About Me

How I became a Bitcoin Developer

- Joined the Seoul Bitcoin Meetup in 2017 (meetup.com/seoulbitcoin)

About Me

How I became a Bitcoin Developer

- Joined the Seoul Bitcoin Meetup in 2017 (meetup.com/seoulbitcoin)
- Started contributing in 2019

About Me

How I became a Bitcoin Developer

- Joined the Seoul Bitcoin Meetup in 2017 (meetup.com/seoulbitcoin)
- Started contributing in 2019
- Got funding in 2020

"The most important book about technology today, with implications that go far beyond programming."
—Guy Kawasaki

Revised & Expanded

THE CATHEDRAL & THE BAZAAR

MUSINGS ON LINUX AND OPEN SOURCE BY AN ACCIDENTAL REVOLUTIONARY

ERIC S. RAYMOND

WITH A FOREWORD BY BOB YOUNG, CHAIRMAN & CEO OF RED HAT, INC.

- No single roadmap

- No single roadmap
- Differing views on how to scale Bitcoin

- No single roadmap
- Differing views on how to scale Bitcoin
- I'll be sharing my take

- Anthony Towns - www.erisian.com.au/wordpress/2023/06/21/putting-the-b-in-btc
- James O’Berine - jameso.be/2023/08/14/scaling-bitcoin.html

Scaling Bitcoin with Utreexo

Current limitations and overcoming them

https://github.com/kcalvinalvin/2023-12-02-Taipei-Meetup

“We very, very much need such a system, but the way I understand your proposal, it does not seem to scale to the required size.”

https://www.mail-archive.com/cryptography@metzdowd.com/msg09963.html

What makes Bitcoin special

Something that can't be negotiated

Trustlessness

What is trustlessness?

Only trusting myself

“Of Bitcoin’s many properties, trustlessness, or the ability to use Bitcoin without trusting anything but the open-source software you run, is, by far, king”

https://bluematt.bitcoin.ninja/2017/02/28/bitcoin-trustlessness/

- Decentralization
- P2P
- Self-custody

Things that hurt trustlessness is rejected

Using Bitcoin Trustlessly

Run a full node

Trustless usage

- Full node = fully validating

Run a full node

Trustless usage

- Full node = fully validating
- Validating transactions and blocks

Validation Process

What all full nodes go through

- Download 548GB of blocks

Validation Process

What all full nodes go through

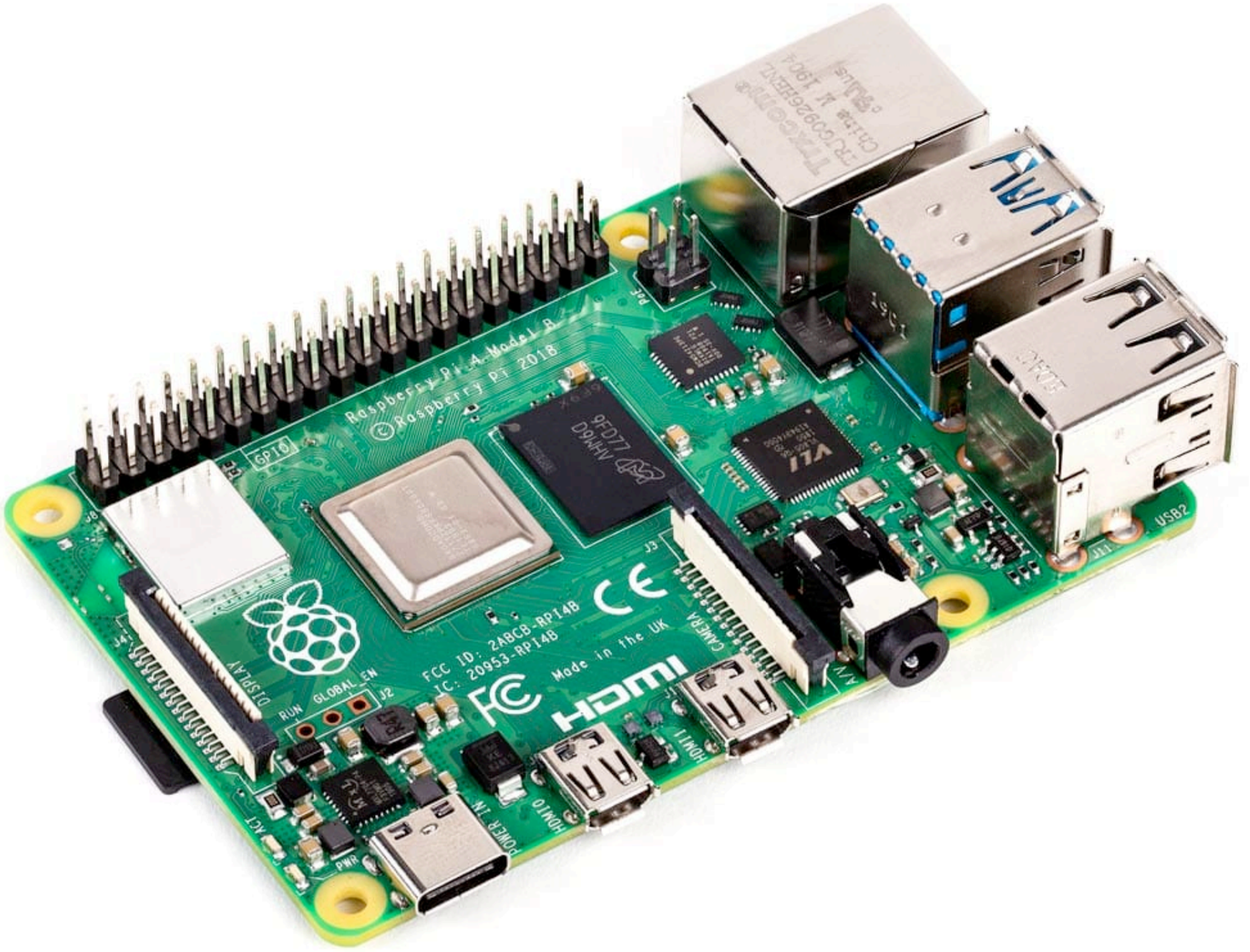
- Download 548GB of blocks
- Update the UTXO set every block

Validation Process

What all full nodes go through

- Download 548GB of blocks
- Update the UTXO set every block
- Validate the signature of the tx

Can everyone use it trustlessly?


Raspberry Pi 4 Model B
Raspberry Pi 2018
GPIO
FCC ID: 2ABCB-RPI4B
IC: 20953-RPI4B
Made in the UK
HDMI
RUN
GLOBAL_EN
DISPLAY
CAMERA
POWER IN
HDMI0
HDMI1
USB2
PWR
ACT

Things limiting scalability

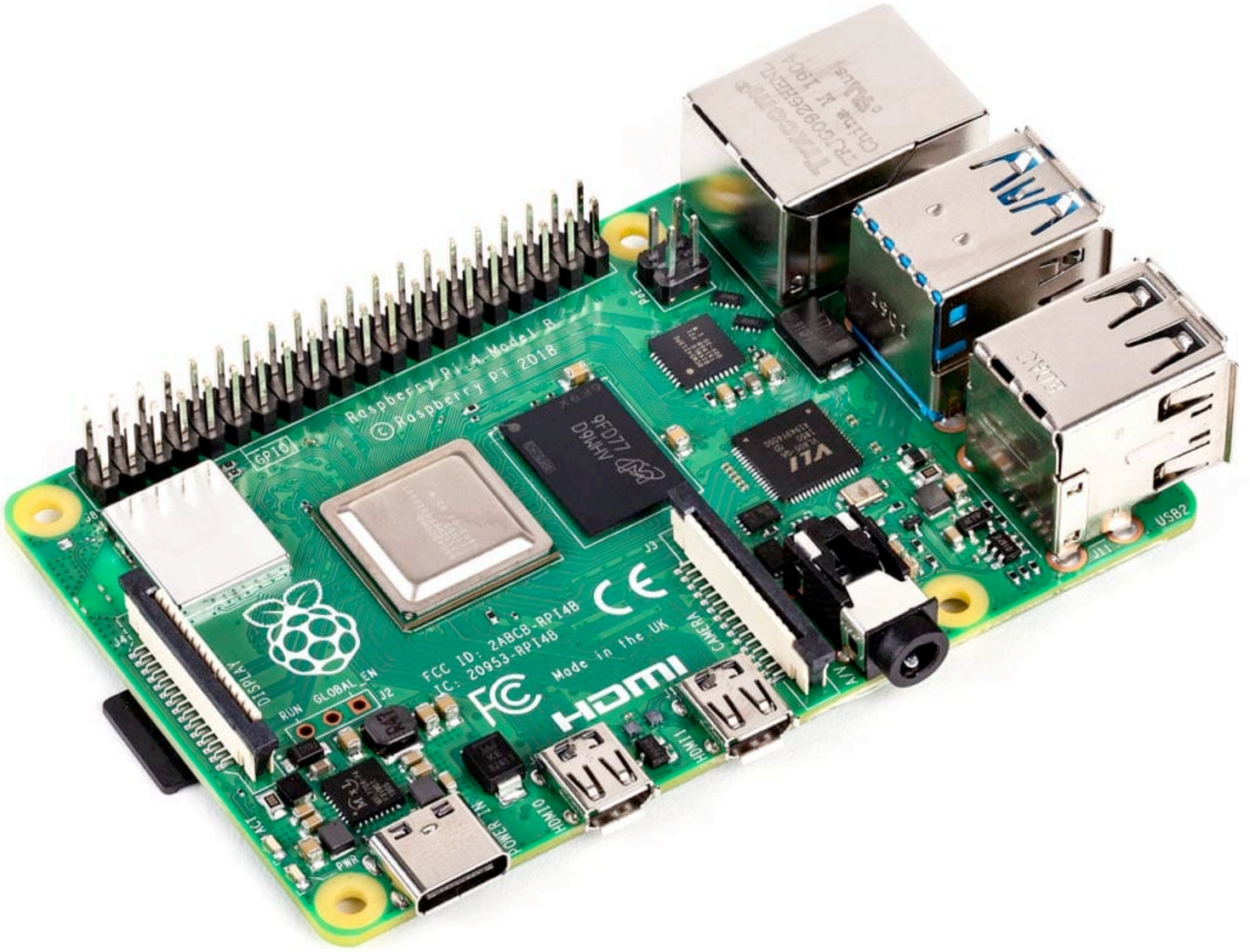
Scalability?

How much txs the system is able to process in a given time

~2MB block every 10minutes

2000-3000 transaction

Limited by the slowest computer


Raspberry Pi 4 Model B
Raspberry Pi 2018
GPIO
FCC ID: 2ABCB-RPI4B
IC: 20953-RPI4B
Made in the UK
HDMI
RUN
GLOBAL_EN
DISPLAY
CAMERA
POWER IN
HDMI0
HDMI1
USB2
PWR
ACT

How to scale Bitcoin

While keeping the trustlessness

- Use off-chain protocols

How to scale Bitcoin

While keeping the trustlessness

- Use off-chain protocols
- Increase the block size

Off-chain Protocols

The gist

- Increase the bang-for-the-buck

Off-chain Protocols

The gist

- Increase the bang-for-the-buck
- Multiple transfers within a single Bitcoin tx

Protocols with security tradeoffs

Off-chain Protocols

- Liquid Network

Protocols with security tradeoffs

Off-chain Protocols

- Liquid Network
- Statechains

Protocols with security tradeoffs

Off-chain Protocols

- Liquid Network
- Statechains
- Fedimint/cashu

Protocols without security tradeoffs

Off-chain Protocols

- Lightning Network

Lightning Network

UX downsides during a high fee environment

- Need to make an expensive on-chain tx when getting started

Lightning Network

UX downsides during a high fee environment

- Need to make an expensive on-chain tx when getting started
- Need to make an expensive on-chain tx when there's liquidity issues

Protocols without security tradeoffs

Off-chain Protocols

- Lightning Network
- Ark

Ark

UX downsides during a high fee environment

- Exiting the Ark system is tricky when the fee is higher than the value of the VTXO

