Course Outline

1. Cryptocurrency and Block chain
2. Delving into BlockChain
3. Bitcoin and Block chain
4. Bitcoin Mining
5. Ethereum
6. Setting up private Blockchain Environment using Ethereum Platform
7. Hyperledger
8. Setting up Development Program using Hyperledger composer
9. Create or Deploy our private Blockchain on Multi chain
10. Prospect of Blockchain

Mining of bitcoin

Agenda

At the end of this session you will be able to:

- Understand Economics of Bitcoin
- Define Bitcoin Mining
- Describe Fabrication of a Block Header
- Define Mining
- Identify Successful Mining
- List Difficulties in Solo Mining
- Understand Mining: By pool of Miners

Economics of Bitcoin

Economics of Bitcoin

> Let us consider the economics of Bitcoin before we get into the details of Bitcoin Mining.

Economics of Bitcoin

1. Bitcoins are "stamped" after the creation of each block at a settled and reducing rate.

2. For each 210,000 blocks or roughly every four years the money issuance rate is diminished by half in bitcoin ecosystem

3. For the first four years of bitcoin operation of the system, each block contains 50 new bitcoins

4. The rate of new coins diminishes exponentially more than 64 "halvings" until block 13,230,000(mined roughly in year 2140), when it achieves the base cash unit of 1 satoshi

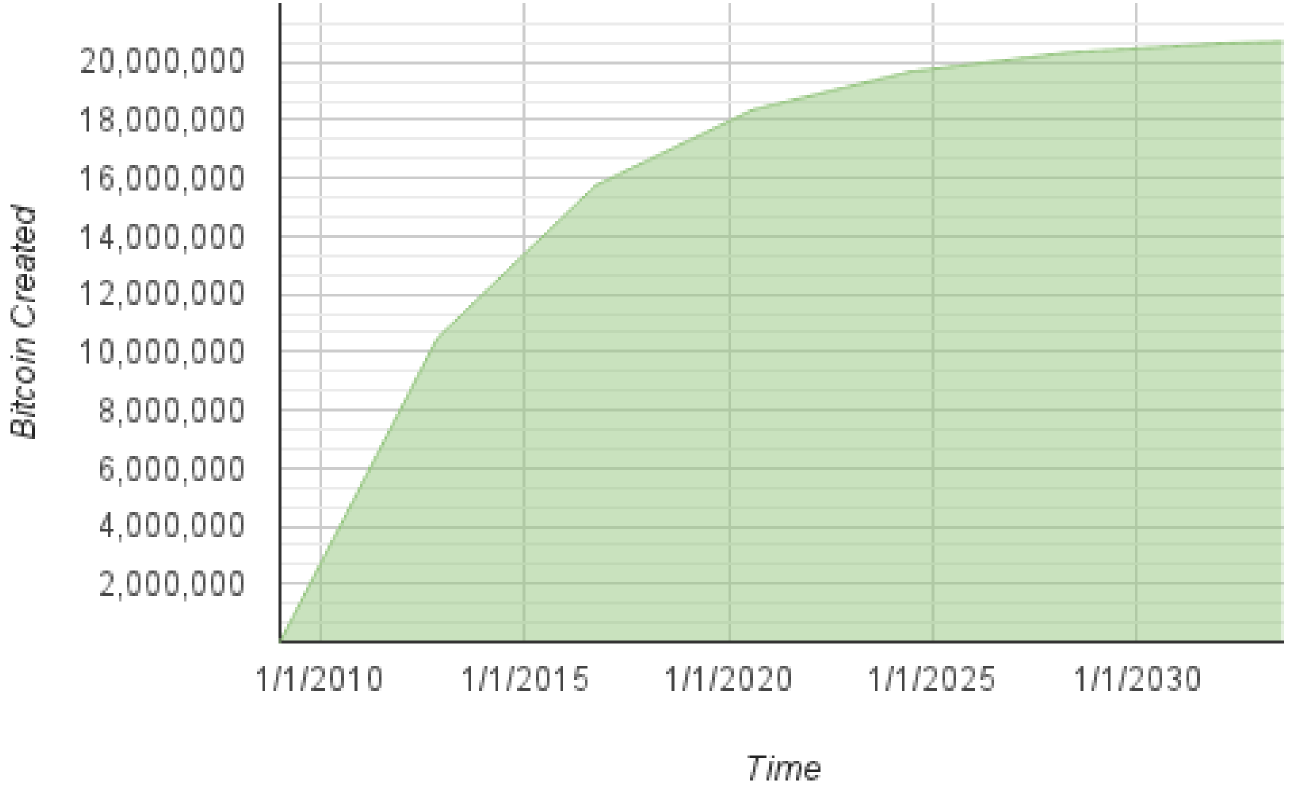
5. Finally, after near about 13.44 million blocks, in approximately 2140 almost 2,099,999,997,690,000 satoshis, or almost 21 million bitcoins will be issued

6. Thereafter, blocks will contain no new bitcoins and miners will be rewarded solely through the transaction fees

Bitcoin Economy: Supply of Bitcoin

The limited and decreasing supply and issuing rate makes a settled financial supply that opposes expansion

Different to a fiat money, which can be imprinted in unending numbers by a national bank, bitcoin can never be swelled by printing



Supply of bitcoin currency over time based on a geometrically decreasing issuance rate

Bitcoin Mining

Bitcoin Mining

So, how do we mine/create Bitcoins?

Bitcoin Mining-A Brief Note

> Bitcoin mining is process of adding transaction records to bitcoin's public ledger of past transactions or blockchain. This ledger of past transactions is called the block chains it is chain of blocks. The block chain serves to confirm transcations to the rest of th enetwork as having taken place.

