Mining
Proof of Work

How to mine

1. Join the P2P network, listen for transactions, validate all incoming transactions

2. Listen for new blocks, build on valid blocks

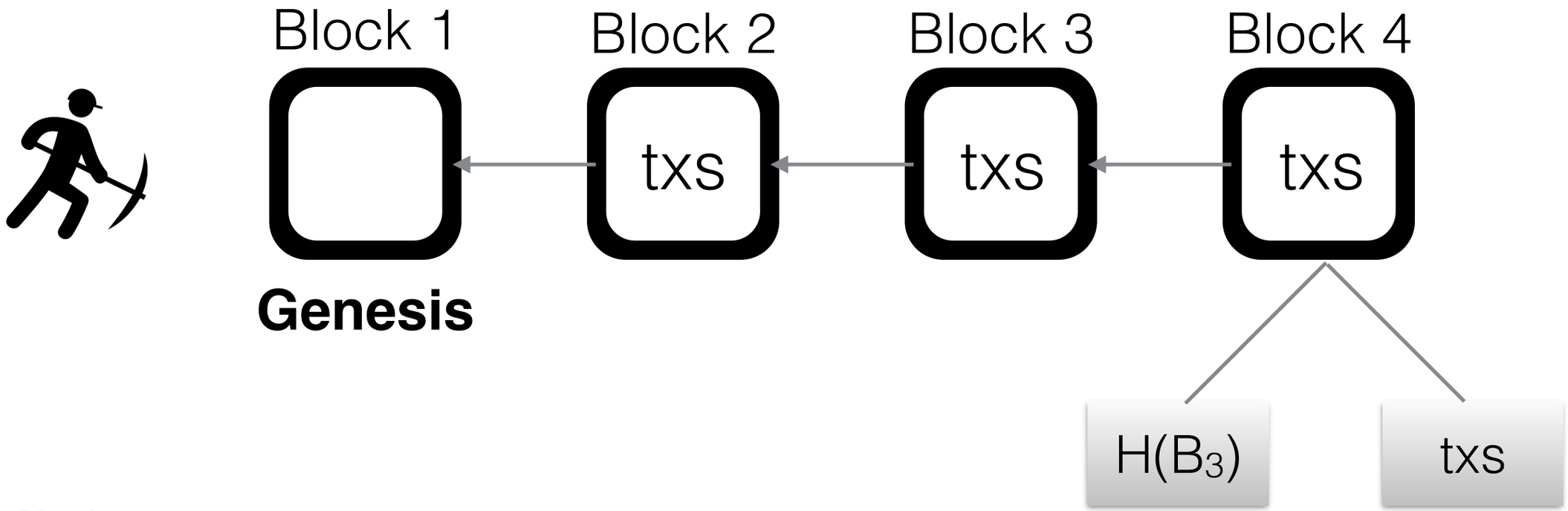
3. Construct a block template

4. Find a nonce that validates the new block

5. Tell all the other miners

6. Receive reward

Blockchain



Mining

- Find Nonce N, s.t.

$$\text{Hash}(\text{Hash}(B_3)|txs|N) < \text{target}$$

Best known approach: **Brute Force**

Controls the **difficult**

Mining Difficulty

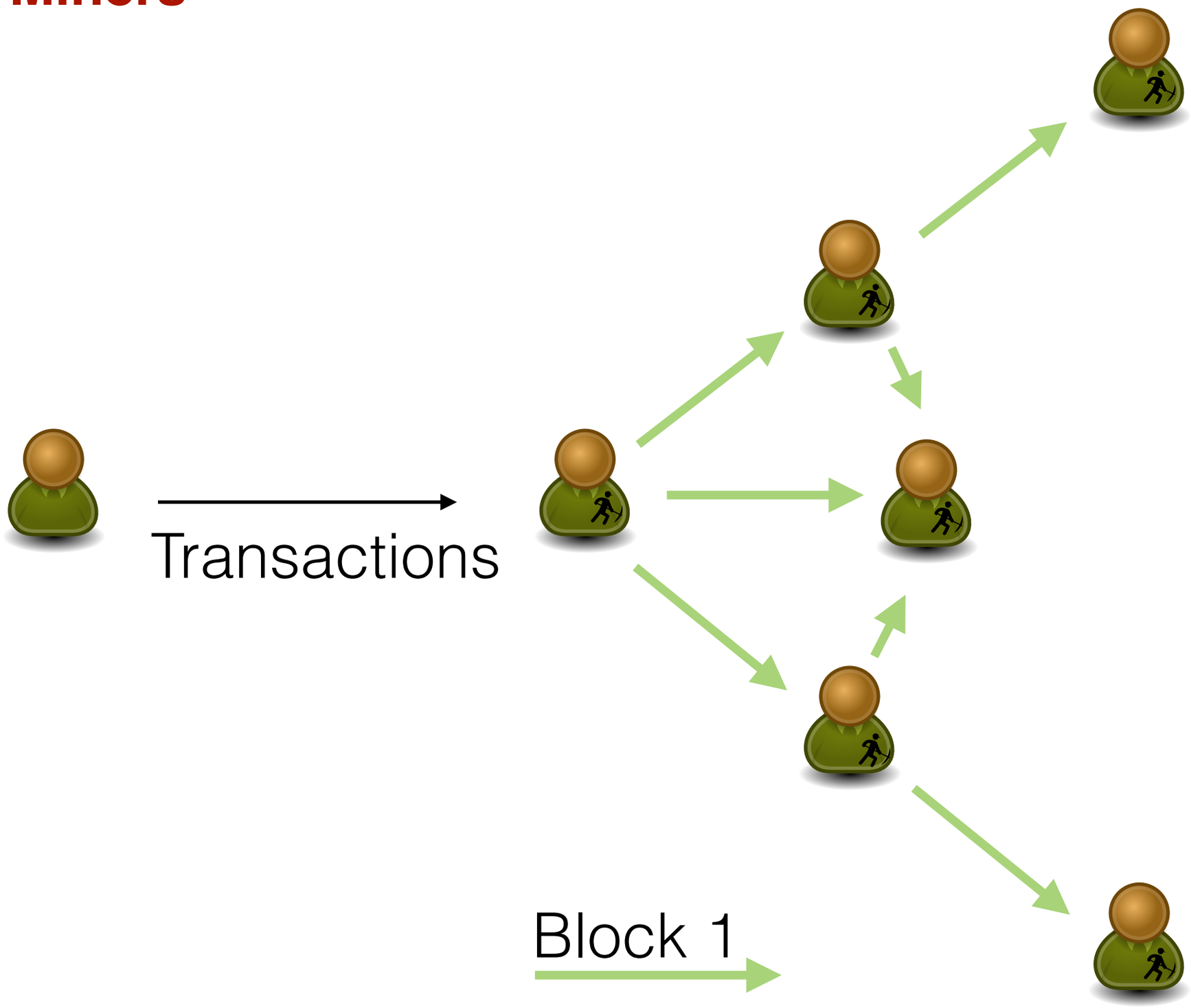
$$Hash(Hash(B_3)|txs|N) < target = 0x000**$$