High fees make L2s suck

My highly opinionated conclusion

Eventually increase the block size

Currently blocked by the technical reasons

Why can't we increase the block size

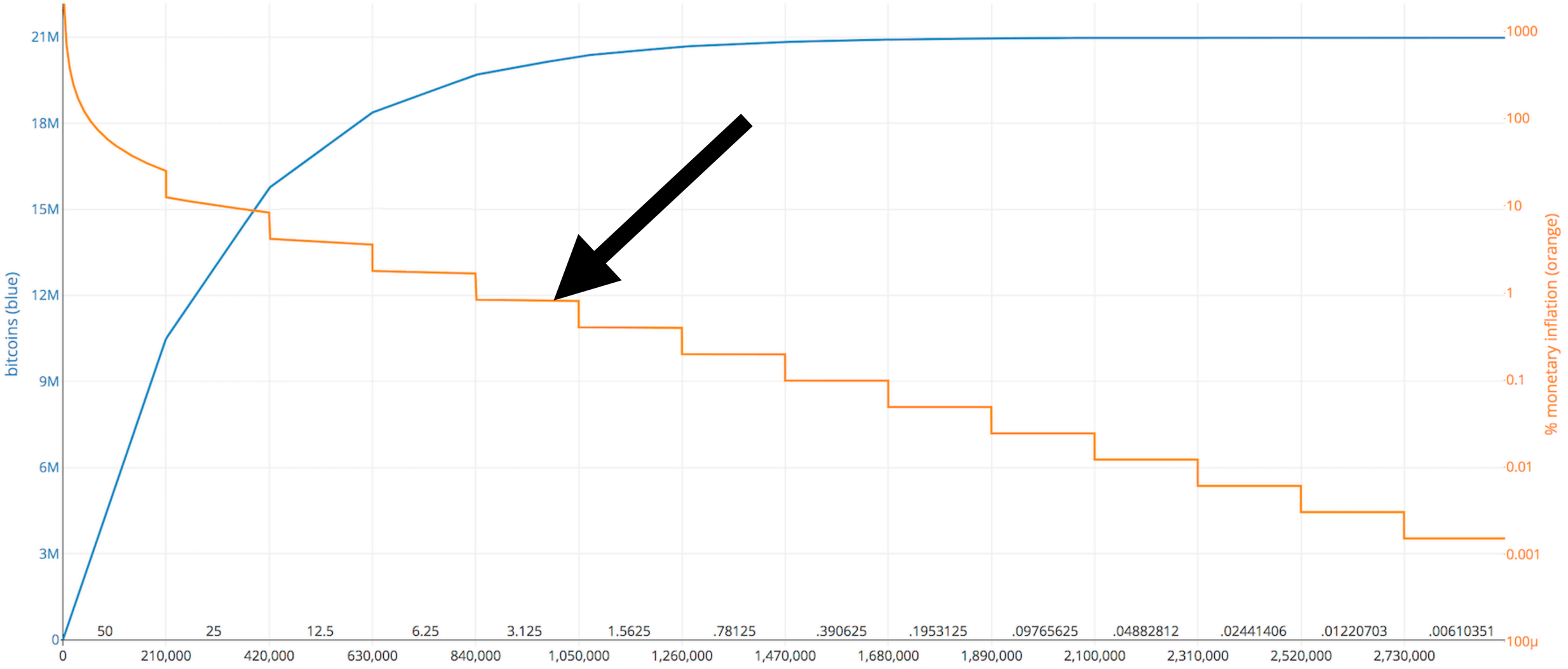
2 reasons

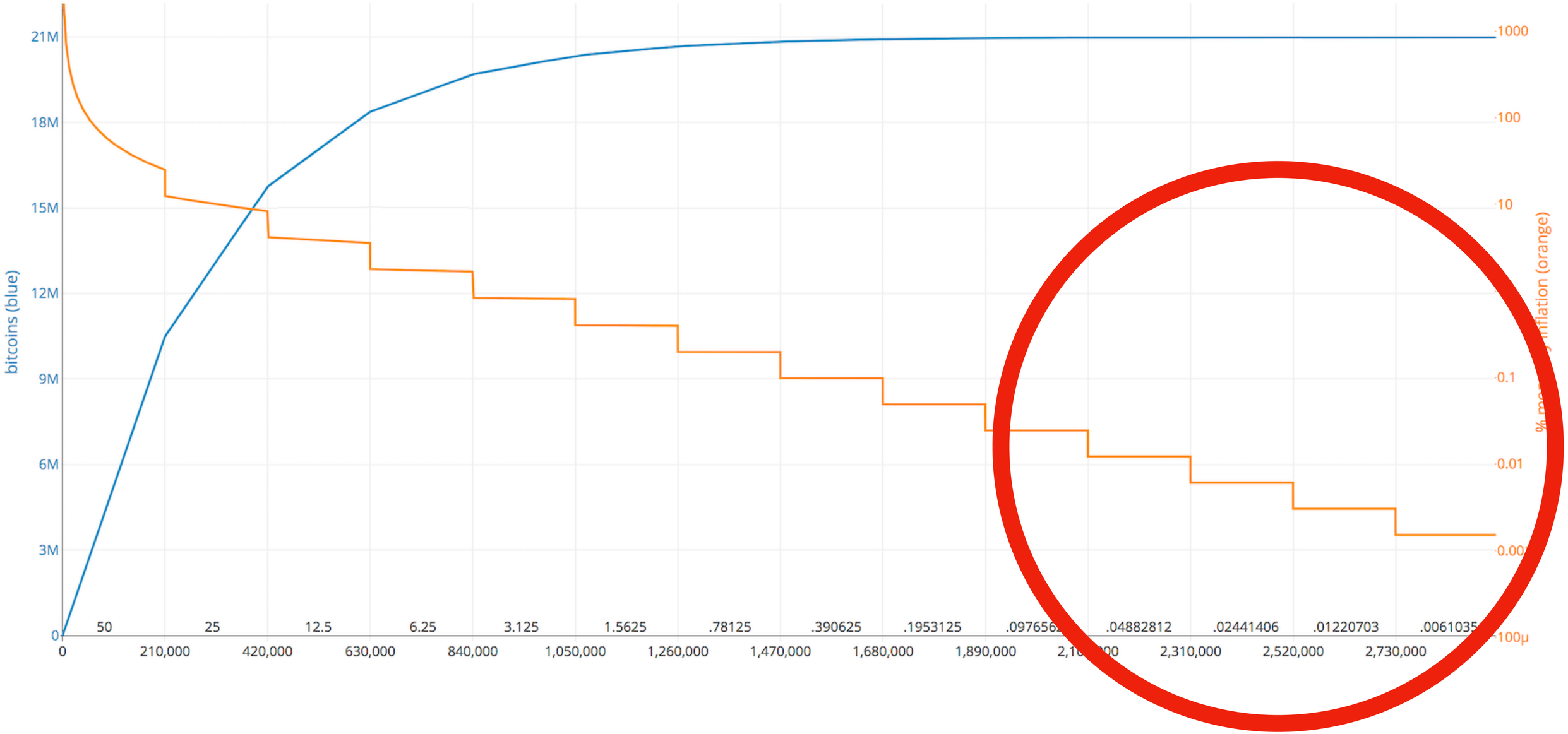
- Economical
- Technical

Economical reasons

Supply and demand

- Are there going to be enough fees?


21M
18M
15M
12M
9M
6M
3M
0
bitcoins (blue)
1000
100
10
1
0.1
0.01
0.001
100μ
% monetary inflation (orange)
50
25
12.5
6.25
3.125
1.5625
.78125
.390625
.1953125
.09765625
.04882812
.02441406
.01220703
.00610351
0
210,000
420,000
630,000
840,000
1,050,000
1,260,000
1,470,000
1,680,000
1,890,000
2,100,000
2,310,000
2,520,000
2,730,000


21M
18M
15M
12M
9M
6M
3M
0
bitcoins (blue)
1000
100
10
0.1
0.01
100µ
% inflation (orange)
50
25
12.5
6.25
3.125
1.5625
.78125
.390625
.1953125
.04882812
.02441406
.01220703
0
210,000
420,000
630,000
840,000
1,050,000
1,260,000
1,470,000
1,680,000
1,890,000
2,310,000
2,520,000
2,730,000

Economical reasons

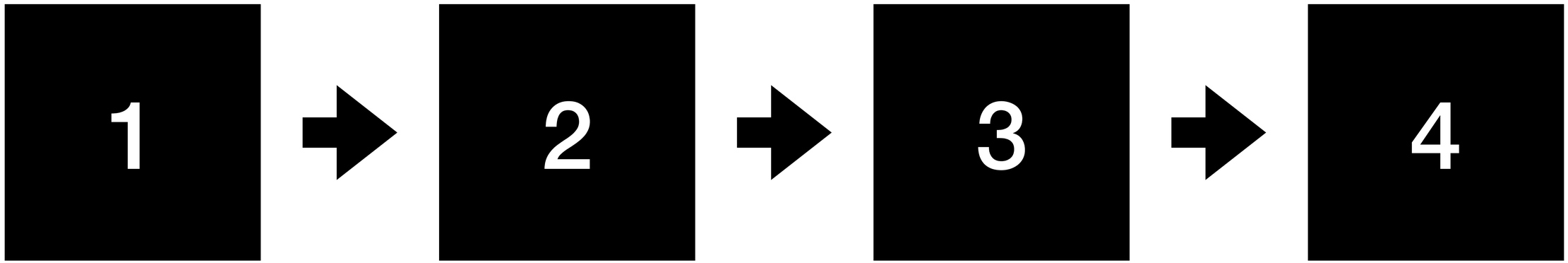
Supply and demand

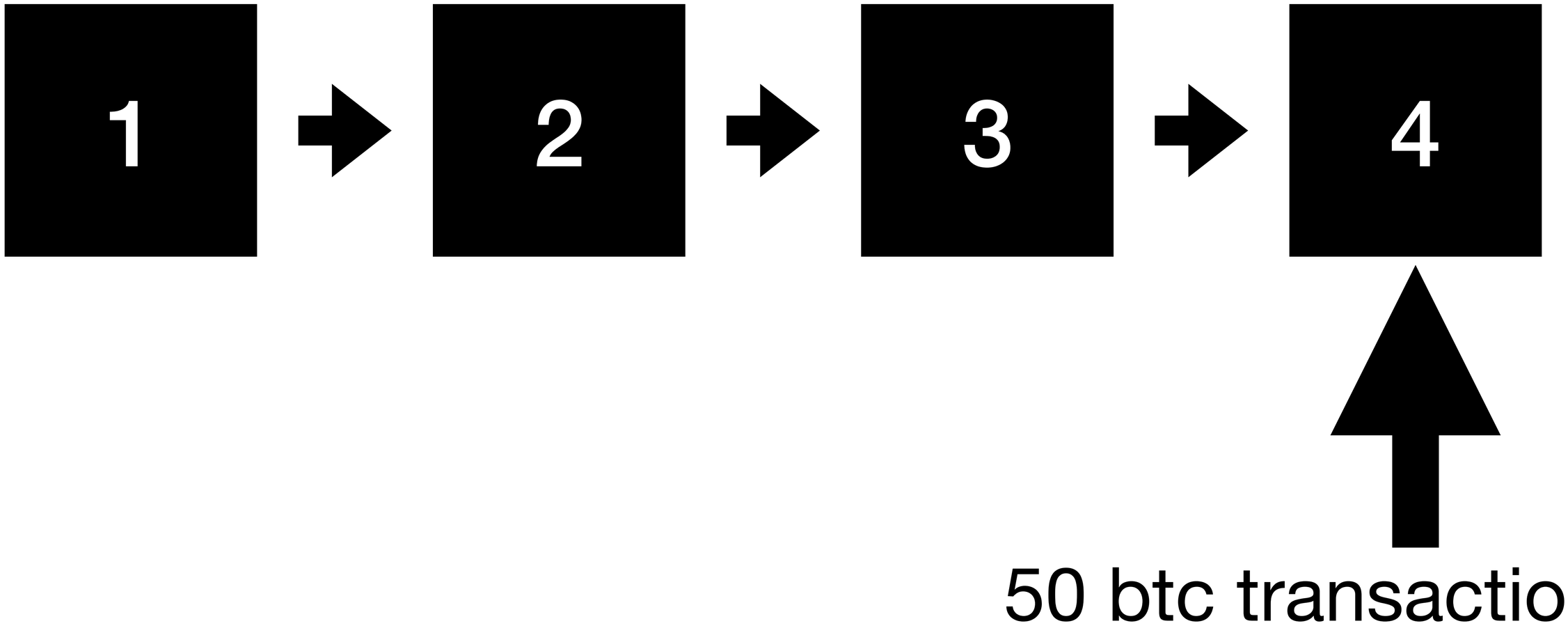
- Current double spend deterrents are mostly from the block rewards

Economical reasons

Supply and demand

- Current double spend deterrents are mostly from the block rewards
- As rewards go down, fees have to go up