Types of Bitcoin mining

Solo Mining

- Miners endeavours to produce new blocks all alone, with the returns from the reward and transaction expenses going altogether to himself
- Enabling miners to receive extensive payments with a higher variance (longer time between payments)

Pooled Mining

- Miner pools assets with different mineworkers to discover blocks all the more regularly, with the returns being shared among the pool miners in rough correlation to the measure of hashing power they each contributed
- Enables the miner to receive little payments with a lower change (shorter time between payments)

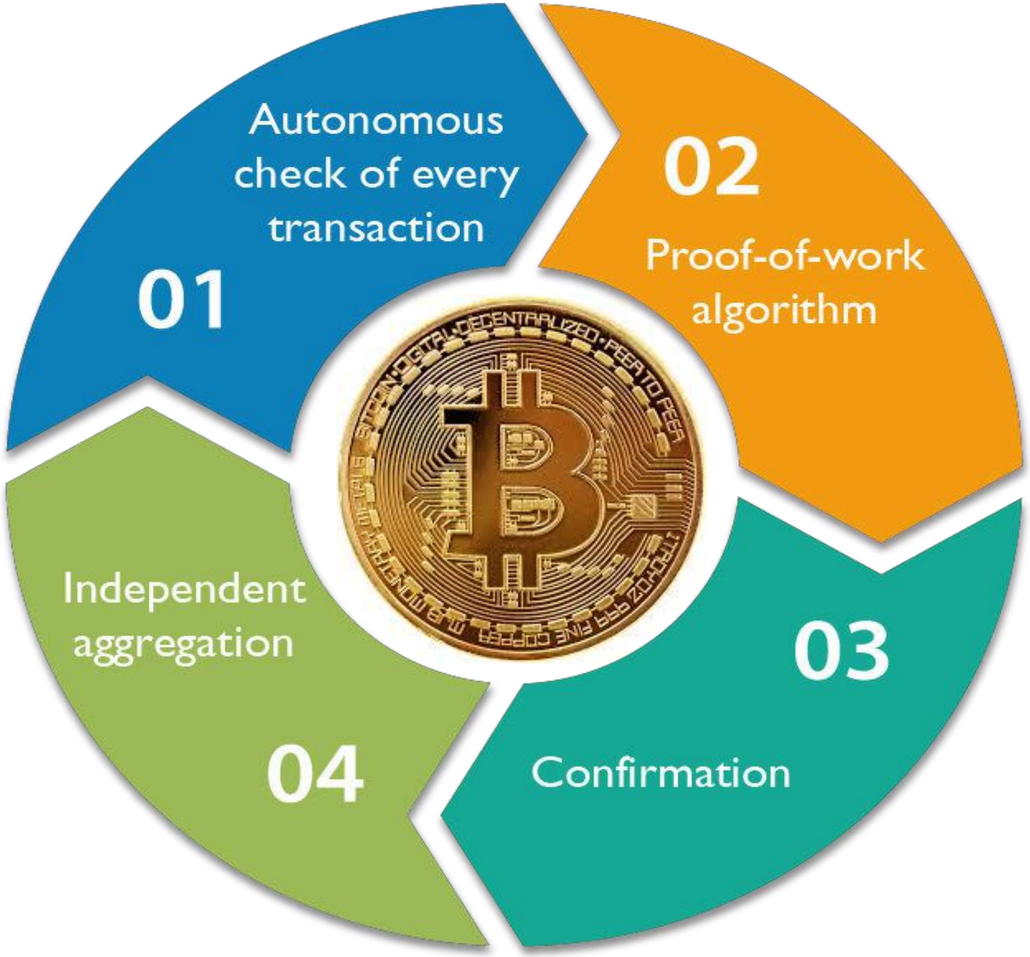
Bitcoin Mining

“Let’s first understand the working of solo miners”

Mining & Consensus

Bitcoin consensus algorithm allows for the creation of new Bitcoins by the process of Mining.

It involves four steps:



01 Autonomous check of every transaction

02 Proof-of-work algorithm

03 Confirmation

04 Independent aggregation

Autonomous Verification of Transactions

1

Autonomous check of every transaction, by each full node, in light of an extensive rundown of criteria

→

2

Independent aggregation of those transactions into new blocks by mining nodes combined with exhibited calculation through a proof-of-work algorithm