~~Hash(Block_3 | merkle_root | 0xbeed) = 0x03ef..~~

~~Hash(Block_3 | merkle_root | 0xbeee) = 0x12ef..~~

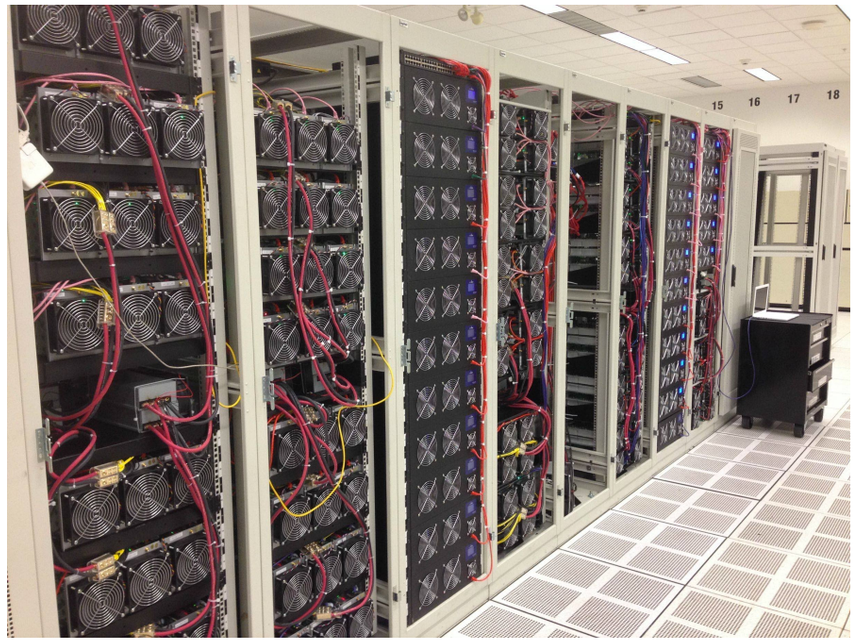
Hash(Block_3 | merkle_root | 0xbeef) = 0x000f..