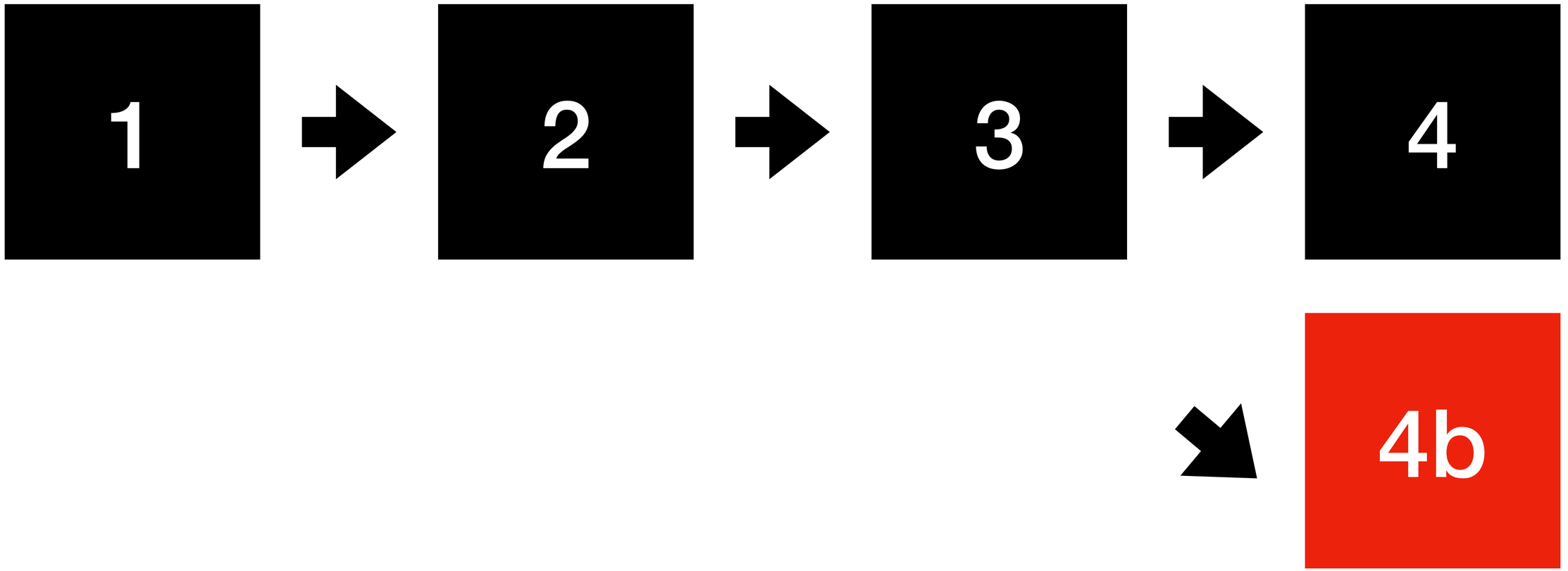

1
2
3
4

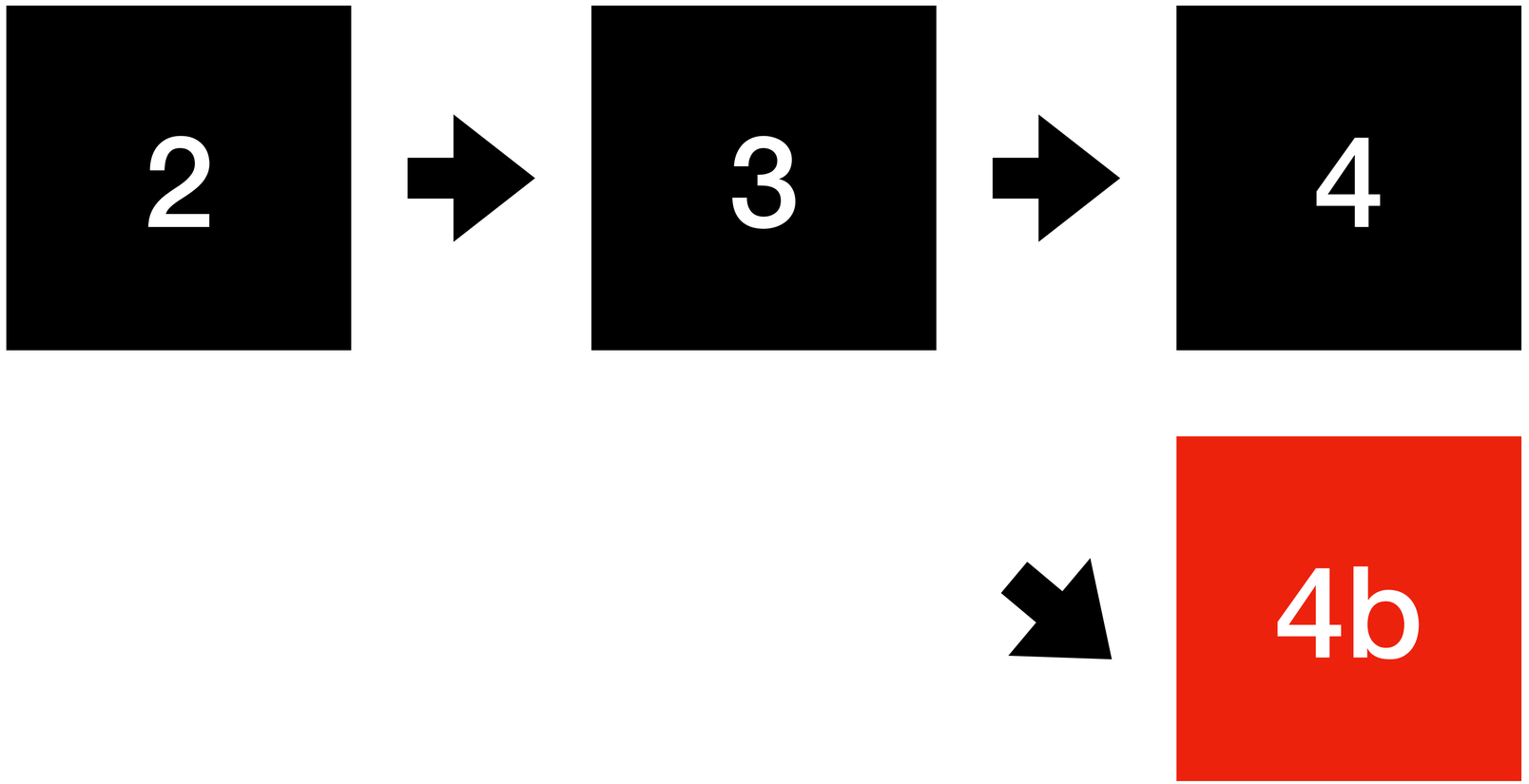

1
2
3
4
50 btc transaction

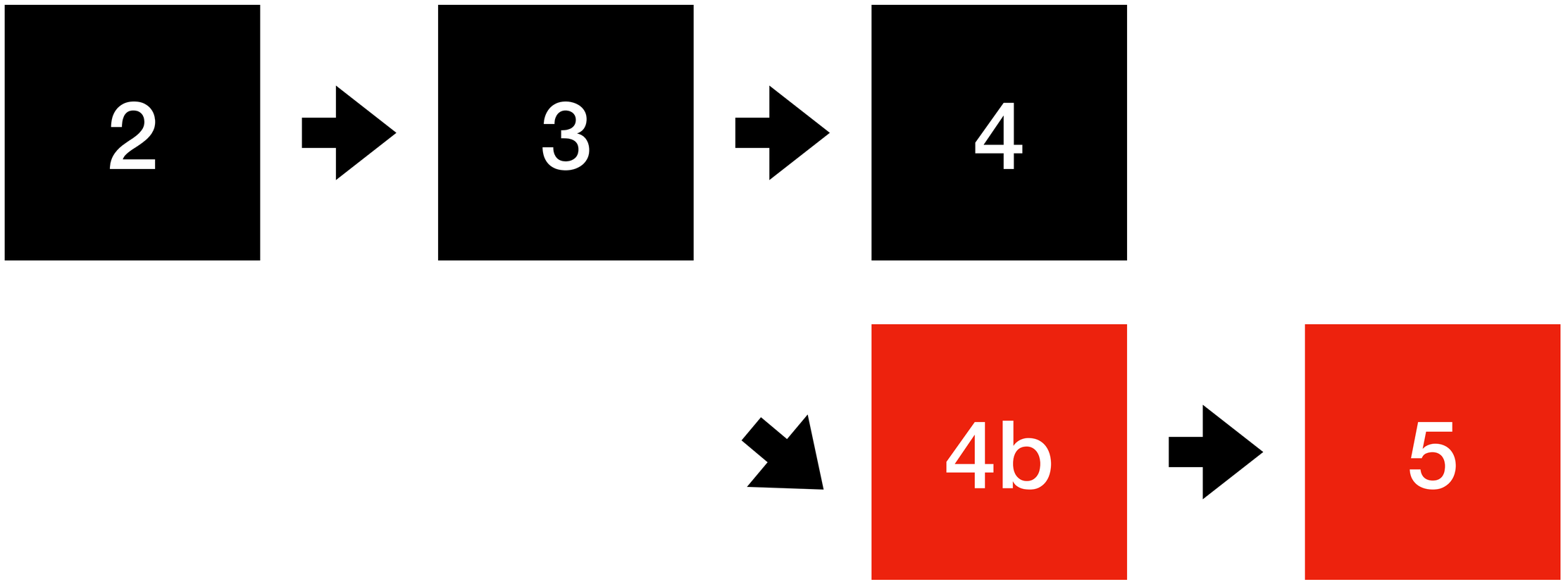
- Alice buys 50 btc with gold

- Alice buys 50 btc with gold
- To revert the 50 btc tx, Alice bribes miner Bob with 10 btc

- Alice buys 50 btc with gold
- To revert the 50 btc tx, Alice bribes miner Bob with 10 btc
- Alice's profit - 40 btc + gold
- Bob's profit - 10 btc + block rewards


1
2
3
4
4b


2
3
4
4b


2
3
4
4b
5

Economical reasons

Supply and demand

- Enough fees need to exist to secure Bitcoin

Economical reasons

Supply and demand

- Enough fees need to exist to secure Bitcoin
- Blocksize ↑, Fees ↓

No research ongoing for economic reasons

Technical reasons

Trustlessness

- Can everyone use Bitcoin trustlessly?

Technical reasons

Trustlessness

- Can everyone use Bitcoin trustlessly?
- Blocksize ↑, Cost of running full nodes ↑

Technical reasons

Process of Validation

- Download ~548GB of data
- Update the UTXO set per block
- Validate the signature of the tx

Technical reasons

Process of Validation

- Internet
- Disk
- CPU

Internet

Block / Transaction Validation

- Download ~500GB on the first start

Internet

Block / Transaction Validation

- Download ~500GB on the first start
- 100mbps Internet = ~666 minutes

Internet

Block / Transaction Validation

- Download ~500GB on the first start
- 100mbps Internet = ~666 minutes
- 1gbps = ~66 minutes

Internet

Block / Transaction Validation

- Download ~500GB on the first start
- 100mbps Internet = ~666 minutes
- 1gbps = ~66 minutes
- 10gbps = ~6 minutes

Internet speeds are enough

Disk

UTXO set read/writes

- UTXO set - All the Bitcoin available to be spent

What's a UTXO set?

- TXO - Transaction Output

Typical TX

https://blockstream.info/tx/8ad63210d059908229f00ad177469d9f16a5d37a7ea64cede9c92684ff7cf06e

Outputs of a TX

https://blockstream.info/tx/8ad63210d059908229f00ad177469d9f16a5d37a7ea64cede9c92684ff7cf06e

What's a UTXO set?

- TXO - Transaction Output
- UTXO - Unspent TXO. Not yet spent TXO

What's a UTXO set?

- TXO - Transaction Output
- UTXO - Unspent TXO. Not yet spent TXO
- UTXO set - All the UTXOs

Disk

UTXO set read/writes

- UTXO set is stored as a key-value database (leveldb)

Disk

UTXO set read/writes

- UTXO set is stored as a key-value database (leveldb)
- leveldb is random read/write

Disk

UTXO set read/writes

- Minimum 59B per UTXO

Outpoint: 33B, p2pkh script: 21B, height+iscoinbase: 4, amount: 1

Disk

UTXO set read/writes

- Minimum 59B per UTXO
- Typical tx is 1 input, 2 output

Outpoint: 33B, p2pkh script: 21B, height+iscoinbase: 4, amount: 1

Disk

UTXO set read/writes

- Minimum 59B per UTXO
- Typical tx is 1 input, 2 output
- For 2000txs: ~2000 read, ~4000 write

Outpoint: 33B, p2pkh script: 21B, height+iscoinbase: 4, amount: 1

기술의 경이로움이 시작되다

Platinum P41 SSD

科技的奇蹟開始了

SK하이닉스 Platinum P41 SSD는 업계 최고 수준의 성능으로
차세대 컴퓨팅 시스템을 완성하는 제품입니다. PCIe 4세대 NVMe 인터페이스와
176단 NAND의 한층 더 강화된 기술력을 토대로 최강의 데이터 전송 속도와
성능을 제공합니다. 이제 차원이 다른 성능을 경험해 보세요.

성능

	2TB	1TB	500GB
순차성능 (최대)	읽기 : 7,000 MB/s 쓰기 : 6,500 MB/s	읽기 : 7,000 MB/s 쓰기 : 6,500 MB/s	읽기 : 7,000 MB/s 쓰기 : 4,700 MB/s
랜덤성능 (최대)	읽기 : 1,400K IOPS 쓰기 : 1,300K IOPS	읽기 : 1,400K IOPS 쓰기 : 1,300K IOPS	읽기 : 960K IOPS 쓰기 : 1,000K IOPS
TBW	최대 1,200TBW	최대 750TBW	최대 500TBW

- 1,400 IOPS= ~5.6MB/s read
- 1,300 IOPS= ~5.2MB/s write

(1,400 * 4) / 1000. 4 is the OS page size (mb)

- 1,400 IOPS= ~5.6MB/s read
- 1,300 IOPS= ~5.2MB/s write
- Per block, 0.11MB read, 0.23MB write

(1,400 * 4) / 1000. 4 is the OS page size (mb)


WD_BLACK™
GAMING HARD DRIVE
www.wdblack.com
8TB
S/N:ACO8JNAL
MDL:WD8002FZWX-00BKUA0
WD8002FZWX
SATA 6Gb/s PC HA710
Made in Thailand
5VDC: 0.42A 12VDC: 0.60A
14 MAR 2023
R/N: 810032
WD, PO Box 13379, Swords, Co Dublin, Ireland
WD UK, PO Box 471, Leatherhead, KT22 2LU, UK
WWN:50014EE2T6A60092
CAN ICES-3 (B)/NMB-3(B)
R-R-WDT-810032
Western Digital®
DO NOT COVER
ANY DRIVE HOLES

- 73 IOPS= ~0.29MB/s read&write
- Per block 0.11MB read, 0.23MB write

https://en.wikipedia.org/wiki/IOPS

Caching

Current way of reducing the disk burden

- Write is fast (only to RAM)

Caching

Current way of reducing the disk burden

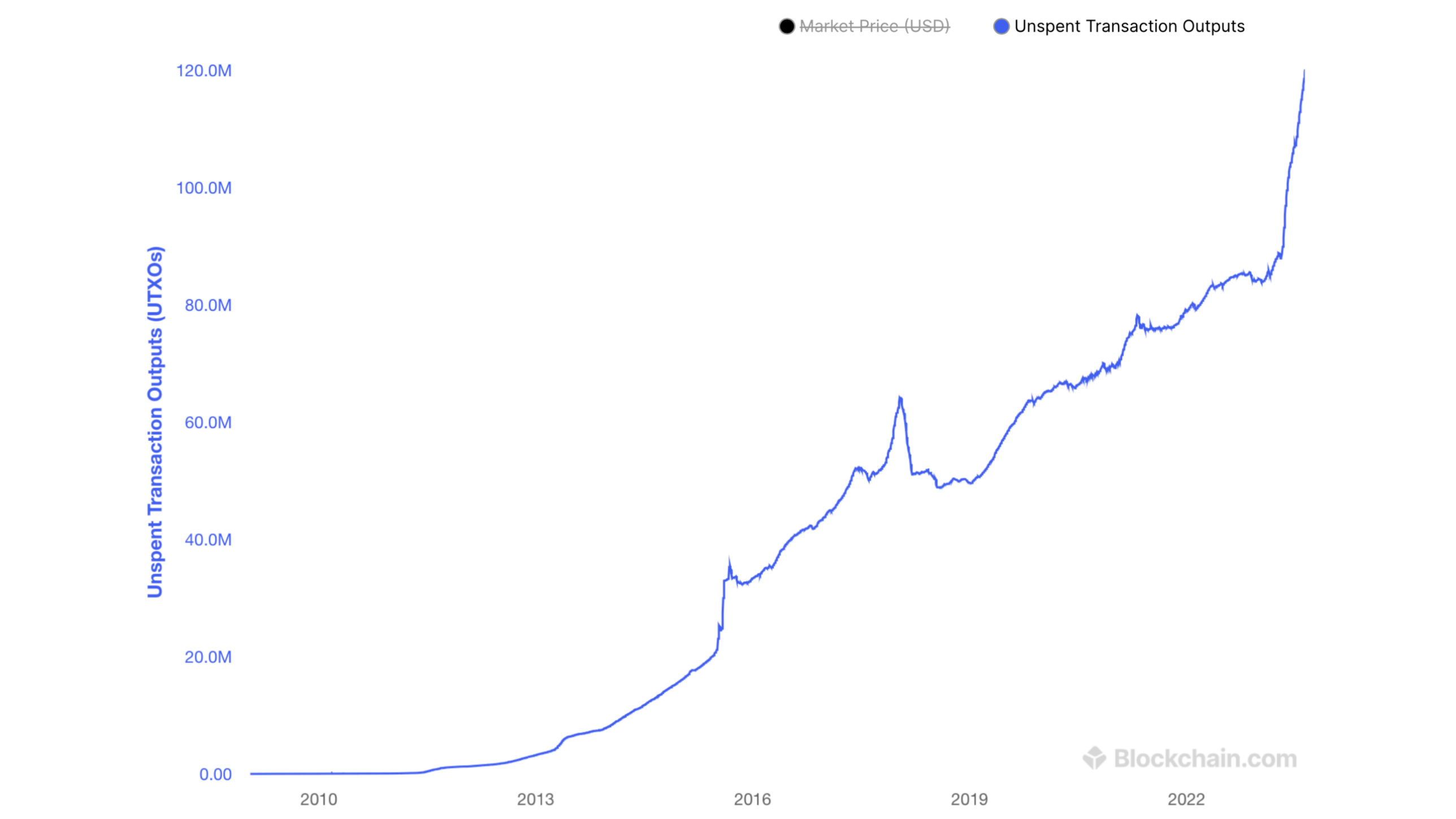
- Write is fast (only to RAM)
- Read is sped-up (if cached)