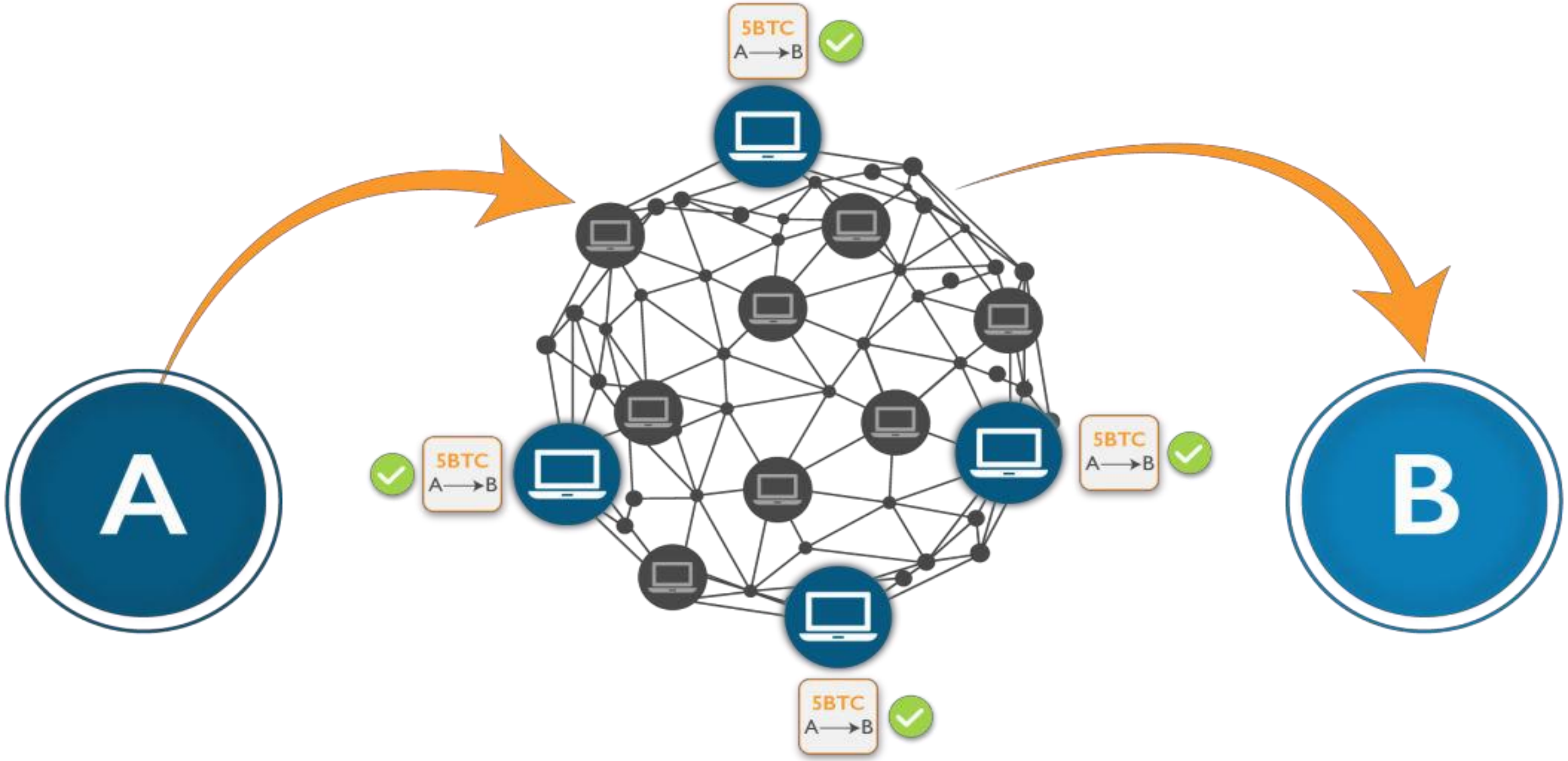
3

Independent confirmation of the new blocks by each node and get together into a chain

4

Independent selection, by every node, of the chain with the most cumulative computation demonstrated through proof of work

Independent Verification of Transactions

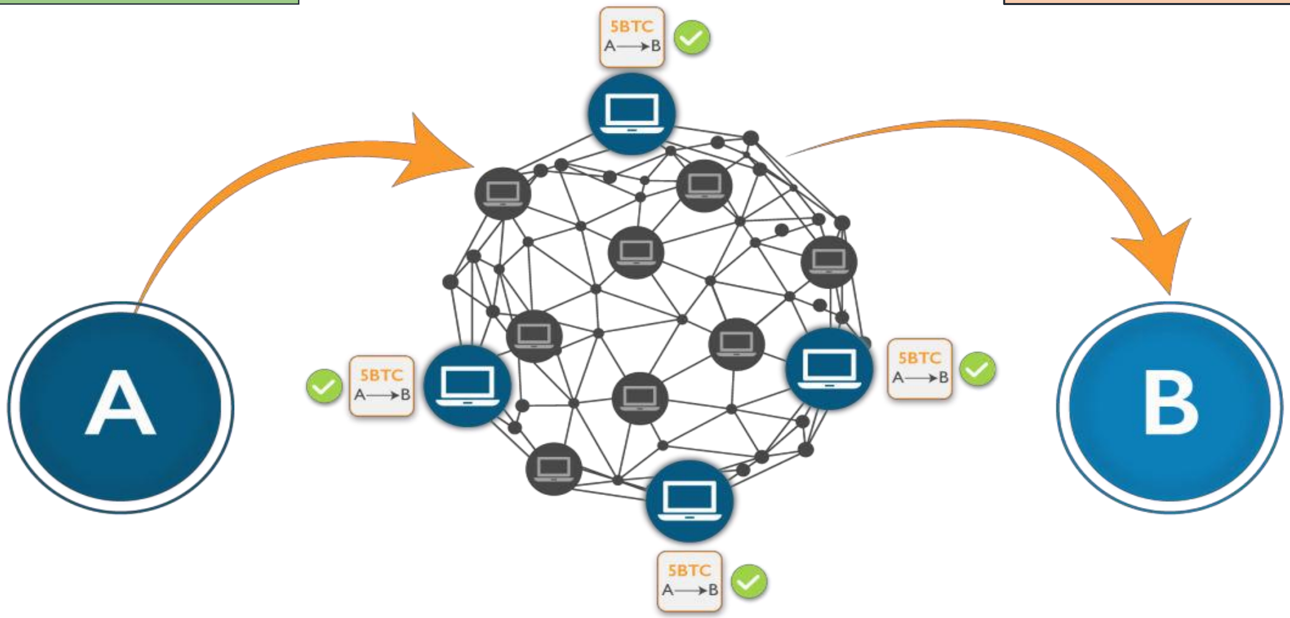


Let's say, Andy initiates a transaction (of 5 BTC to Bob) in a bitcoin network:

Independent Verification of Transactions

1. Before sending transactions to its neighbours, each bitcoin node that gets the transaction will at first confirm the transaction

2. This ensures that only valid transactions are proliferated across the system while invalid transactions are rejected of at the first node which receives them