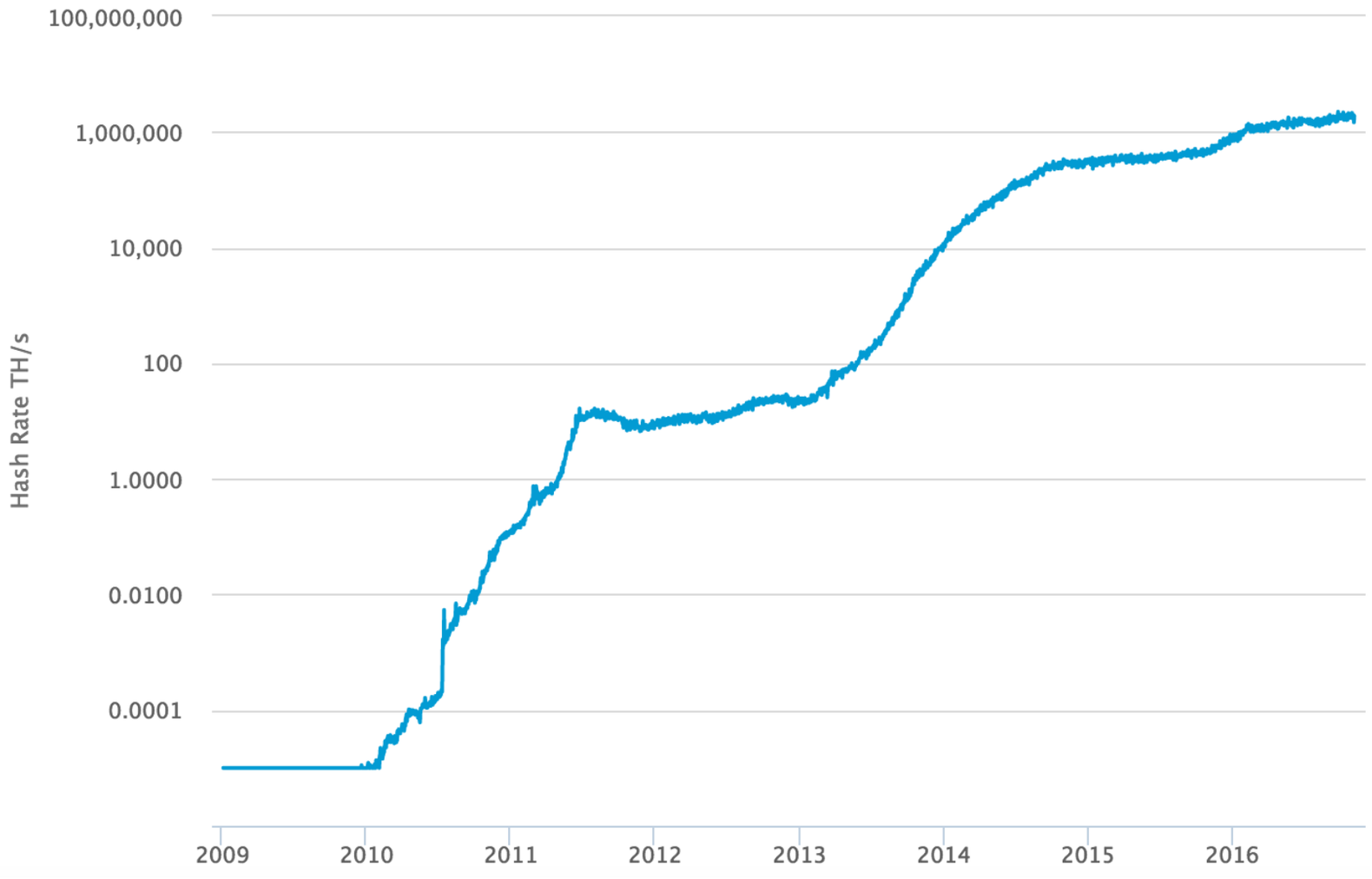
Miners

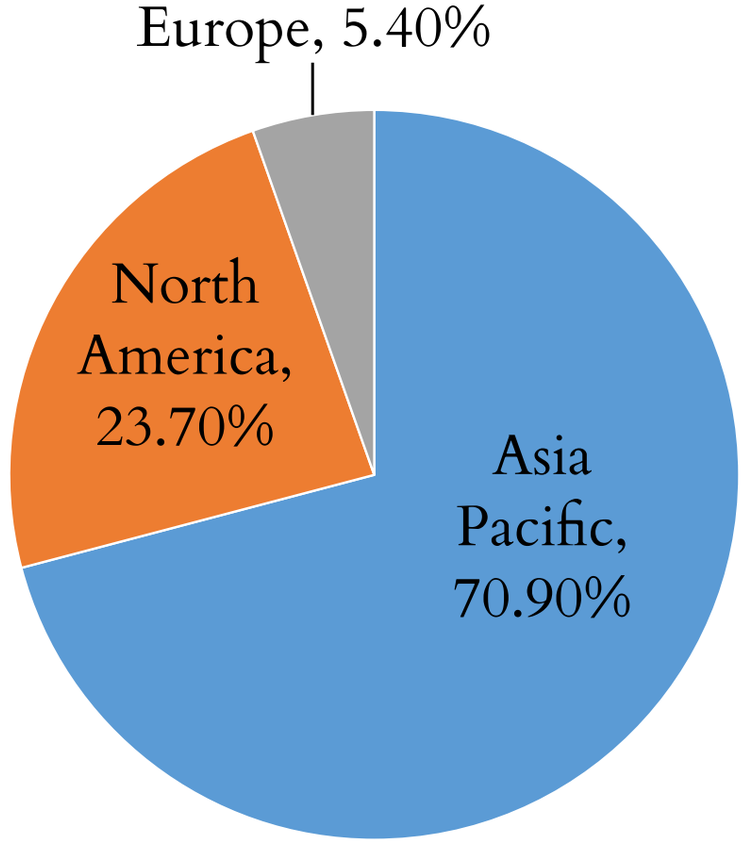


Mining

- From CPU to GPU to ASICs

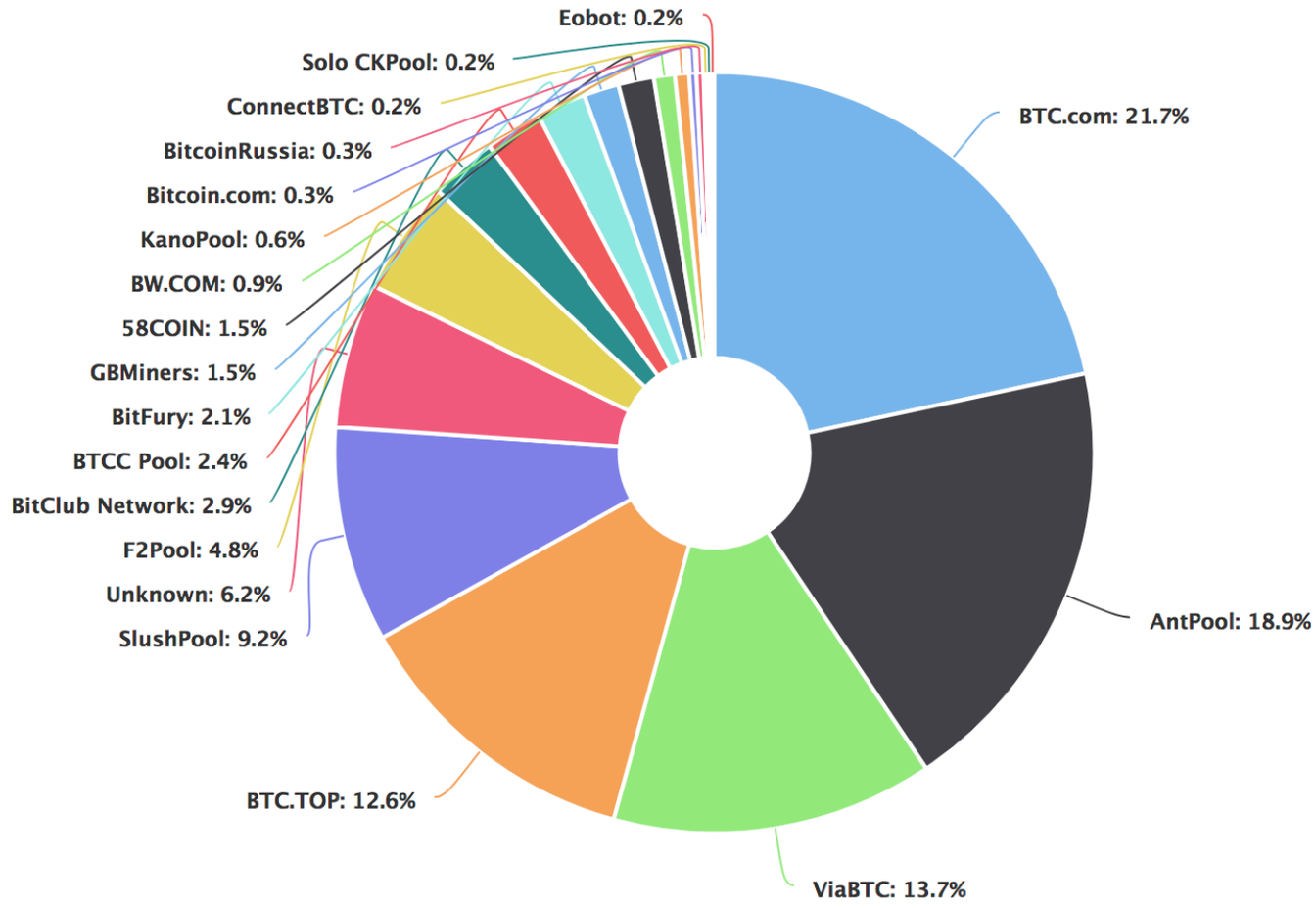






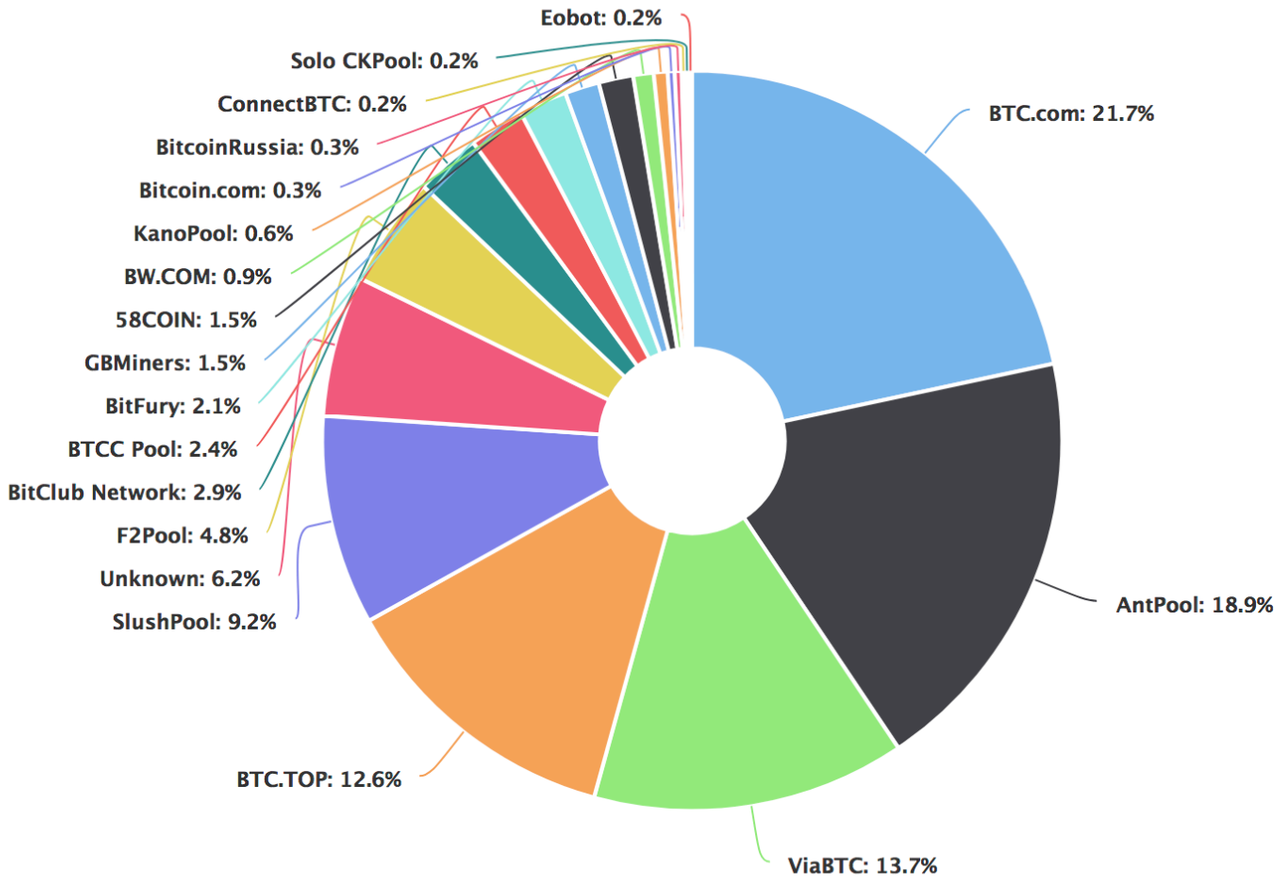
Mining Pools

- Probability of finding a block alone is very small
- Unite in Mining pools
- Payout is done proportional to the work