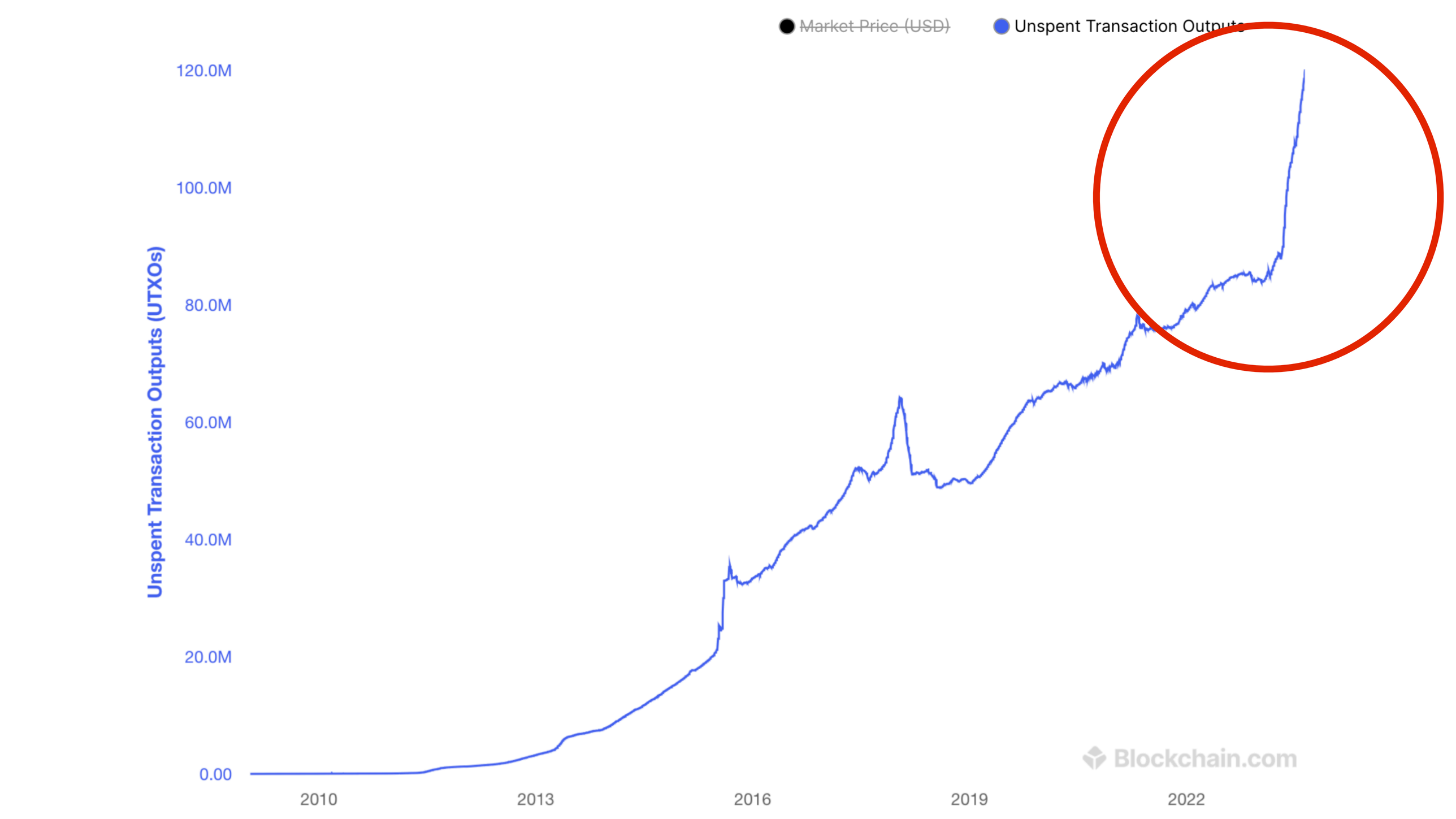
Caching

Current way of reducing the disk burden

- As the UTXO set grows, the RAM size limit will make optimization less effective

Outpoint: 33B, p2pkh script: 21B, height+iscoinbase: 4, amount: 1


Market Price (USD)
Unspent Transaction Outputs
120.0M
100.0M
80.0M
60.0M
40.0M
20.0M
0.00
Unspent Transaction Outputs (UTXOs)
2010
2013
2016
2019
2022
Blockchain.com


Market Price (USD)
Unspent Transaction Outputs
Unspent Transaction Outputs (UTXOs)
120.0M
100.0M
80.0M
60.0M
40.0M
20.0M
0.00
2010
2013
2016
2019
2022
Blockchain.com

As the UTXO set grows, it gets slower

UTXO set size has no bounds

8 billion UTXO, 59B each

472GB

Outpoint: 33B, p2pkh script: 21B, height+iscoinbase: 4, amount: 1

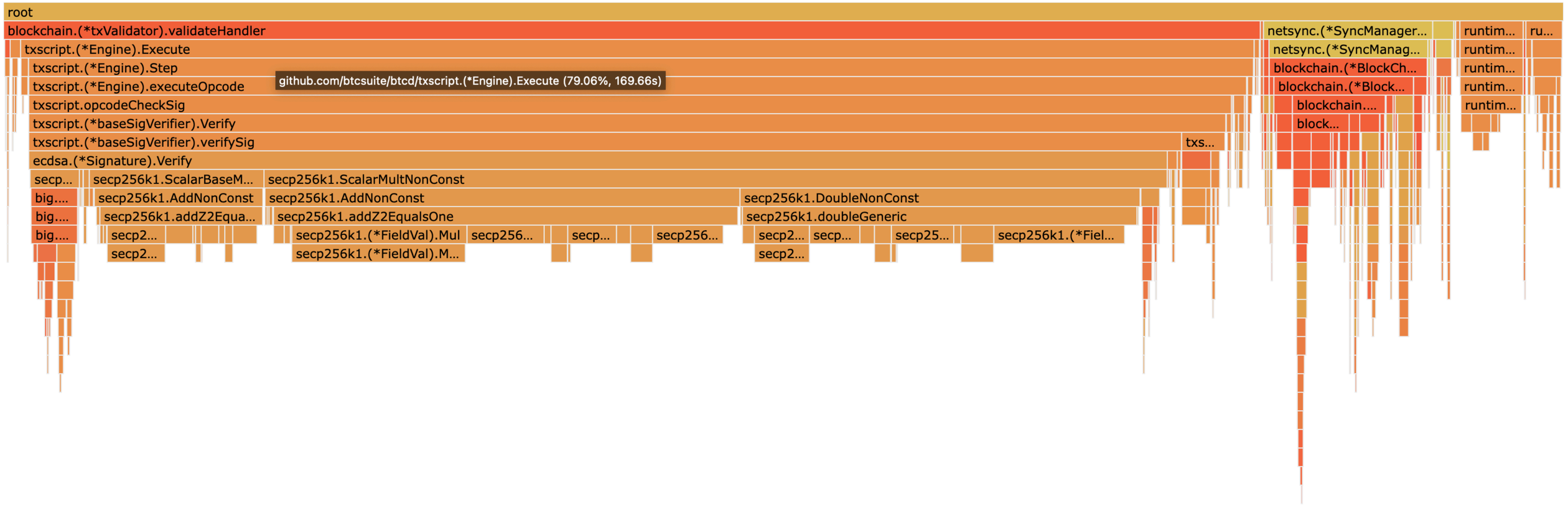
Already bottleneck for HDD

CPU

Transaction signature validation

- Signature operations take up a bulk of the block validation

github.com/btcsuite/btcd/txscript.(*Engine).Execute (79.06%, 169.66s)



80% time on sig validation

For btcd

CPU

Transaction signature validation

- CPU cores increasing
- Taproot txs can be batch verified

README.md

libsecp256k1

Optimized C library for ECDSA signatures and secret/public key operations on curve secp256k1.

This library is intended to be the highest quality publicly available library for cryptography on the secp256k1 curve. However, the primary focus of its development has been for usage in the Bitcoin system and usage unlike Bitcoin's may be less well tested, verified, or suffer from a less well thought out interface. Correct usage requires some care and consideration that the library is fit for your application's purpose.

Features:

- secp256k1 ECDSA signing/verification and key generation.
- Additive and multiplicative tweaking of secret/public keys.
- Serialization/parsing of secret keys, public keys, signatures.
- Constant time, constant memory access signing and public key generation.
- Derandomized ECDSA (via RFC6979 or with a caller provided function.)
- Very efficient implementation.
- Suitable for embedded systems.
- No runtime dependencies.
- Optional module for public key recovery.
- Optional module for ECDH key exchange.
- Optional module for Schnorr signatures according to BIP-340.

Already CPU bottlenecked but rooms for improvement

What do we need to increase the block size?

Signature validation ↑
UTXO read/write ↑

How to improve read/write?

Utreexo

The layer 1 scaling solution!

Merklize the UTXO set


472GB
Utreexo
Magic
Box
1KB

~Same performance on HDD and SSD

disk read/write X

No more random reads

Testing the performance

Test methodology

- Use an option that does a lot of read/writes and compare

36% slower

On current nodes

https://blog.bitmex.com/progress-towards-utreexo-goals/

3% slower

On Utreexo nodes

https://blog.bitmex.com/progress-towards-utreexo-goals/

The writes that happen for Utreexo

No reads

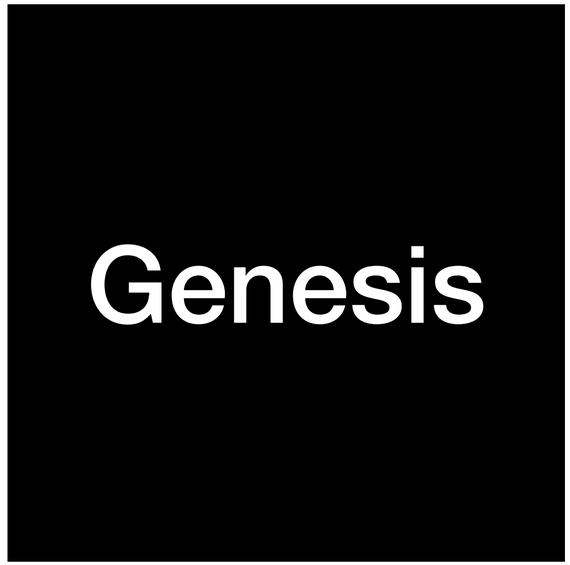
```
	{443000, newHashFromStr("00000000000000000047efc3c126c3398cfdef1609682d1492cd2909e3c0a17f"),
		43243051, []*chainhash.Hash{
			newLeafHashFromStr("df31c496a54a8021c38c77809da15a8bd3968970315f5a56c07f0fd9763262ff"),
			newLeafHashFromStr("755f4cfbf5727c3e14aaf6dcc6e5ab150c61101cead06fade5a886ee8d1dd377"),
			newLeafHashFromStr("c572a92c7cd9e14ce5d7c94473ef90bee21856654eed5535314b3c117006bc83"),
			newLeafHashFromStr("9e85567347ef0ea38c1797d8ed8e399ed0d99a07025388f6e06791ff6b88643c"),
			newLeafHashFromStr("ab4efc738bd7b8281689e899979d44ed0ca8fe1489efa0f5a09f5b1c276fe792"),
			newLeafHashFromStr("1c763d8f490baef0b756597dfc2da5a31d0aee324da6314b52f21707351cf51a"),
			newLeafHashFromStr("29fcd9989939a17e5d113fb1f247f3bd05606488bcb48c9cfad55a57006f2248"),
			newLeafHashFromStr("e79f840fe1bcdbb6b638a53b637885dda7ea3da2b2b5f1f5447ebf4d54fc0a2e"),
			newLeafHashFromStr("02f33c3ce0d684e644410fa35fee4c7572454facc710389ea0f69a00e27c1780"),
			newLeafHashFromStr("4e29ad940088ab42991bd99d7943a68ea38d498f8435c4bea21b8252788b5a7d"),
			newLeafHashFromStr("feabfc6565e918827df8ade12d0885e7da28bb81410c71a0fcbd92da28e5538e"),
			newLeafHashFromStr("42275481edcd150c0891f8f7486941e7ecc9dbec0d1554d1ae3027d8523571b4"),
			newLeafHashFromStr("84dd84bbb564ab9e878ea774d66a2cb818c2b74680974e01a07dde68bd064bfd"),
			newLeafHashFromStr("19a64f2f77403d4a1e3482b3a1801aaf360120f6429f5ed47261fc8676920d9d"),
		},
	},
```

https://github.com/mit-dci/utcd/blob/7310b99094acbeb11ecda230475d127af064c82e/chaincfg/mainnetroots.go#L6665-L6682