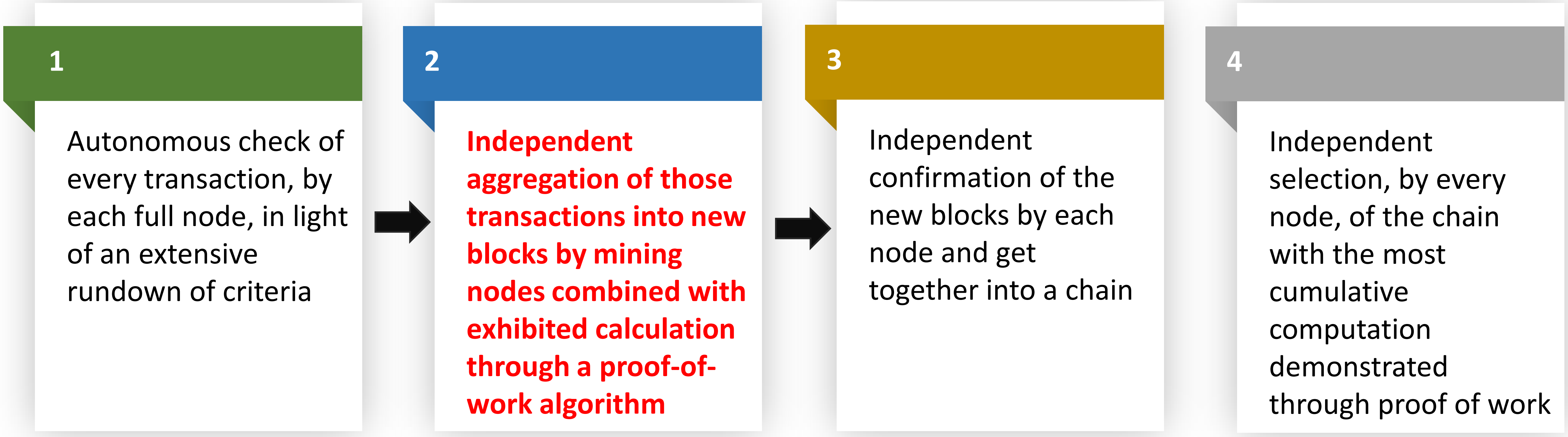


Checklist of transaction

Every node confirms each transaction against a long agenda of criteria:

- Transaction's punctuation and information stack has to be ins yc with the bitcobg protocal/
- Neither lists of inputs or outputs are empty
- The transaction size in bytes is less than MAX_BLOCK_SIZE
- Each output value, and also the aggregate, must be inside the permitted scope of qualities (under 21m coins, more than 0)
- A matching transaction in the pool, or in a block in the main branch, must exist
- For each input, if the referenced output exists in any other transaction in the pool, the transaction must be rejected
- For each input, the referenced output must exist and cannot already be spent
- Reject if the sum of input values is less than sum of output values

Combination of Verified Transactions



1

Autonomous check of every transaction, by each full node, in light of an extensive rundown of criteria

2

Independent aggregation of those transactions into new blocks by mining nodes combined with exhibited calculation through a proof-of-work algorithm

3

Independent confirmation of the new blocks by each node and get together into a chain

4

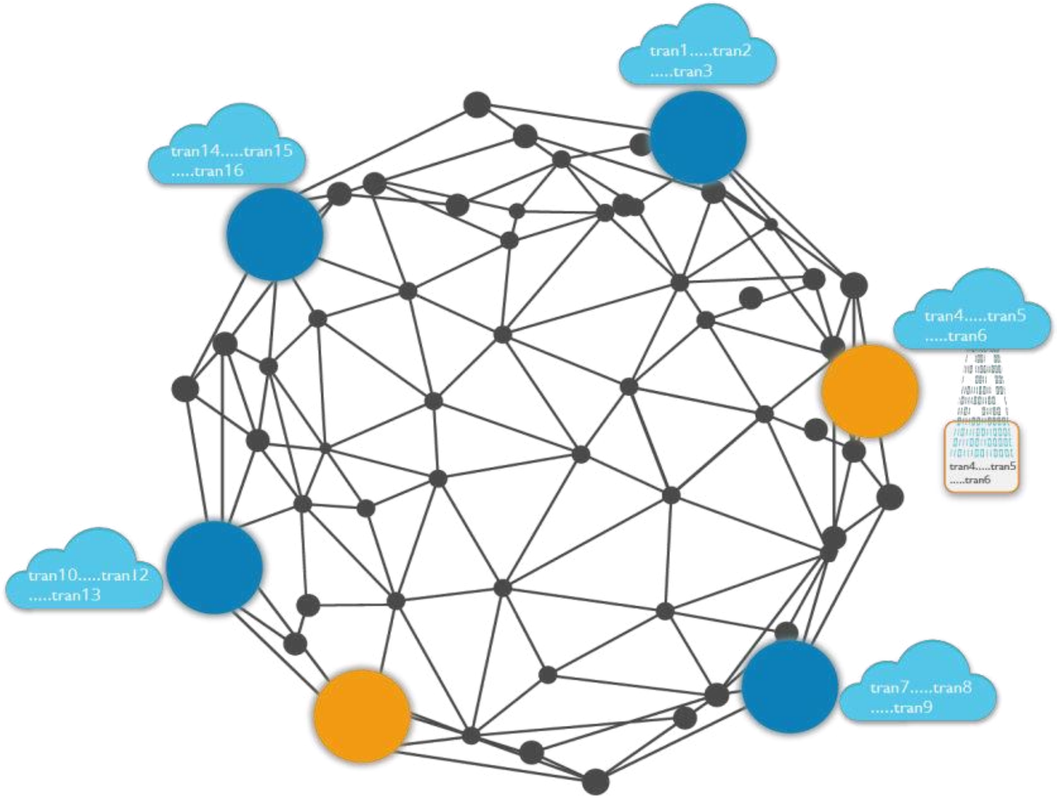
Independent selection, by every node, of the chain with the most cumulative computation demonstrated through proof of work