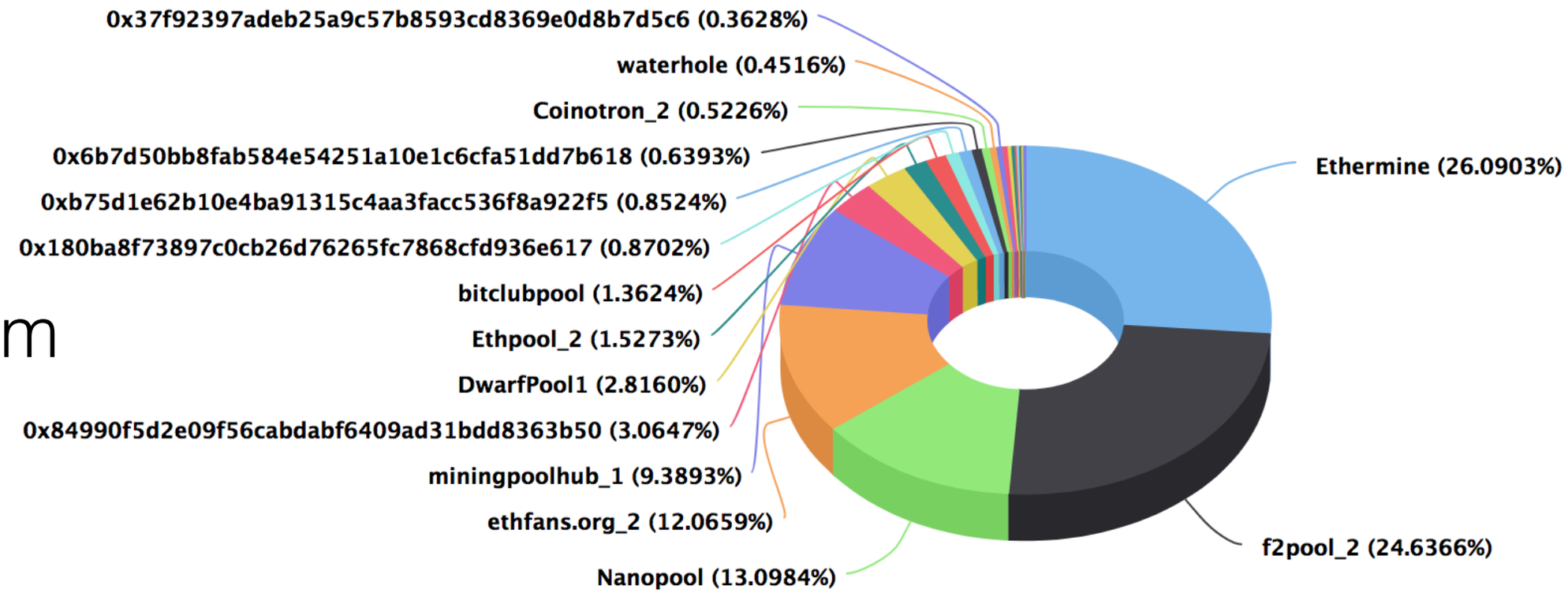


Mining Pools

Bitcoin



Ethereum



Mining Pools

- Probability of finding a block alone is very small

AntMiner S7

Advertised Capacity:
4.73 Th/s

Power Efficiency:
0.25 W/Gh

Weight:
8.8 pounds

Guide:
Yes

Price:
$479.95

Buy from amazon.com

Bitcoin Mining Calculator and Profitability Calcul

Bitcoin Mining Calculator is used to calculate mining profitability for Bitcoin mining. Enter your Bitcoin mining hardw electricity - dollars per kilowatt hour ($/kWh). The current Bitcoin difficulty, Bitcoin block reward, and Bitcoin price w

Hash Rate (GH/s): 4730.00	**Power (Watts):** 2600.00
Power Cost ($/kWh): 0.10	**Pool Fees %:** 0.00
Bitcoin Difficulty: 2227847638503.6300(	**Block Reward:** 12.50000000
Bitcoin to Dollar (USD): 13000.03000000	**Hardware Costs (USD):** 0.00

Calculate

Bitcoin Mining Calculator Summary

Days to generate one block mining solo: **23413.72 Day(s)** (can vary greatly depending on your luck)
Days to generate one BTC: **1873.10 Day(s)** (can vary greatly depending on the current exchange rates)
Days to break even: **N/A** (can vary greatly depending on the current exchange rates)