Out of Order Block Validation

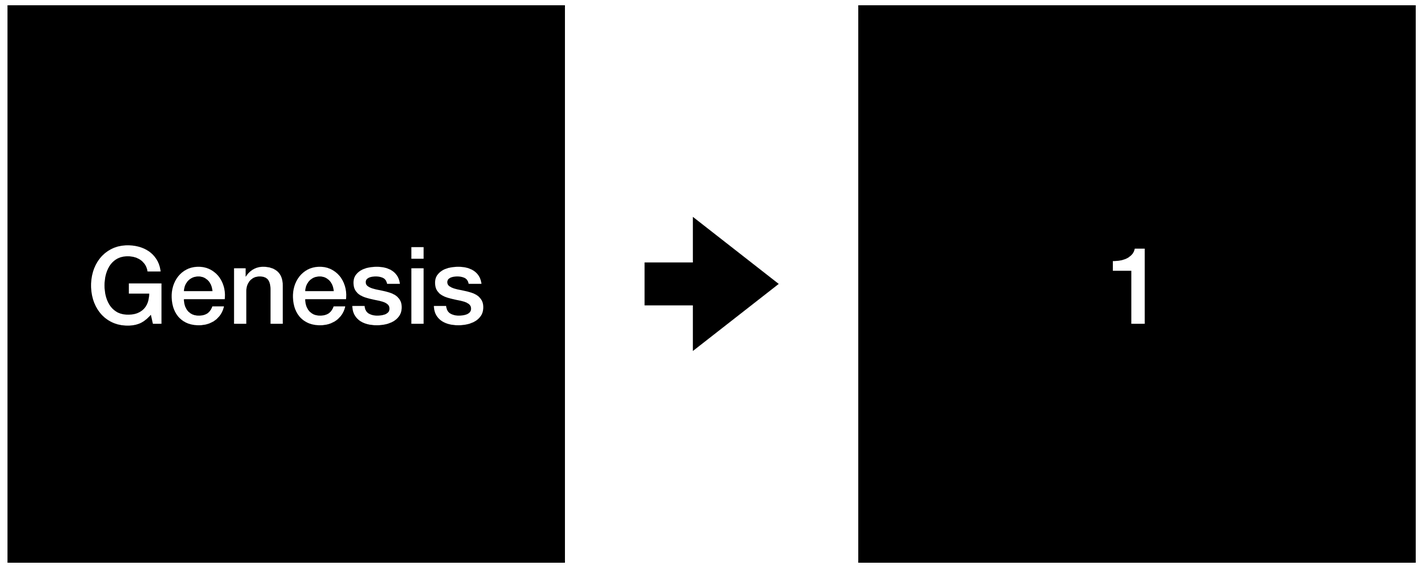
Current Block Validation

Sequential



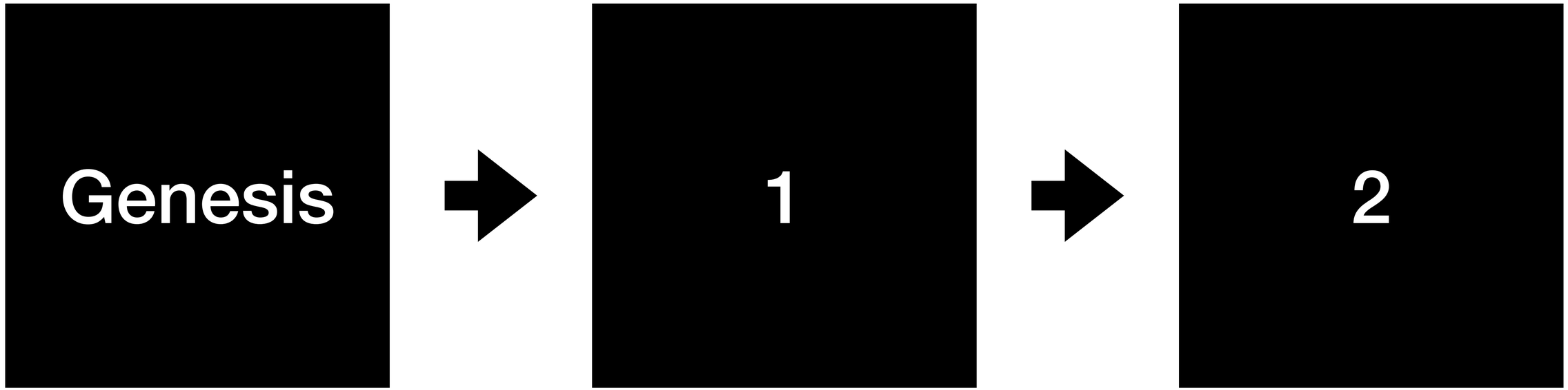
Current Block Validation

Sequential



Current Block Validation

Sequential



Out of Order Block Validation

Doesn't have to be sequential

- Provide the merkle roots of the Utreexo at each block

- https://github.com/mit-dci/utcd/blob/master/chaincfg/mainnetroots.go

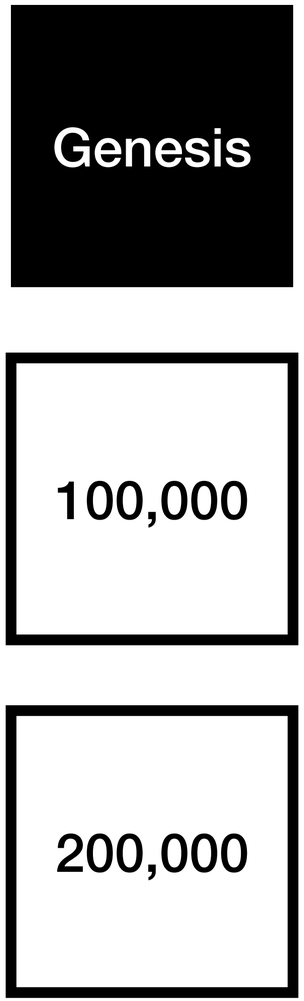
Out of Order Block Validation

Doesn't have to be sequential

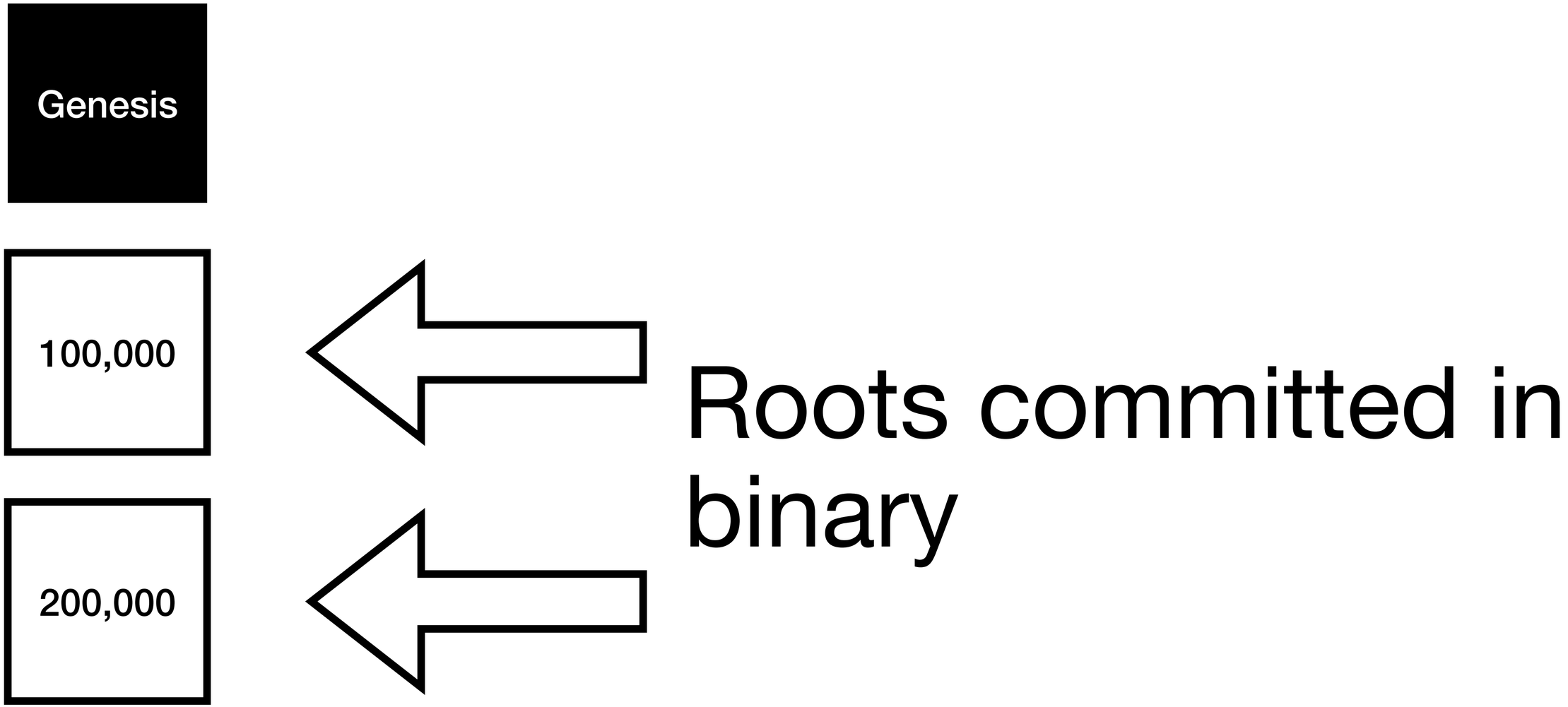
- Provide the merkle roots of the Utreexo at each block
- Use them to validate the next block in line