Combining Transactions into Blocks

By autonomously confirming each transaction as it is received and before transmitting it, each node mainitnis a pool of valid (however unconfirmed) transactions known as the **transaction pool, memory pool or mempool**

Transaction reaches Mining nodes it has collected, validates, and relays new transactions just like other node

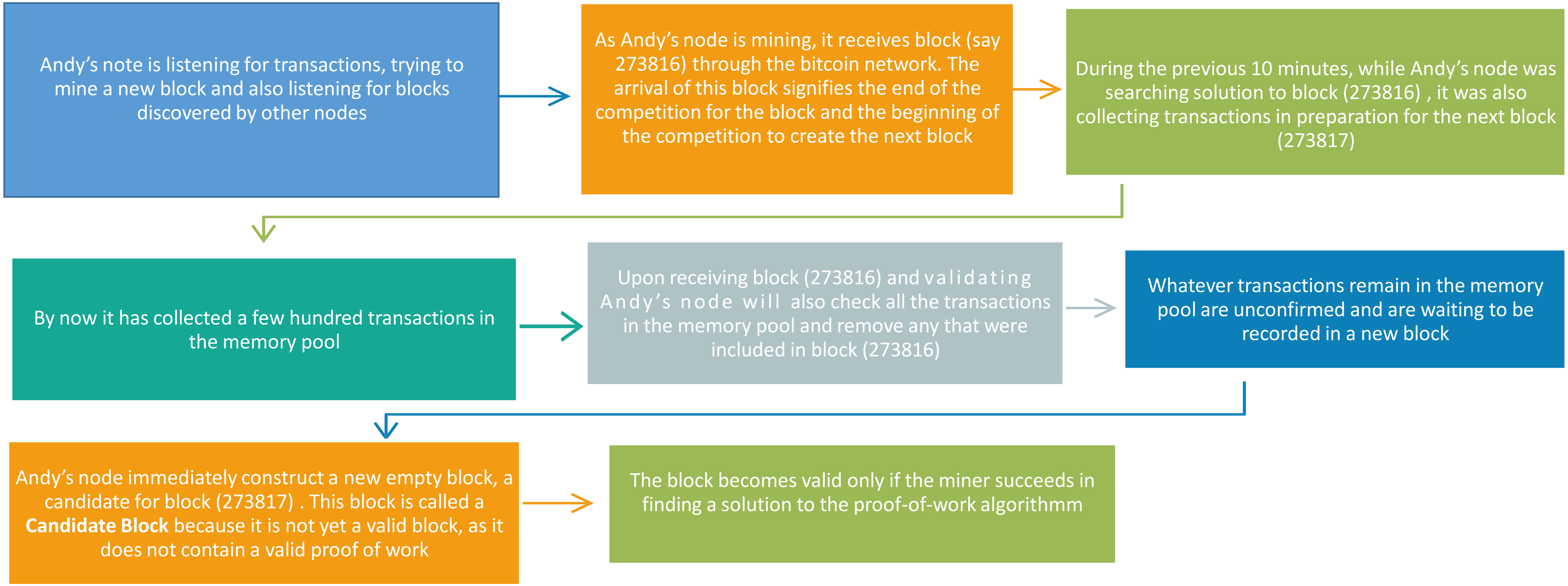
Unlike other nodes, miner node will then aggregate these transactions into a candidate block



In the network shown above, the miner nodes (in orange colour) aggregates the transactions in a candidate block

Combining Transactions into Blocks

Example: Let's say Andy is a miner. A mining code maintains a local copy of the BLOCKCHAIN, the list of all blocks created. Since the beginning of the BITCOIN system in 2009



Andy's note is listening for transactions, trying to mine a new block and also listening for blocks discovered by other nodes

As Andy's node is mining, it receives block (say 273816) through the bitcoin network. The arrival of this block signifies the end of the competition for the block and the beginning of the competition to create the next block

During the previous 10 minutes, while Andy's node was searching solution to block (273816) , it was also collecting transactions in preparation for the next block (273817)

By now it has collected a few hundred transactions in the memory pool

Upon receiving block (273816) and validating Andy's node will also check all the transactions in the memory pool and remove any that were included in block (273816)

Whatever transactions remain in the memory pool are unconfirmed and are waiting to be recorded in a new block

Andy's node immediately construct a new empty block, a candidate for block (273817) . This block is called a **Candidate Block** because it is not yet a valid block, as it does not contain a valid proof of work

The block becomes valid only if the miner succeeds in finding a solution to the proof-of-work algorithmm

Fabrication of a Block Header

Fabrication of a Block Header

> Now, Andy needs to construct the block header after collecting all the transactions in a block.

fabrication OF a Block Header

To construct the block header, the mining node needs to fill in six fields, as listed in the table:

Size	Field	Description
4 bytes	Version	To construct the block header, the mining node needs to fill in six fields, as listed
32 bytes	Previous Block hash	A reference to the hash of the previous (parent) block in the chain
32 bytes	Merkle Root	A hash of the root of the ŵerkle tree of this ďloĐk’s Transactions
4 bytes	Timestamp	The approximate creation time of this block (seconds from Unix Epoch)
4 bytes	Difficulty Target	The proof-of-work algorithm difficulty target for this block
4 bytes	Nonce	A counter used for the proof-of-work algorithm

Mining

Mining

> Andy started Mining the block once his node has all the fields filled in the block header in the blockchain.