Miners

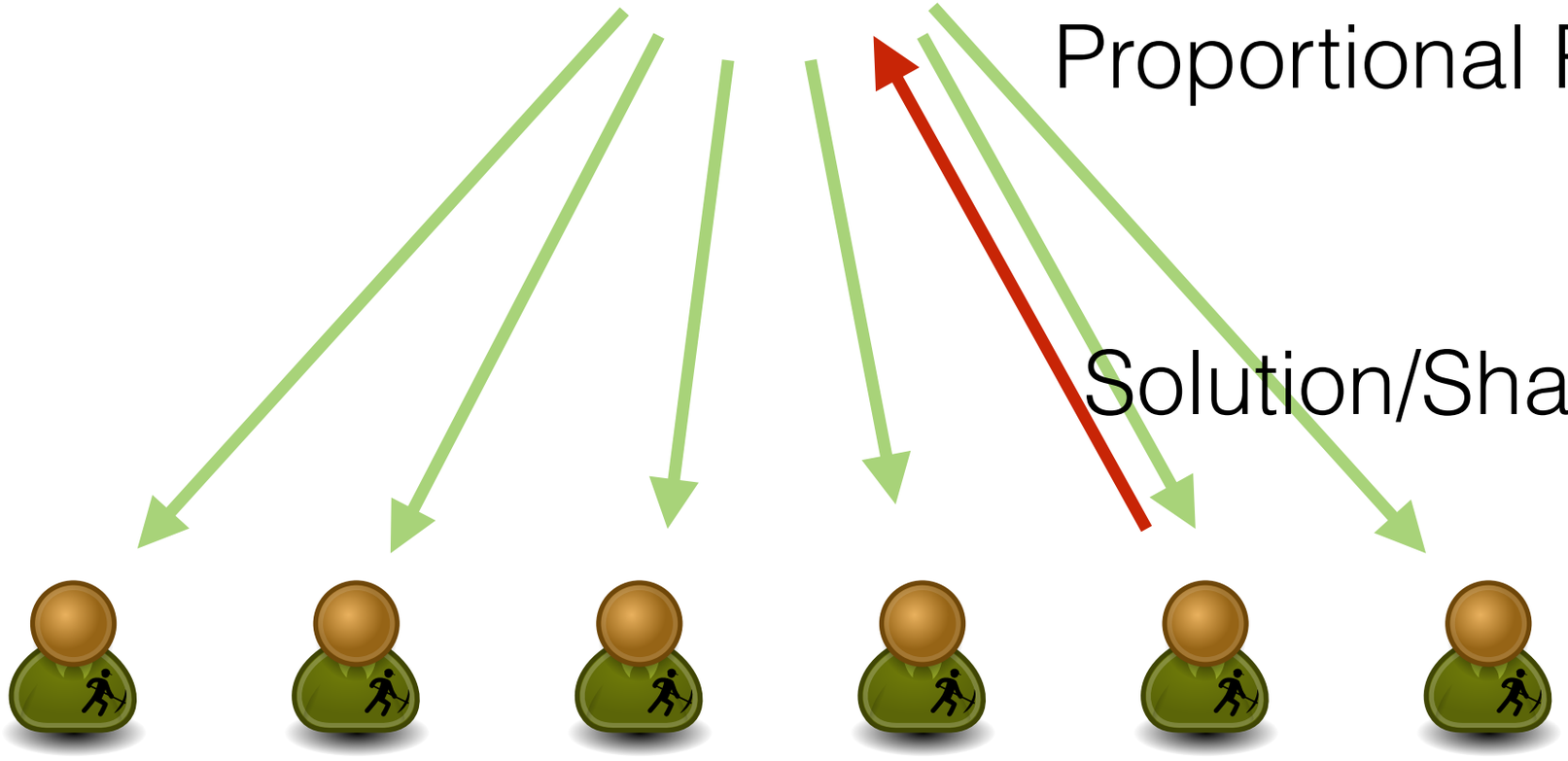
Solo Miner

Solo Miner

Pool Operator

Proportional Reward

Solution/Share



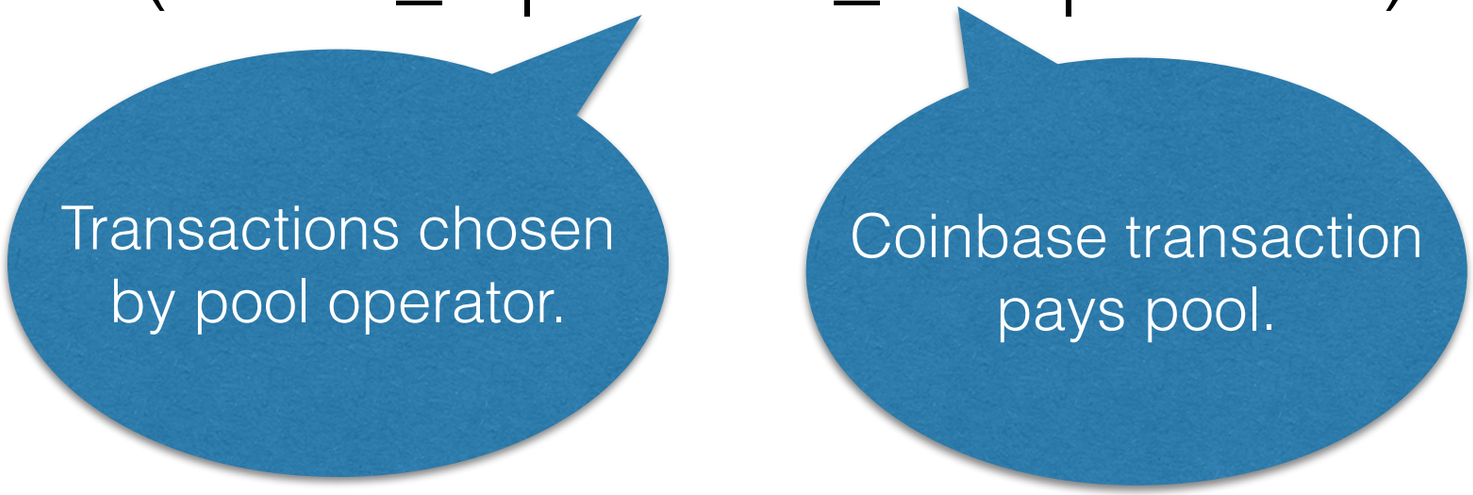
Pool Members

Mining Difficulty and Target in a Pool

Bitcoin's difficulty: $\text{Hash}(\text{Hash}(B_3)|txs|N) < T = 0x000^{**}$

Pool's difficulty: $\text{Hash}(\text{Hash}(B_3)|txs|N) < t = 0x00^{**}$

Hash(Block_3 | merkle_root | 0xbeef) = 0x000f..



- Difficulty of t << T

Merged Mining

Merged Mining

- Mining typically exclusive
- Each attempt aimed at a Bitcoin block
- How to start a new blockchain with the same Proof of Work algorithm?

Sha256
based PoW

20’000’000 Tera Hash

BoringCoin

Sha256
based PoW

20 Tera Hash

Merged Mining

- Mine a Bitcoin and an Altcoin block at the same time!
- Bitcoin coinbase transaction has no input (scriptSig).

Bitcoin

Bitcoin tx

Hash(prev | merkle_root | N) < target

Altcoin

Altcoin tx

Hash(prev_alt | merkle_root | N) < target

Merged Mining

Hash(prev | merkle_root | N) < target

tx[0] Coinbase
scriptSig: **alt header** → Hash(prev_alt | merkle_root | N) < target
scriptPubKey: …

Altcoin tx

tx[1]
tx[2]
…

Ignored by Bitcoin

Merged Mining

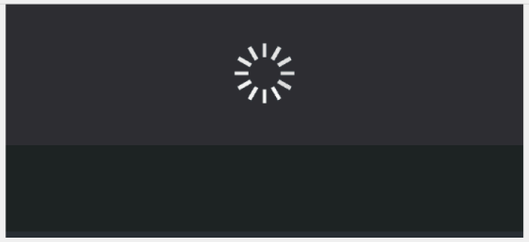
- **Advantages**
  - Easier bootstrapping of mining power
- **Disadvantages**
  - Cheaper for attackers (cf. CoiledCoins)
  - Inconsistency because miners might not validate transactions.
- Namecoin
  - One mining pool owned 60-70% hashrate

Mining Pool Dangers

Mining Pools

The Bitcoin Mining Arms Race: ×

Secure | https://www.coindesk.com/bitcoin-mining-detente-ghash-io-51-issue/



Mining Pools Danger

- Influence which transactions enter the blockchain
- Inflate transaction fees/Ethereum gas price
- Collude with each other (to reach 51%, selfish mining)

F2Pool Allegedly Prevented Users From Investing in Status ICO

https://themerkle.com/f2pool-allegedly-prevented-users-from-investing-in-status-ico/

Mining Pools Danger

- Network layer attacks
- Political and Regulatory danger
- Pool operators are trusted to pay out reward
- Influence voting mechanisms (cf. SegWit)

https://coinshares.com/assets/resources/Research/bitcoin-mining-network-december-2019.pdf

Decentralised Mining

P2Pool

- P2Pool builds a *separate* PoW chain: **share chain**
- Each block of the share chain is a share
- Each share chain block has a lower difficulty than the bitcoin block chain
- Once a share satisfies the same difficulty as Bitcoin, it's a valid Bitcoin block
- Coinbase transaction is used for reward and tracking of share chain.

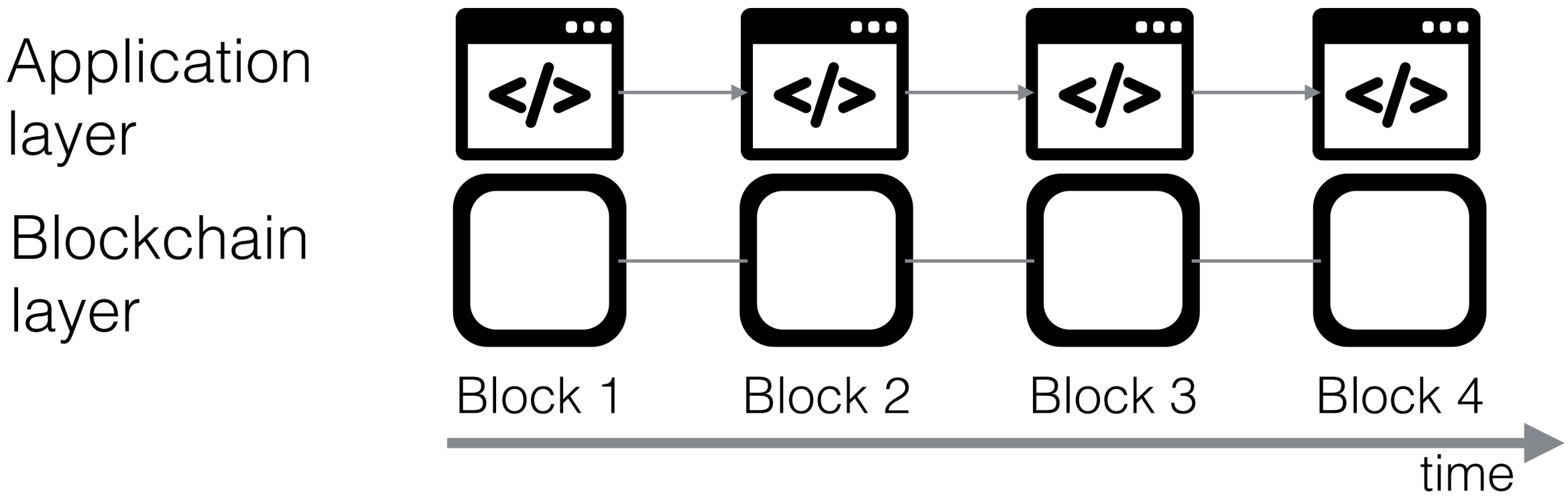
P2Pool Problems

- Efficiency
  - A share is set to be found every 30 seconds
  - The more miners, the higher payout variance
- Security
  - Many orphan shares (due to short block time)
  - Small P2Pools have weak security
  - Need to incentivize block submission and validation
- Last block found 2 years ago.. http://p2pool.info/