- https://github.com/mit-dci/utcd/blob/master/chaincfg/mainnetroots.go

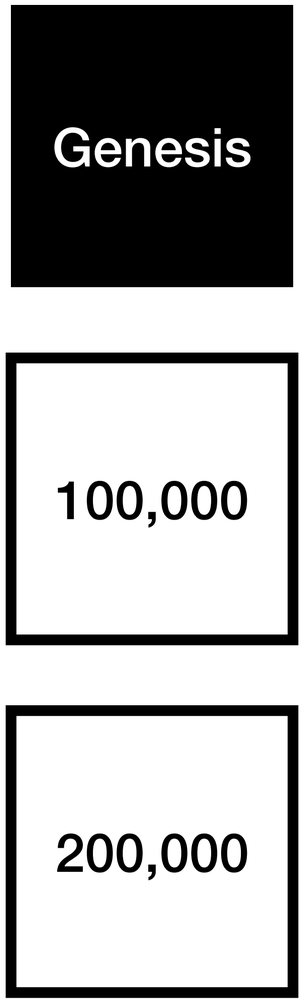
Out of Order Validation



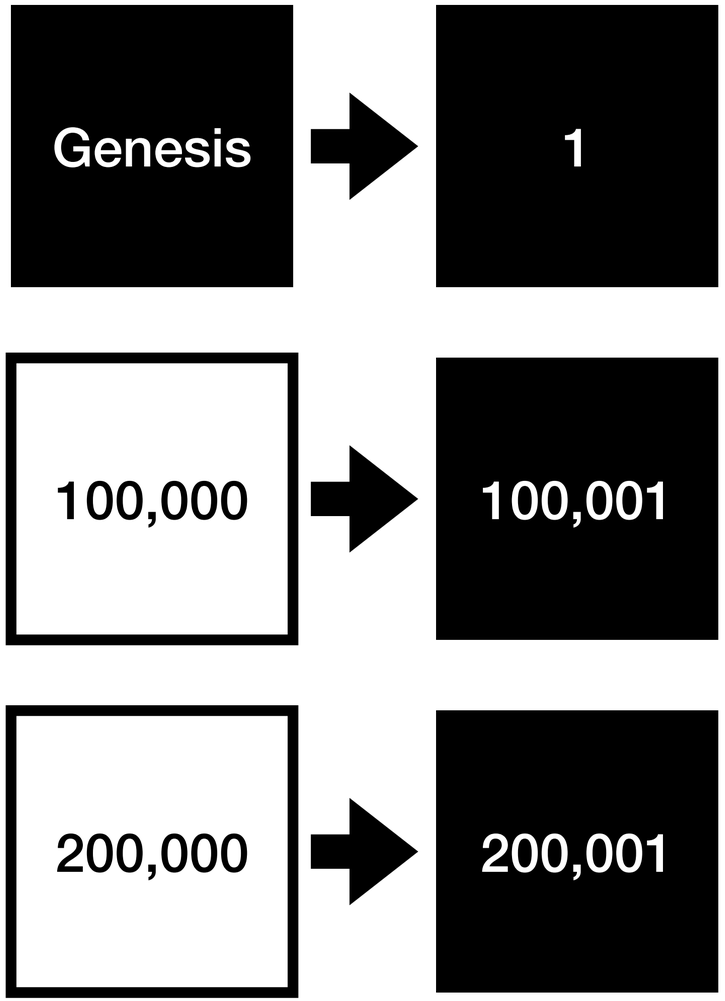
Out of Order Validation



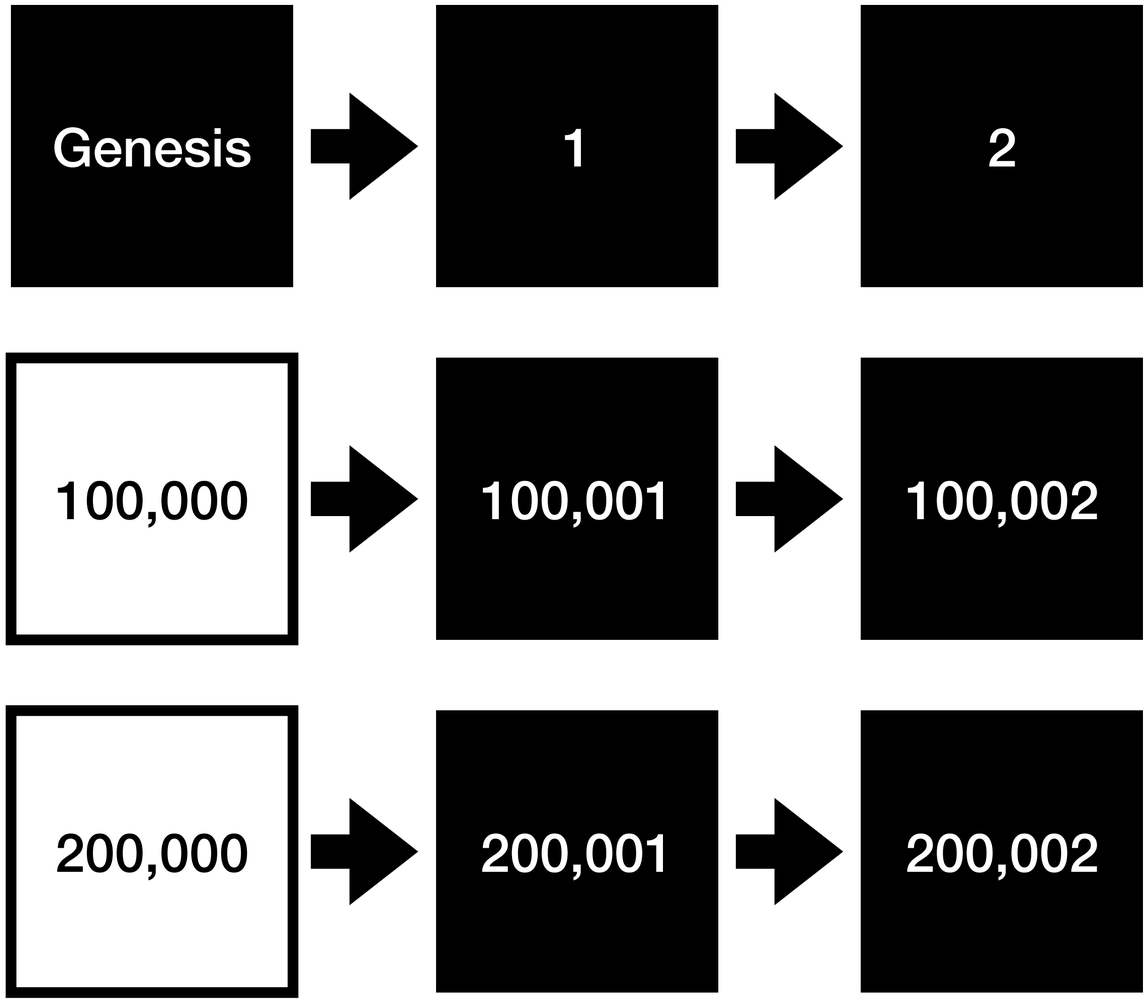
Out of Order Validation



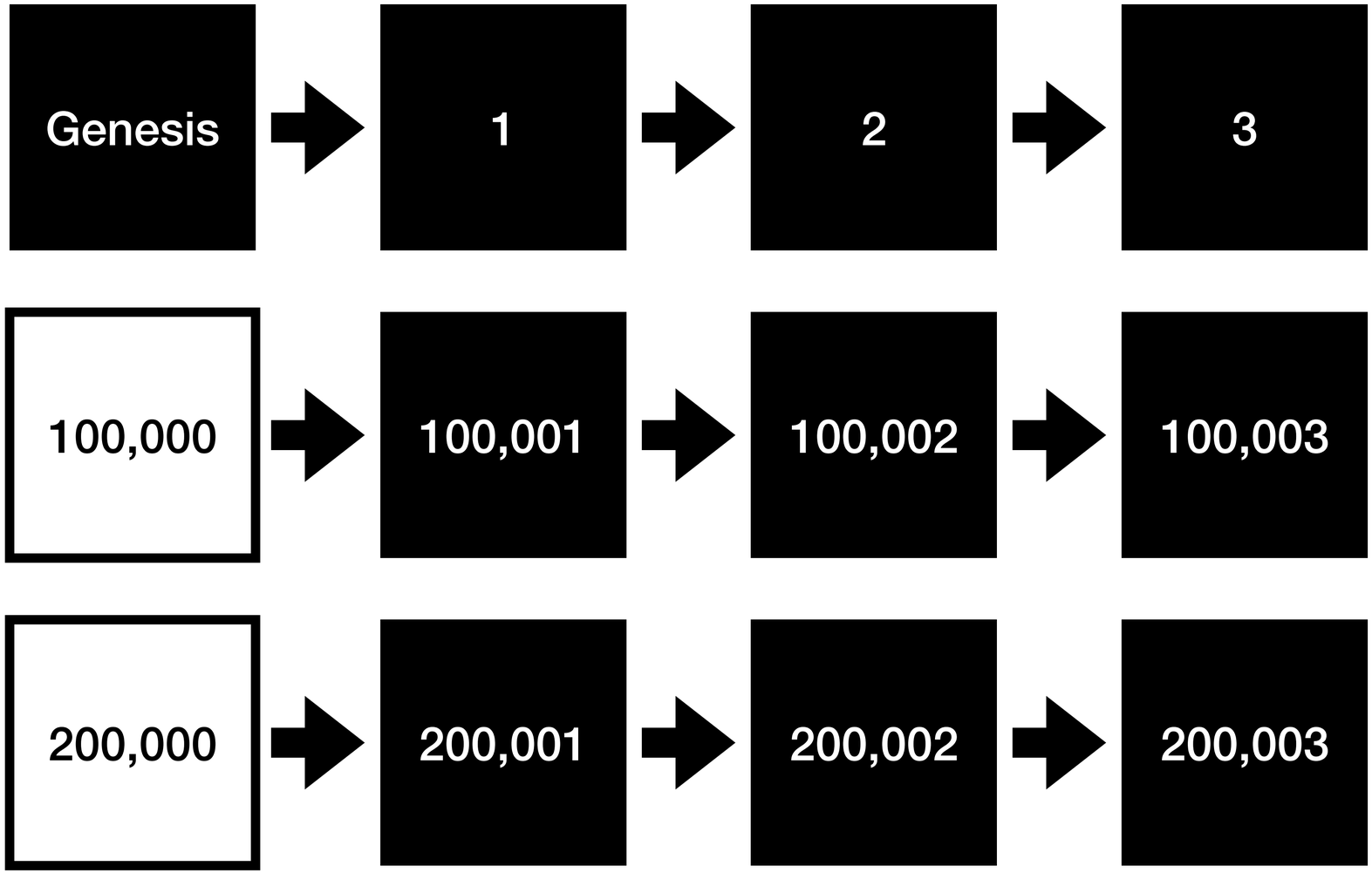
Out of Order Validation



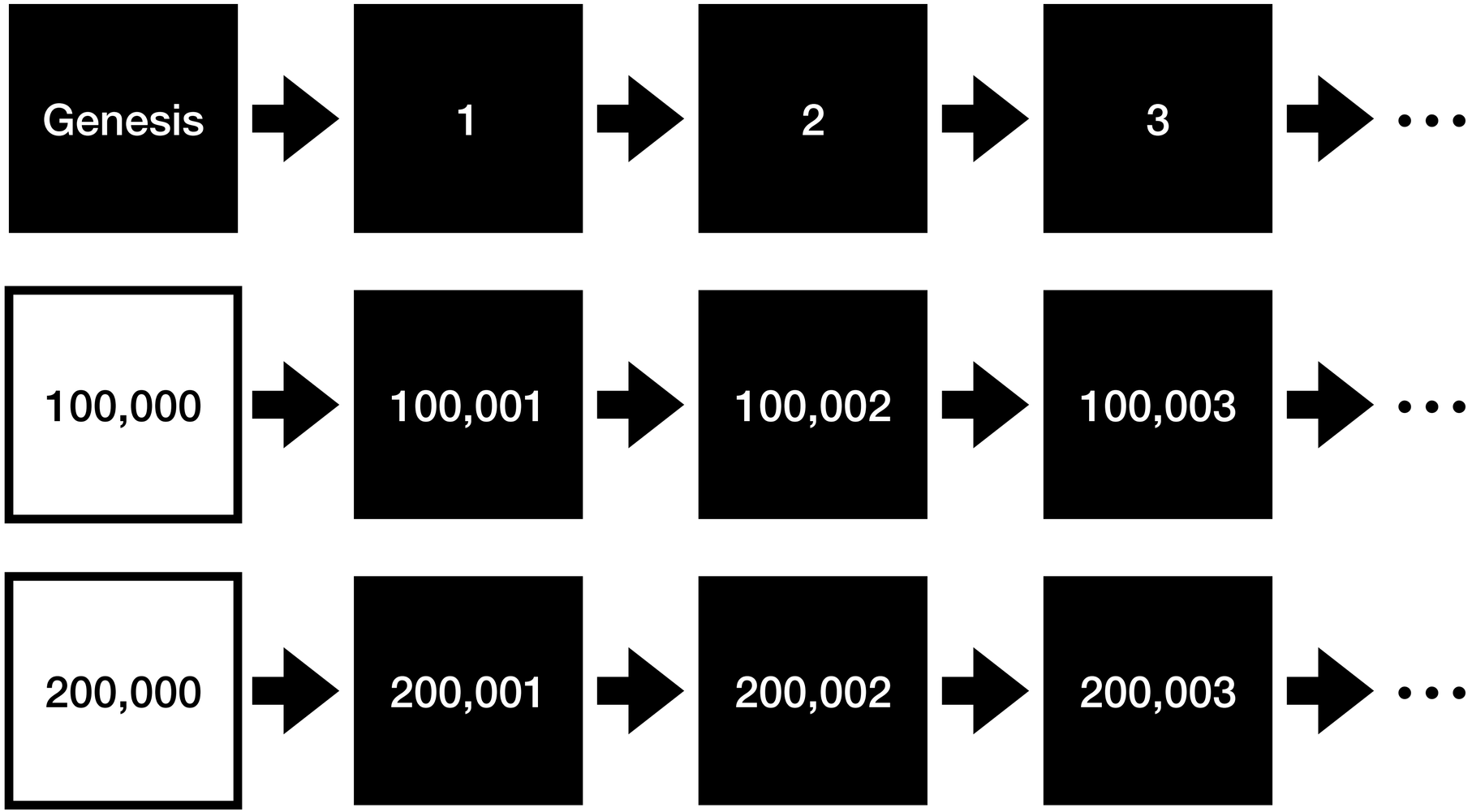
Out of Order Validation



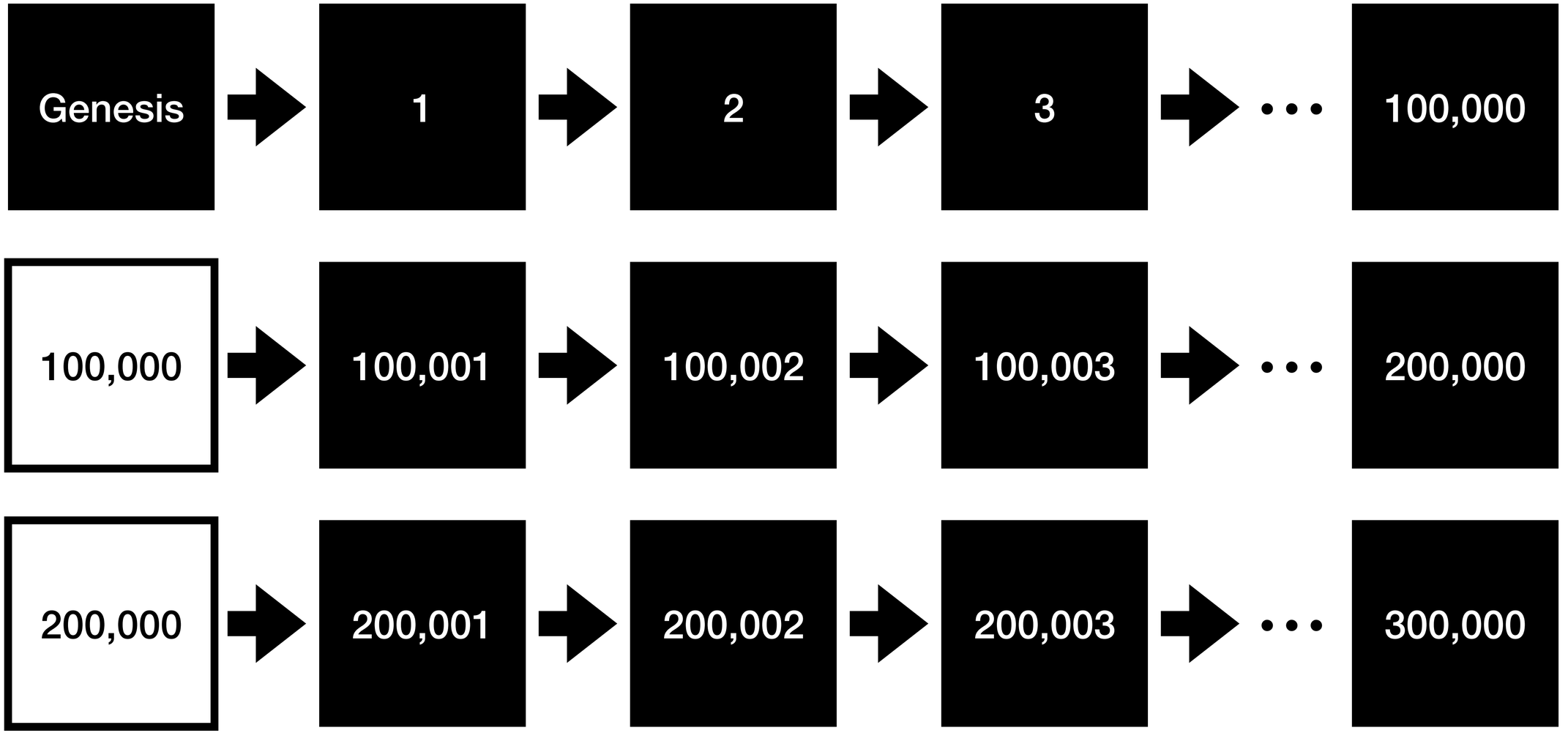
Out of Order Validation



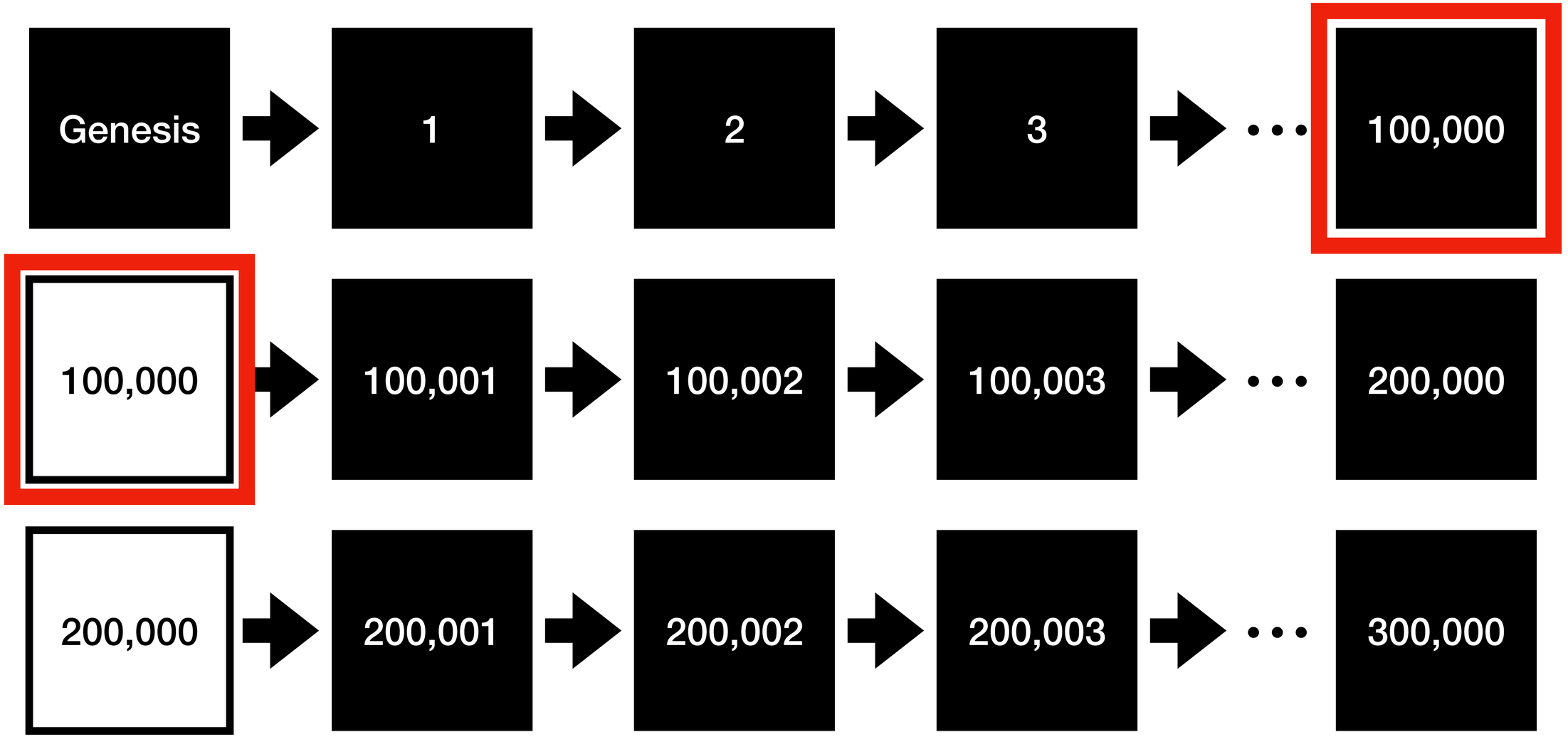
Out of Order Validation



Out of Order Validation

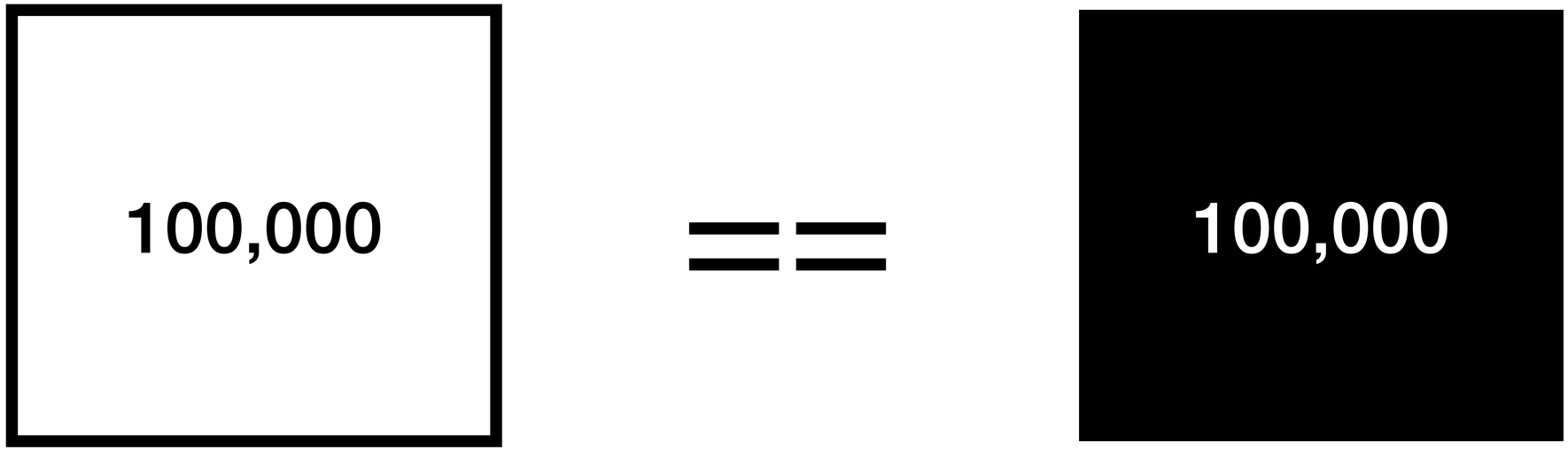


Out of Order Validation



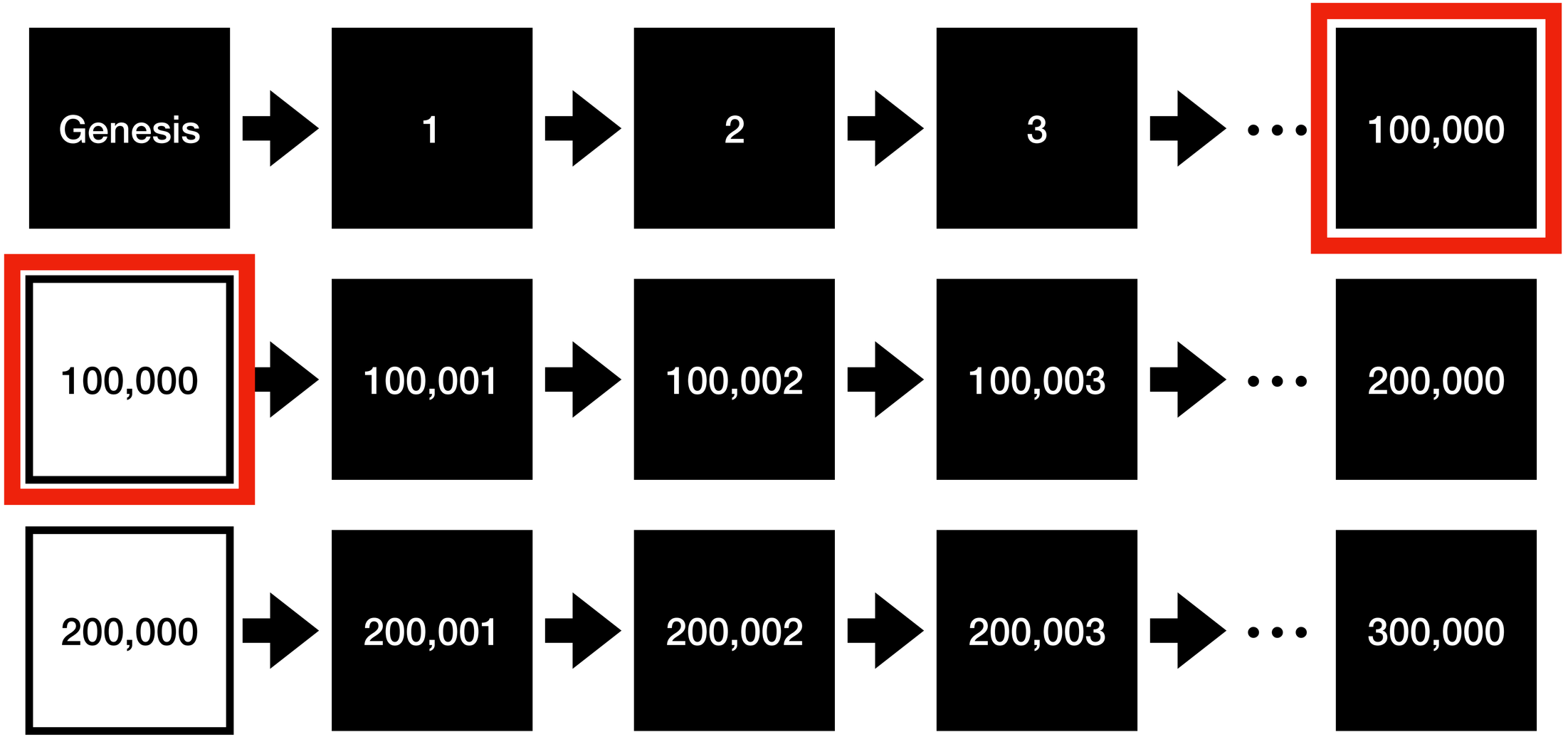
Compare Merkle Roots