Mining

With all the other fields filled, the block header is now finalized and the process of mining can begin

01

The goal is now to find a none value which results in a block header and hash that is less than the difficulty target of the block

02

The mining hub should test billions or trillions of nonce esteems before a nonce is found that fulfills the prerequisite

03

Since an approve piece has been developed by Andy's hub, it is the ideal opportunity for Andy's equipment mining apparatus to "mine" the square, to discover an answer for the verification of-working calculation that makes the square legitimate

Mining

> Presently, to approve the piece as per the confirmation of-work calculation, Andy's mining hub needs to achieve the trouble target. How about we perceive how the trouble is spoken to?

Portrayal of Difficulty

> The block contains the trouble focus, in a documentation called "difficulty bits" or just "bits“
> Let’s say a block has 0x1903a30c as the difficulty bits. This notation expresses the difficulty target as a coefficient/exponent format, with the first two hexadecimal digits for the exponent and the next six hex digits as the coefficient

The formula to calculate the difficulty target from this illustration is:

target = coefficient * 2^(8 * (exponent – 3))

Condition of Difficulty

- Difficulty recalculation happens automatically and on every full node
- Every 2,016 blocks, all nodes recalculate the proof-of-work difficulty level
- The condition for retargeting difficaluty measures the time it took to find the last 2,016 block and looks at that to the typical time of 20,160 minutes
- The proportion between the real timespan and coveted timespan is figured and a comparing change (up or down) is made to the difficulty

The equation can be summarized as:

New Difficulty = Old Difficulty * (Actual Time of Last 2016 Blocks / 20160 minutes)

Successful Mining

Successful Mining

> “We have seen that Andy’s mining node has worked hard to reach the difficulty target. Let’s see what happens next.”

Successfully Mining the Blocks

As we saw earlier, Andy's node has created a candidate block and prepared it for mining.

Andy has several hardware mining devices with ASIC, where hundreds of thousands of integrated circuits run the SHA256 encryption algorithm in parallel at unbelievable speeds

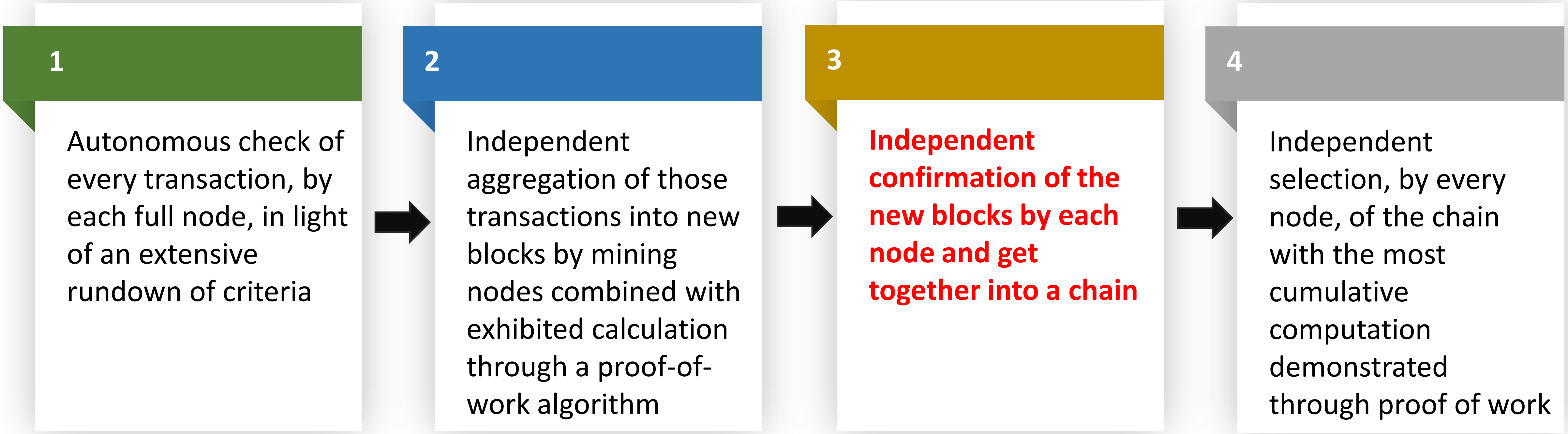
The mining code running on Andy's desktop transmits the block header to his mining hardware, which starts testing trillions of nonces per second

Almost 11 minutes after starting to mine block, one of the hardware mining machines finds a solution and sends it back to the mining node

Immediately, Andy's mining code transmits the block to all its peers.

They receive, validate, and then propagate the new block. As the block ripples out across the network

Independent Confirmation of each Block



1

Autonomous check of every transaction, by each full node, in light of an extensive rundown of criteria

2

Independent aggregation of those transactions into new blocks by mining nodes combined with exhibited calculation through a proof-of-work algorithm

3

Independent confirmation of the new blocks by each node and get together into a chain