Smart Contracts

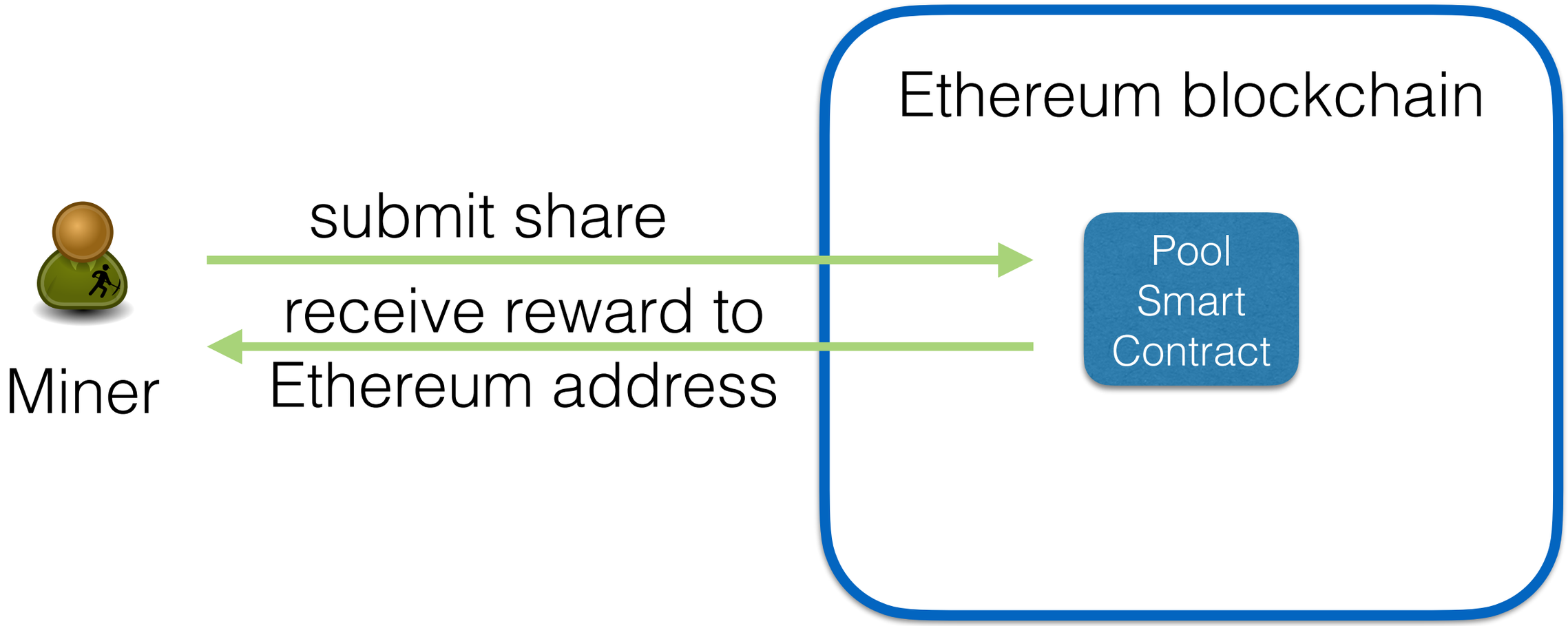
- Smart Contracts are similar to Bitcoin Script, just more powerful (next lecture is dedicated on Ethereum smart contracts).

- Code and data storage



SmartPool

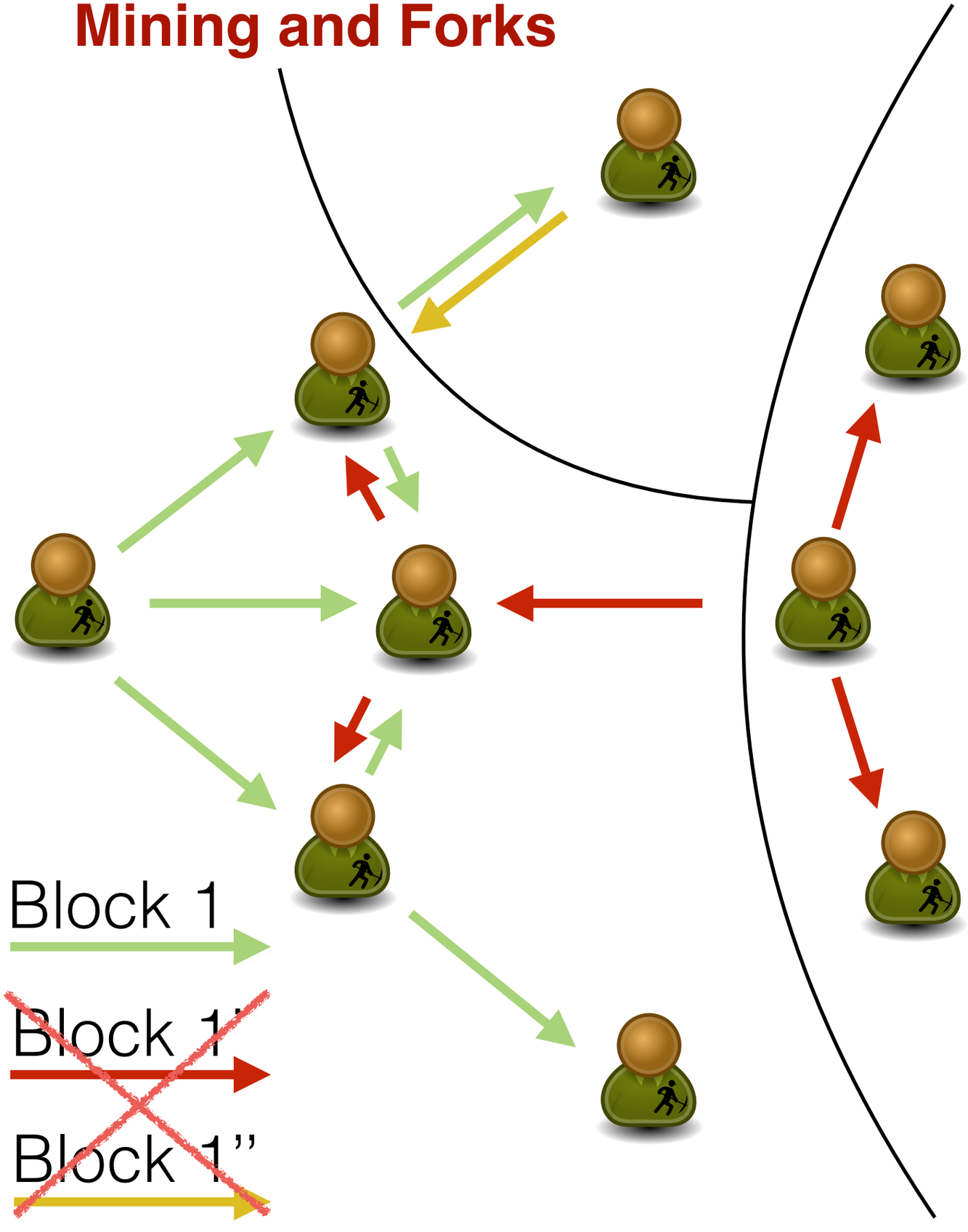
- SmartPool leverages Ethereum Smart Contracts
- Many miners —> many shares
- Smart Pool probabilistically samples submitted shares