Check if the generated roots at 100,000 is the same



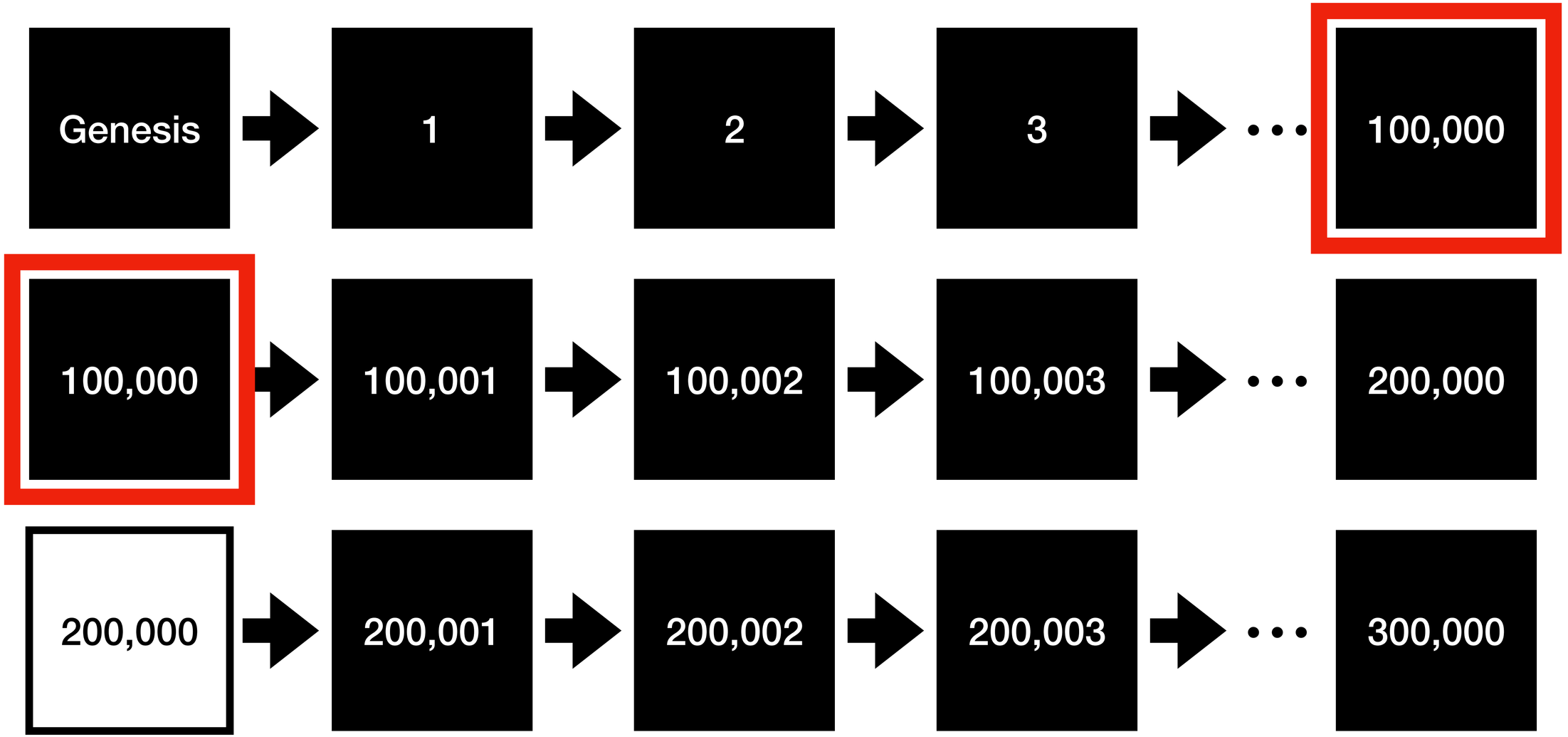
With Utreexo

Efficient parallel validation

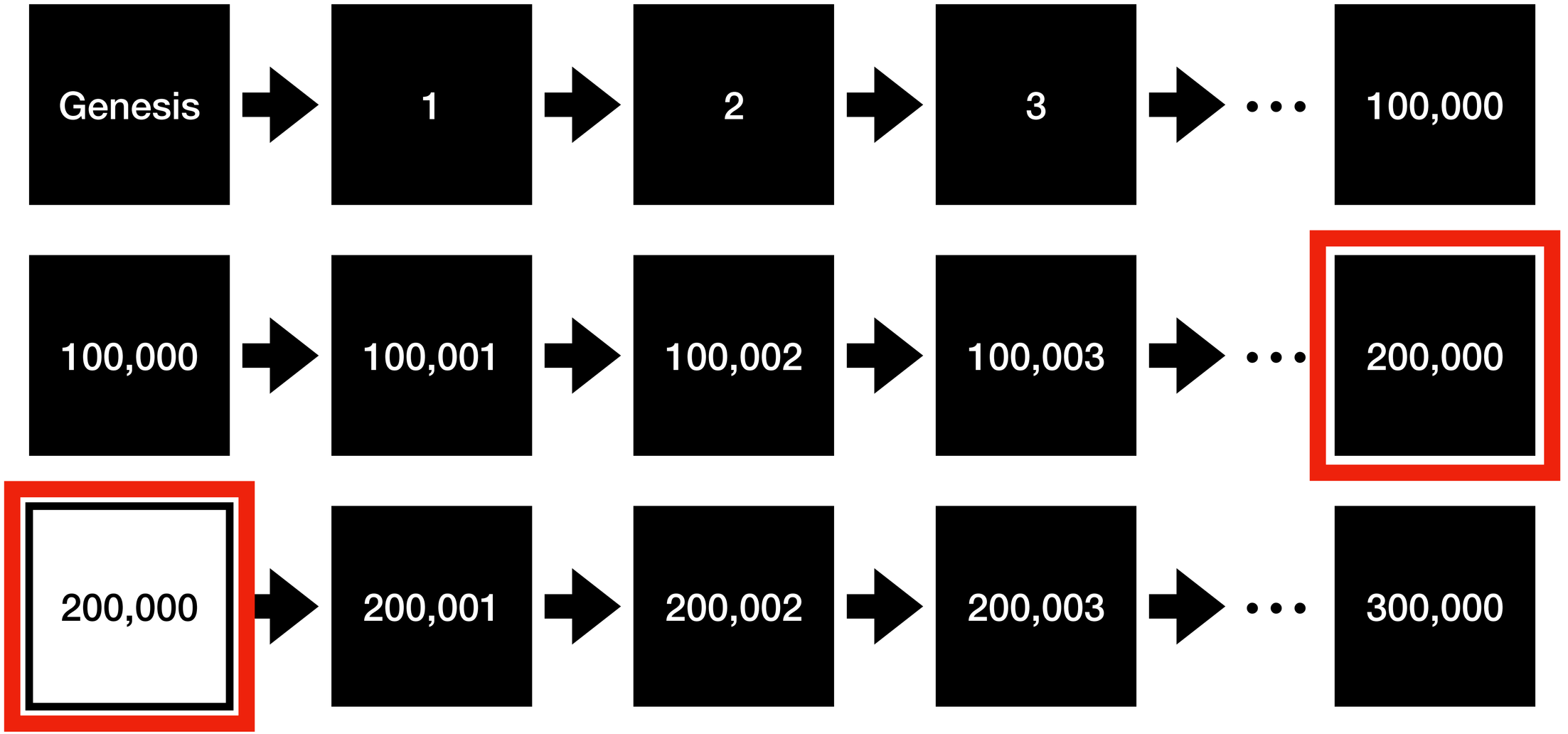


With Utreexo

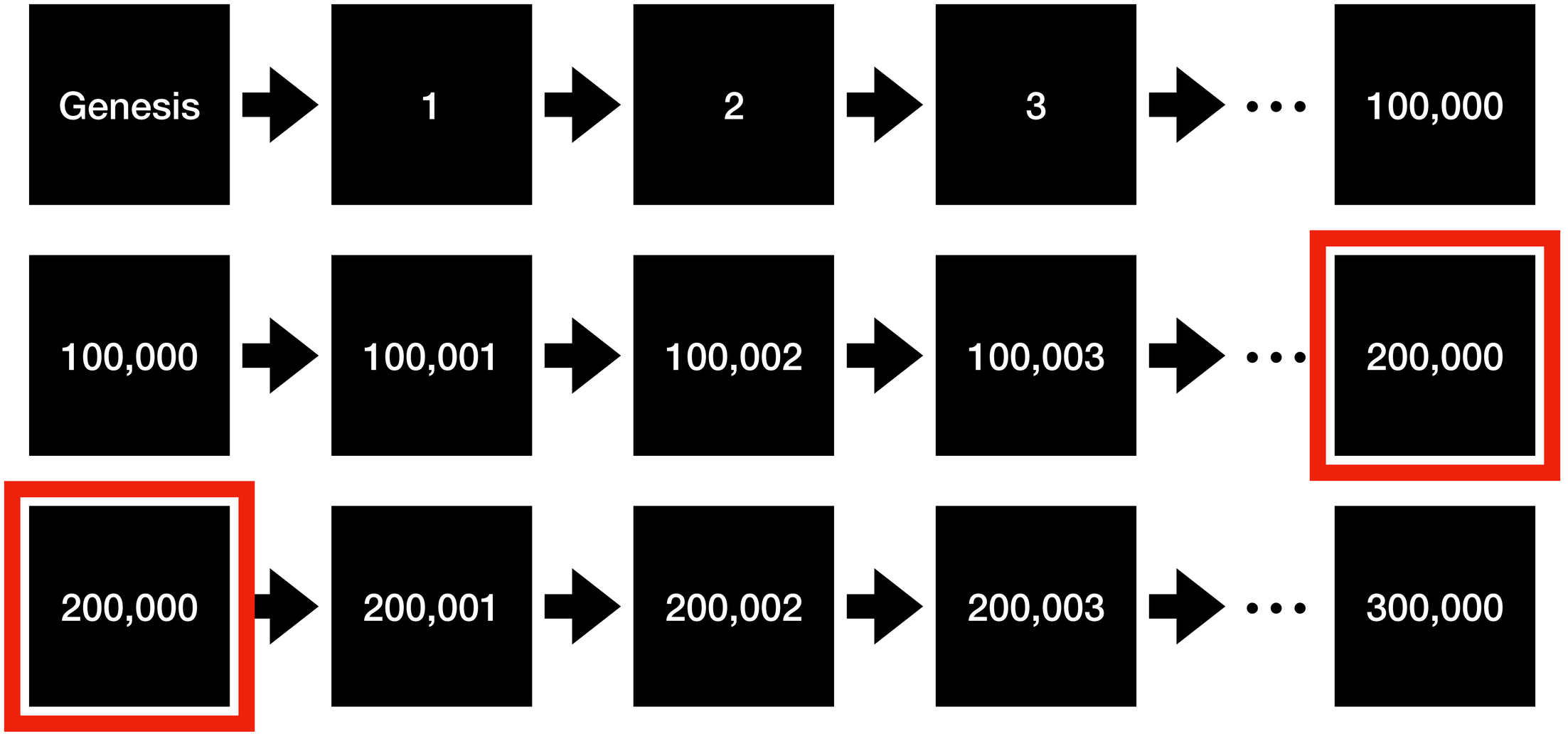
Efficient parallel validation



Out of Order Validation

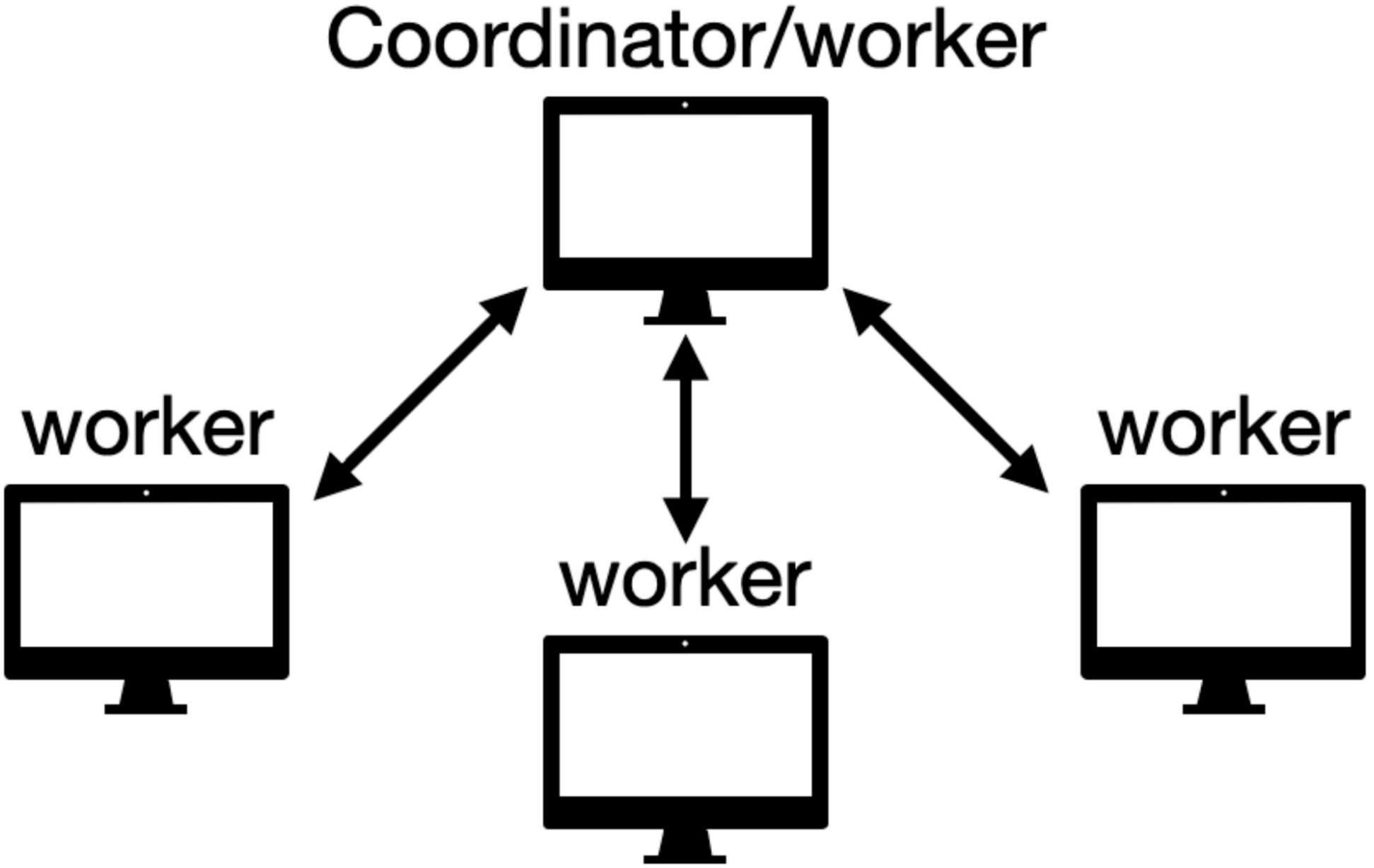


Out of Order Validation



~62% faster block validation

https://blog.bitmex.com/faster-blockchain-validation-with-utreexo-accumulators/

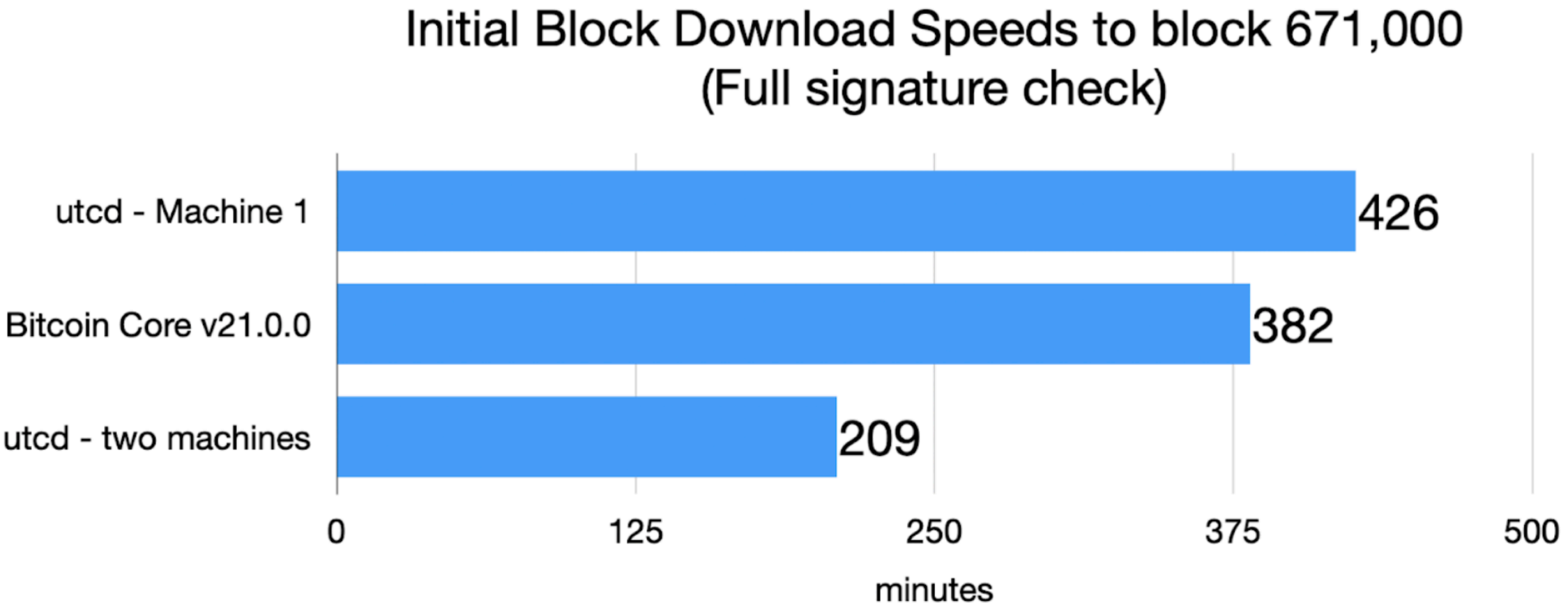


https://blog.bitmex.com/faster-blockchain-validation-with-utreexo-accumulators/

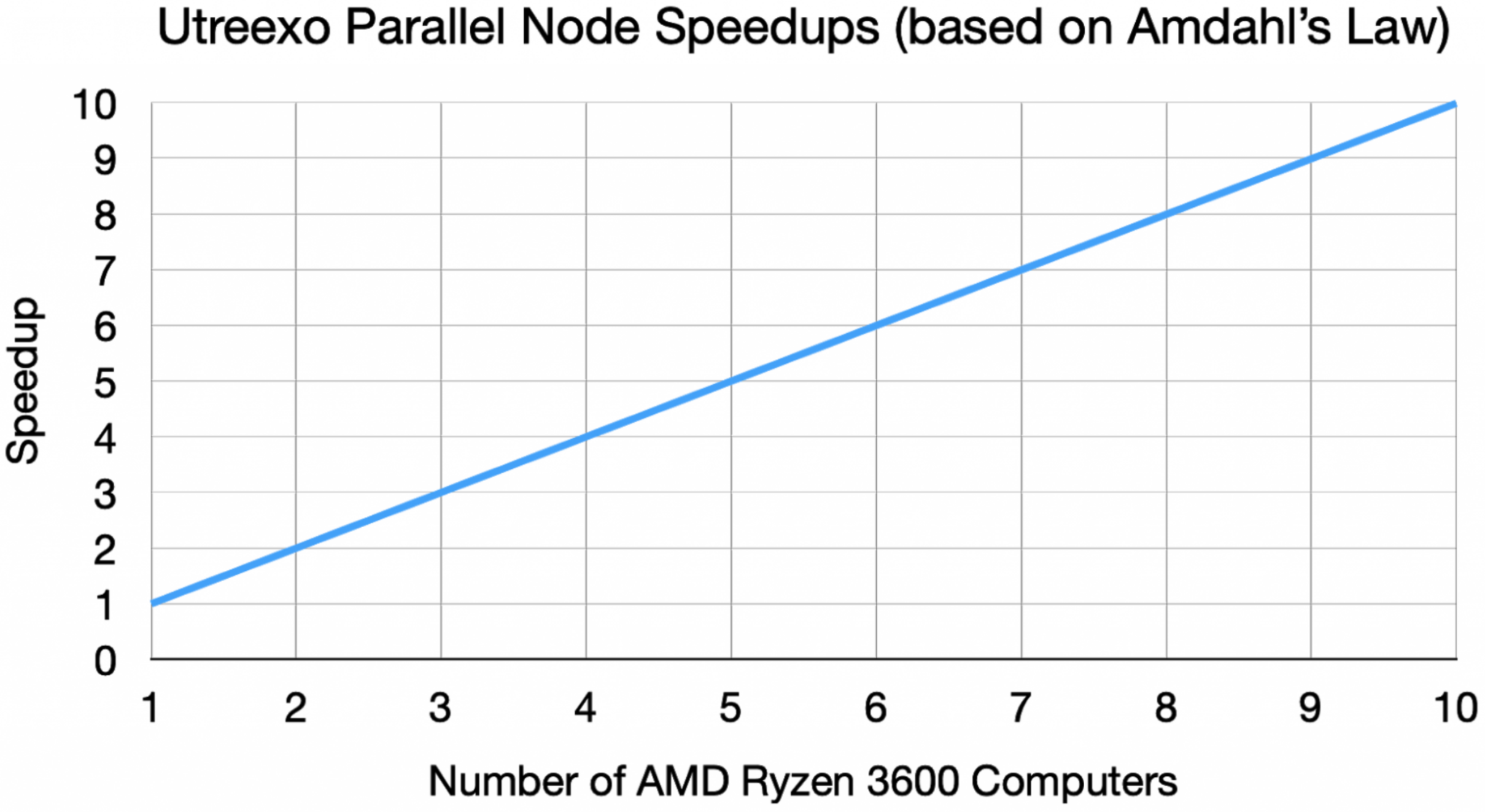
2 times faster block validation

When tested with 2 computers