4

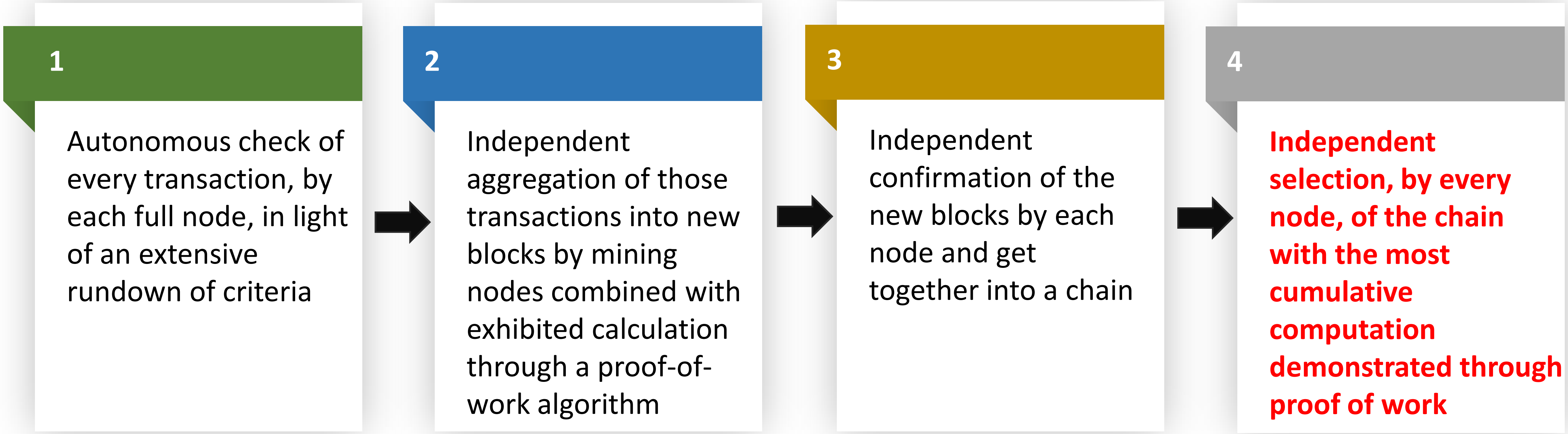
Independent selection, by every node, of the chain with the most cumulative computation demonstrated through proof of work

Validation of the New Block

- In Bitcoin's consensus process, every new block is validated by every node on the network
- This makes sure that only valid blocks are propagated on the network
- Nodes validates the block by verifying it against a long list of criteria that must all be met

The list includes:
The block structure is syntactically valid
The block header hash is less than the target difficulty (enforces the proof-of-work)
The block timestamp is less than two hours in the future (allowing for time errors)
The block size is within acceptable limits
The first transaction (and only the first) is a coinbase generation transaction
All transactions within the block are valid using the transaction checklist discussed independent verification of transactions.

Independent Aggregation of the Block in the Chain



1

Autonomous check of every transaction, by each full node, in light of an extensive rundown of criteria

2

Independent aggregation of those transactions into new blocks by mining nodes combined with exhibited calculation through a proof-of-work algorithm

3

Independent confirmation of the new blocks by each node and get together into a chain

4

Independent selection, by every node, of the chain with the most cumulative computation demonstrated through proof of work

The Main Chain

The chain of blocks with the most cumulative difficulty associated with it

Under most conditions this is likewise the chain with the most blocks in it, unless there are two equivalent length chains and one has more confirmation of work

When a new block is received, a node will try to slot it into the existing blockchain

The node will look at the block's "previous block hash" field, which is the reference to the new block's parent

Then, the node will attempt to find that parent in the existing blockchain

Most of the time, the parent will be the "tip" of the main chain, meaning this new block extends the main chain

Orphan Blocks

If a valid block is received and no parent found in the existing chains, that block is considered an "orphan"

Orphan blocks are saved in the orphan block pool where they will stay until their parent is received

Once the parent is received and linked into the existing chains, the orphan can be pulled out of the orphan pool and linked to the parent, making it part of a chain

Orphan blocks usually occur when two blocks that were mined within a short time of each other are received in reverse order (child before parent)

Bitcoins are Assembled in the Longest Chain

By choosing the greatest difficulty chain, all nodes in the long run accomplish organize wide consensus

Brief errors between chains are settled in the long run as more proof of work is included, broadening one of the conceivable chains

Mining nodes "vote" with their mining power by picking which chain to extend out by mining the next block

When they mine another block and expand the chain, the new block itself speaks to their vote

Difficulties in Solo Mining

Difficulties in Solo Mining

> So we have seen how Solo Mining of Bitcoins works. However, solo Mining is becoming difficult, let’s find out why?