Forks 🍴


© Katerina Kamprani - The Uncomfortable

Mining and Forks



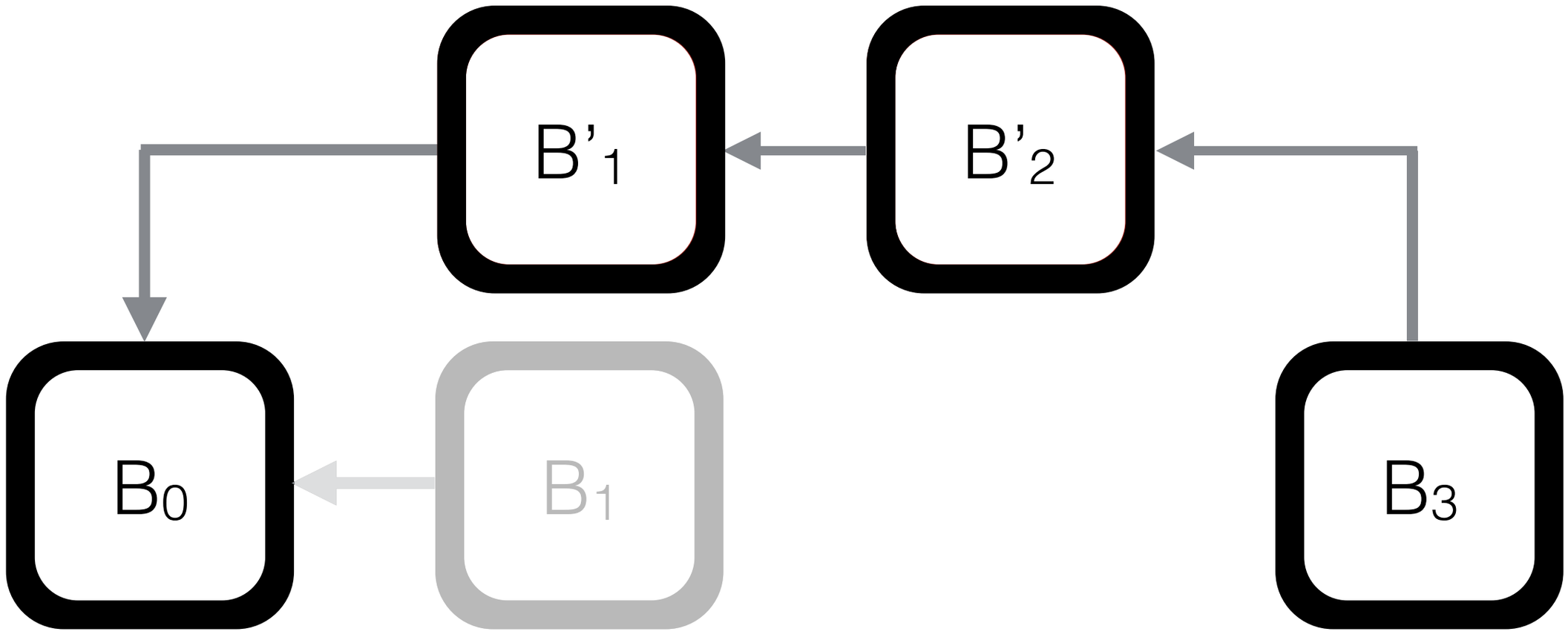
- Network partition
- **Stale blocks = lost efforts**

Selfish Mining

Denial of Service

Double Spending

Blockchain Forks



Types of Fork

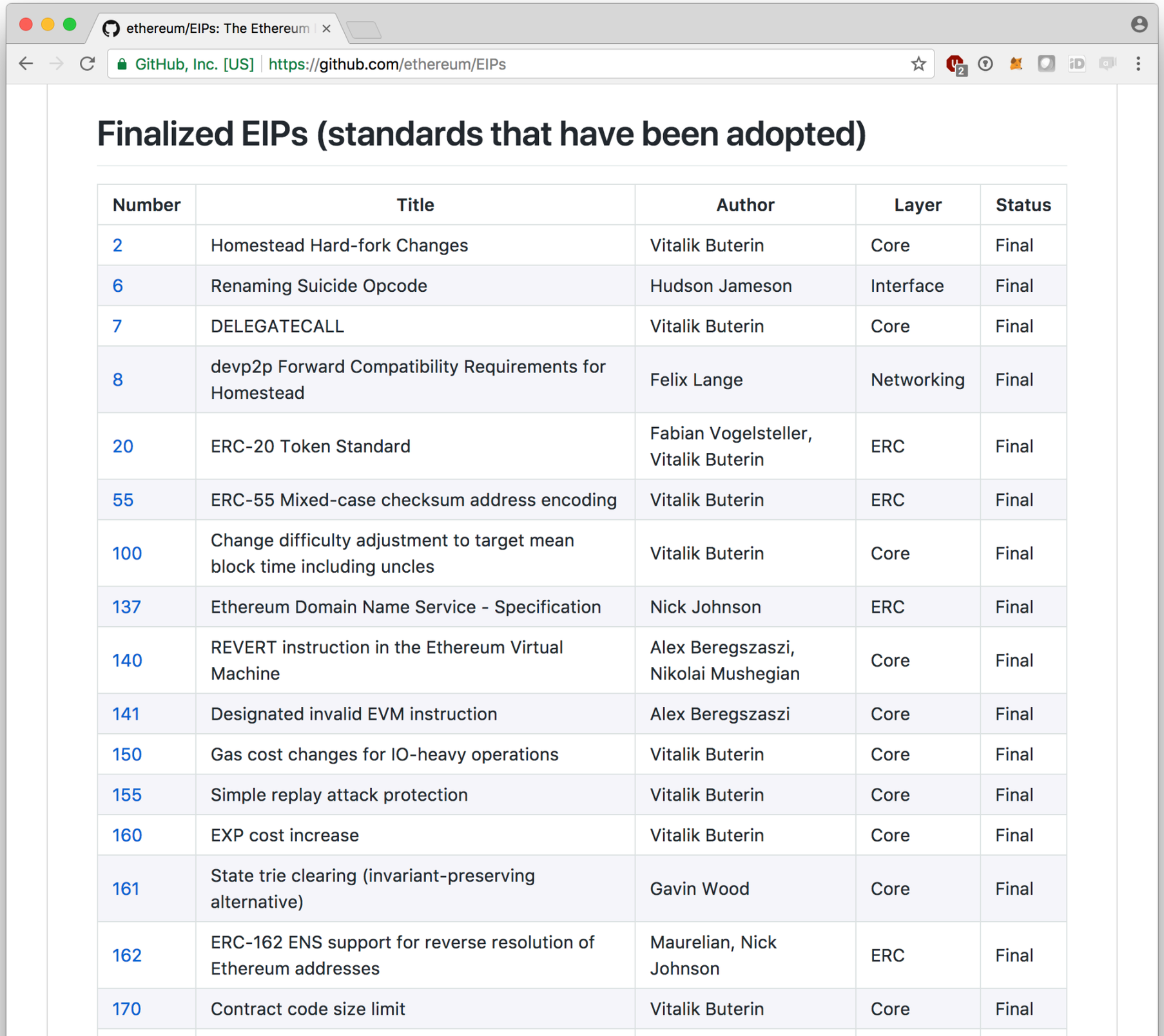
- P —> P’
  Protocol is updated from P to P’

- V —> V’
  Validity set is changed from old P to P'

- N
  The difference between V and V’