https://blog.bitmex.com/faster-blockchain-validation-with-utreexo-accumulators/



https://blog.bitmex.com/faster-blockchain-validation-with-utreexo-accumulators/


Utreexo Parallel Node Speedups (based on Amdahl's Law)
Speedup
0
1
2
3
4
5
6
7
8
9
10
1
2
3
4
5
6
7
8
9
10
Number of AMD Ryzen 3600 Computers

Trade offs

More internet usage

Extra data needed to be downloaded for Utreexo nodes

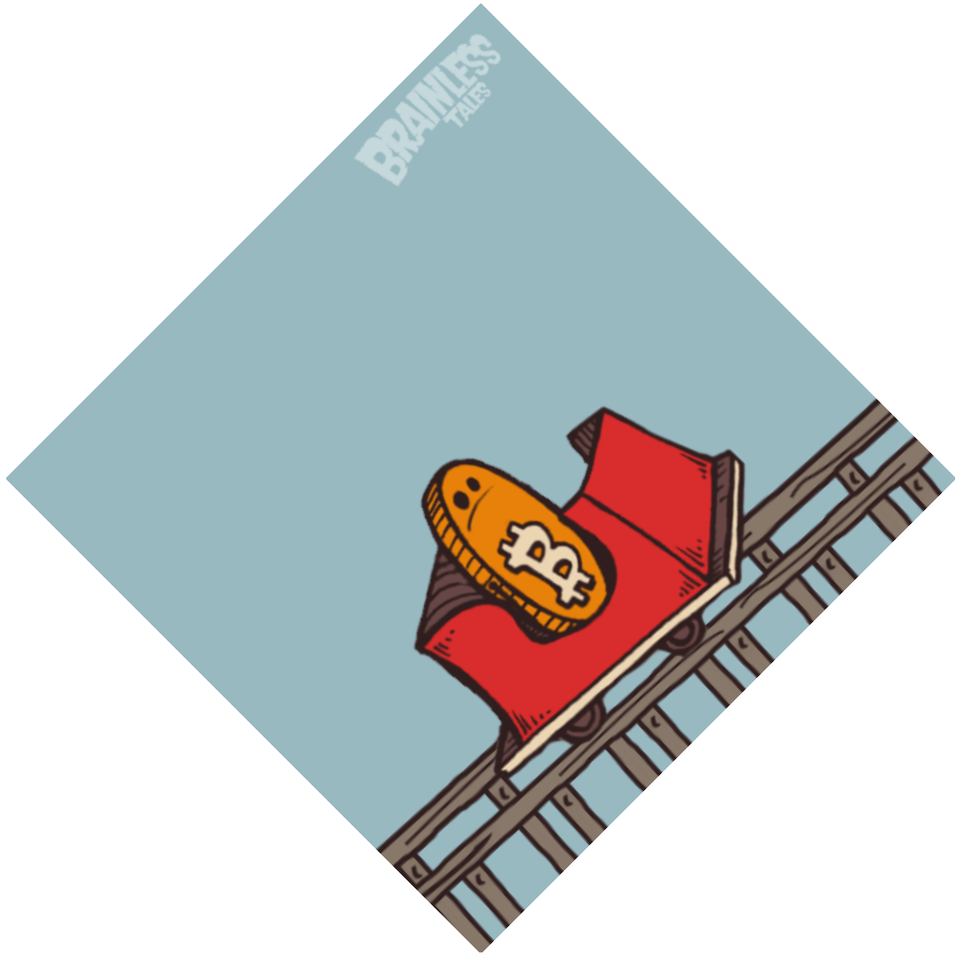
- Need to download 364GB Utreexo proofs (up until height 710,000)

More internet usage

Extra data needed to be downloaded for Utreexo nodes

- Need to download 364GB Utreexo proofs (up until height 710,000)
- Working on caching to reduce the amount

Internet
CPU



Disk read/write
Memory Usage