
https://blockchair.com/bitcoin
https://blockchain.info

Test Net Faucet and stats

https://tbtc.bitaps.com/



Last block

1 693 368

Time from last block

00 hours | 01 minutes | 35 seconds

58

unconfirmed transactions

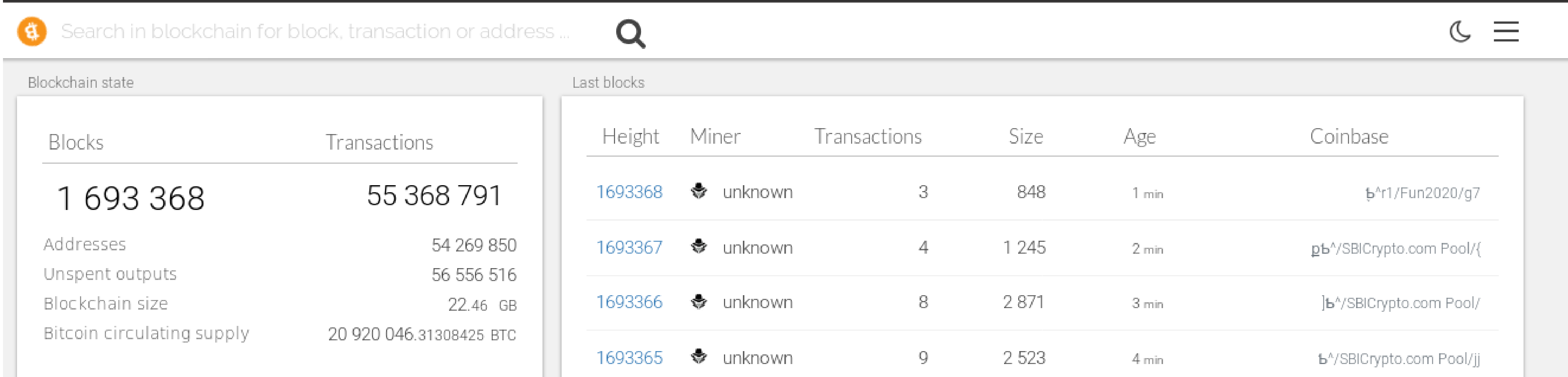
Bitcoin testnet faucet

Your testnet3 address

0.01 BTC

1 request per 5 minutes

Onetime limit 0.01 BTC



Blockchain state

Blocks	Transactions
1 693 368	55 368 791
Addresses	54 269 850
Unspent outputs	56 556 516
Blockchain size	22.46 GB
Bitcoin circulating supply	20 920 046.31308425 BTC

Last blocks

Height	Miner	Transactions	Size	Age	Coinbase
1693368	unknown	3	848	1 min	ƀ^r1/Fun2020/g7
1693367	unknown	4	1 245	2 min	pƀ^/SBICrypto.com Pool/{
1693366	unknown	8	2 871	3 min	]ƀ^/SBICrypto.com Pool/
1693365	unknown	9	2 523	4 min	ƀ^/SBICrypto.com Pool/jj

Blockchain.com API

https://blockchain.info/rawaddr/ + address

https://api.blockcypher.com/v1/btc/main/addrs/ + address + /balance

https://api.blockcypher.com/v1/btc/test3/addrs/ + address + /balance

Homework

1. Read Satoshi Bitcoin Paper and Bitcoin Wiki.
Prepare questions for discussion:

https://bitcoin.org/bitcoin.pdf
https://en.bitcoin.it/Transaction
https://en.bitcoin.it/wiki/Script

2. Join with 1-2 of your group mates in a team.

3. Install Copay.