- Bitcoin Improvement Proposals (BIP)
  https://github.com/bitcoin/bips
- Ethereum Improvement Proposals (EIP)
  https://github.com/ethereum/EIPs

Ethereum Improvement Proposals

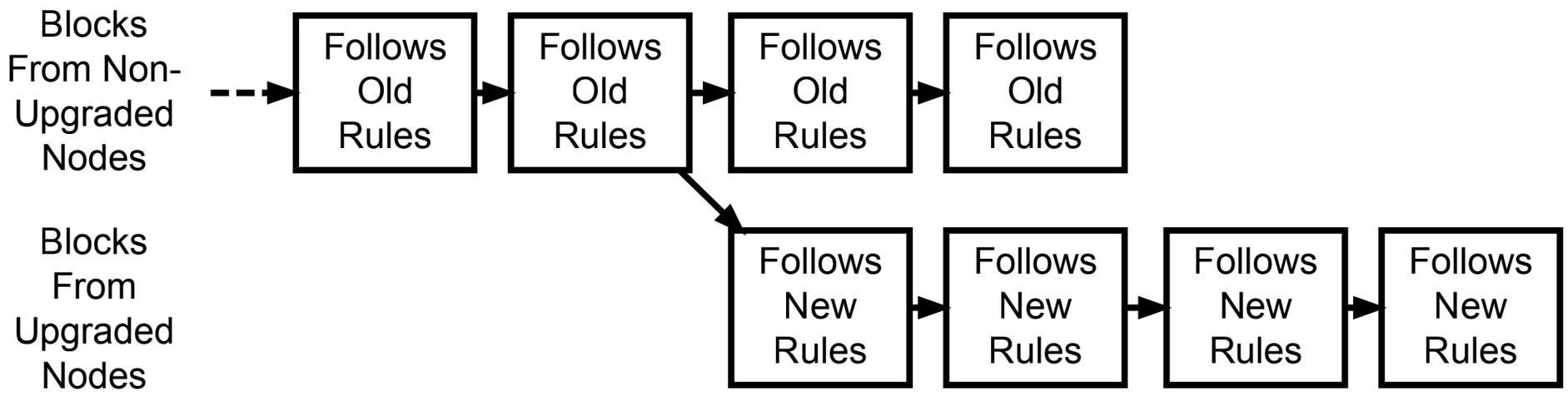


Finalized EIPs (standards that have been adopted)

Number	Title	Author	Layer	Status
2	Homestead Hard-fork Changes	Vitalik Buterin	Core	Final
6	Renaming Suicide Opcode	Hudson Jameson	Interface	Final
7	DELEGATECALL	Vitalik Buterin	Core	Final
8	devp2p Forward Compatibility Requirements for Homestead	Felix Lange	Networking	Final
20	ERC-20 Token Standard	Fabian Vogelsteller, Vitalik Buterin	ERC	Final
55	ERC-55 Mixed-case checksum address encoding	Vitalik Buterin	ERC	Final
100	Change difficulty adjustment to target mean block time including uncles	Vitalik Buterin	Core	Final
137	Ethereum Domain Name Service - Specification	Nick Johnson	ERC	Final
140	REVERT instruction in the Ethereum Virtual Machine	Alex Beregszaszi, Nikolai Mushegian	Core	Final
141	Designated invalid EVM instruction	Alex Beregszaszi	Core	Final
150	Gas cost changes for IO-heavy operations	Vitalik Buterin	Core	Final
155	Simple replay attack protection	Vitalik Buterin	Core	Final
160	EXP cost increase	Vitalik Buterin	Core	Final
161	State trie clearing (invariant-preserving alternative)	Gavin Wood	Core	Final
162	ERC-162 ENS support for reverse resolution of Ethereum addresses	Maurelian, Nick Johnson	ERC	Final
170	Contract code size limit	Vitalik Buterin	Core	Final
181	ERC-181 ENS support for reverse resolution of			

Hard Fork

- Makes previously invalid blocks/transactions valid (and vice-versa)
- All nodes need to upgrade to the latest version



A Hard Fork: Non-Upgraded Nodes Reject The New Rules, Diverging The Chain

https://bitcoin.org/en/developer-guide#consensus-rule-changes

Hard Fork Example

- Ethereum forked because a smart contract vulnerability was exploited.
- The fork “reversed” the hack.
- The blockchain was no longer append-only 😥

Soft Fork

- Only previously valid blocks/transactions are made invalid —> reducing functionality

- Backwards compatible
  - Old nodes accept new blocks as valid

- Miner-activated Soft Fork (MASF)
  - Majority of miners upgrade to enforce

- User-activated Soft Fork (UASF)
  - Full nodes coordinate to enforce rules without miners.

Soft Fork Examples

- Make transactions >1 kb invalid
- Pay-to-script Hash (P2SH)
  - Send transactions to a Script hash (3xxxx) instead of a public key hash (1xxxx)
  - Recipient must provide a Script matching the Script hash and input data that makes the script evaluate to *true*.

Types of Fork

Type	Validity Set		Incurred Fork		Examples
	New	Relation to Old	Soft	Permanent / Hard	
Expanding	$\mathcal{V}' = \mathcal{V} \cup \mathcal{N}$, $\exists n \in \mathcal{N} : n \notin \mathcal{V}$	$\mathcal{V}' \supset \mathcal{V}$	never	$\mathcal{V}'$ is majority	Blocksize increase, new opcode
Reducing	$\mathcal{V}' = \mathcal{V} \setminus \mathcal{N}$, $\mathcal{N} \subset \mathcal{V}$	$\mathcal{V}' \subset \mathcal{V}$	$\mathcal{V}'$ is majority	$\mathcal{V}$ is majority	Blocksize decrease, opcode removal, SegWit
Conflicting (Bilateral)	$\mathcal{V}' = (\mathcal{V} \cup \mathcal{N}) \setminus (\mathcal{V} \cap \mathcal{N}) = V \triangle N$	$(\mathcal{V}' \not\subseteq \mathcal{V})$, $(\mathcal{V} \not\subseteq \mathcal{V}')$, $V' \cap V \neq \emptyset$	never	always	Opcode redefinition, chain ID for replay protection
Conditionally Reducing (Velvet)	$\mathcal{V}' = \mathcal{V}$	$\mathcal{V}' = \mathcal{V}$	never	never	P2Pool, merged mining, colored coins

Segregated Witness

<Sig> <PubKey> OP_DUP OP_HASH160 <PubKeyHash> OP_EQUALVERIFY OP_CHECKSIG

- Signatures are malleable (can change while remaining valid)

- Separating transaction signature (65% of the data) from transaction data.

- Advantages
  - Increases the number of transactions that can be stored in a block.
  - Removes transaction malleability
- Disadvantages
  - Signatures required to validate a block
  - Complex

Voting through Blocks

- BIP written
- BIP discussed
- BIP implemented
- BIP voted by miners
  - Coinbase transaction contains data field
  - E.g. vote over the last 100 blocks
  - If majority (>55%) then miners implement

Proof of ..Stake ..Authority ?:

Various Leader Election Mechanisms

- Proof of Stake

Vote with your staked assets.

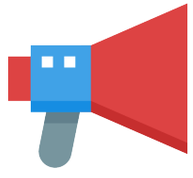
- Delegated PoS

Allow someone else to vote on your behalf.

- Proof of Authority

== centralised system

Idea of Proof of Stake

- Validators commit stake 
- Propose and attest blocks 
- Randomly selected as block proposer

- Once “enough” validators attest a block, it’s “valid”
- Block rewards paid for proposing and attestation
- Penalties for malicious behaviour:
  being offline, attempting a fork, etc.

Proof of Stake Myths

Maybe with sharding, but at the cost of a complexity increase.

- PoS increases scalability
- PoS reduces centralisation