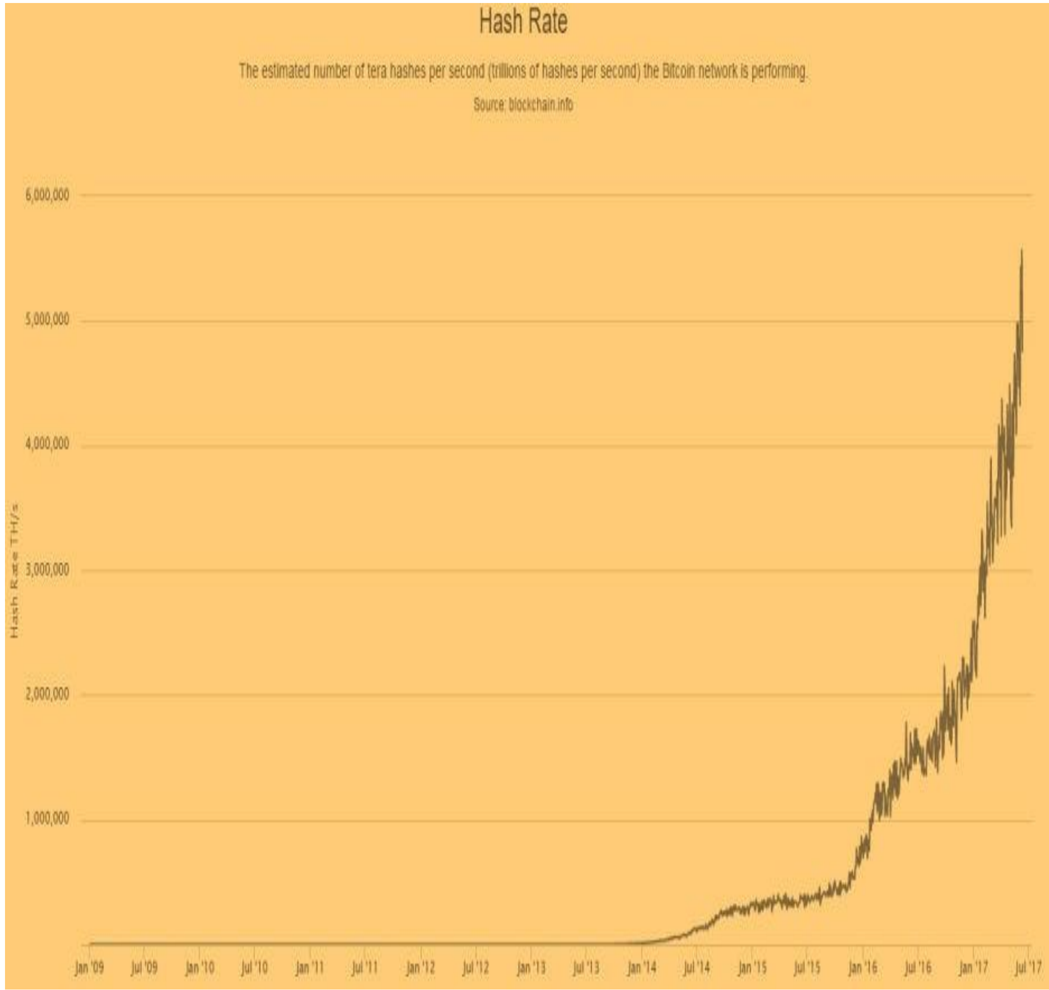
Mining & the Hashing Race

- Bitcoin mining is a to a great degree aggressive industry

- The hashing power has expanded exponentially each year of bitcoin's presence

- The competition between miners and the growth of bitcoin has resulted in an exponential increase in the hashing power (total hashes per second across the network)

- The acquaintance of ASIC mining lead with another goliath jump in mining power, by setting the SHA256 work straightforwardly on silicon chips particular with the end goal of mining



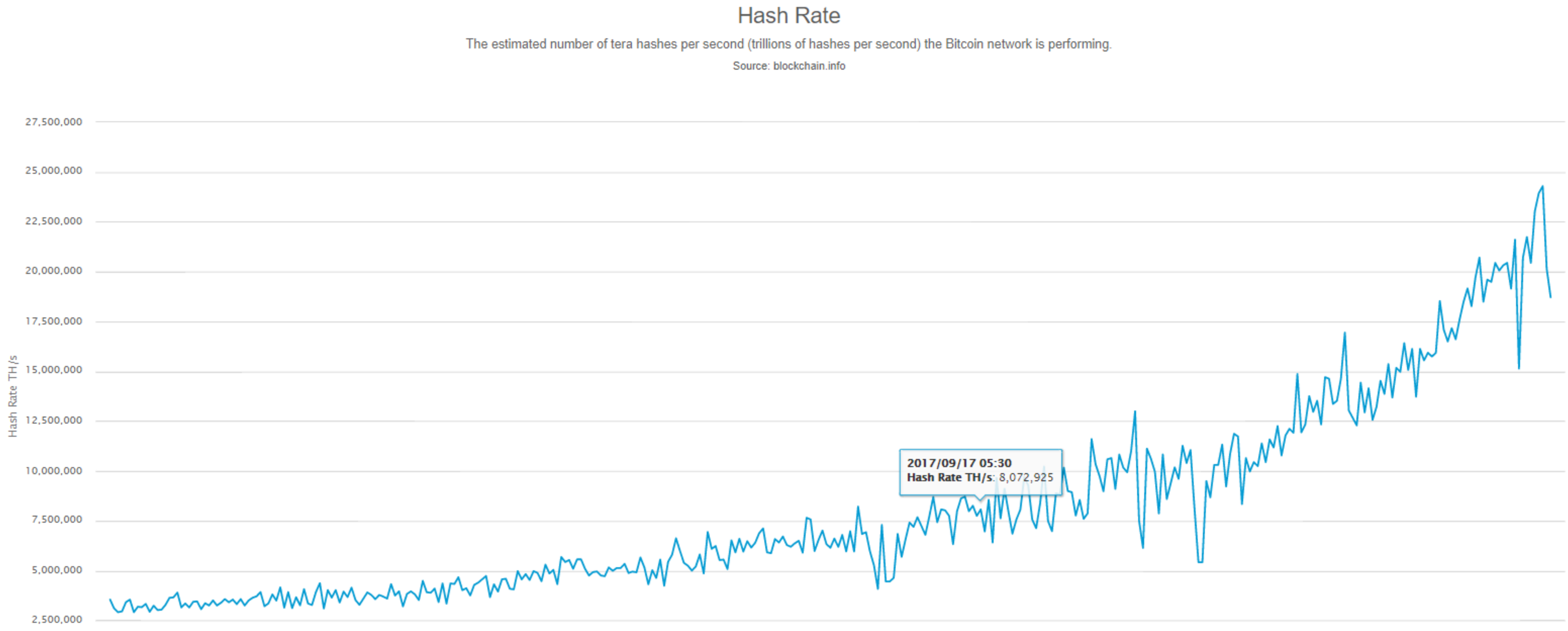
Hash Rate

The estimated number of tera hashes per second (trillions of hashes per second) the Bitcoin network is performing.

Source: blockchain.info

Difficulty Rises with the Hashing Power of the Miners

As the amount of hashing power applied to mining bitcoin has exploded, the difficulty has risen to match it



Hash Rate

The estimated number of tera hashes per second (trillions of hashes per second) the Bitcoin network is performing.

Source: blockchain.info

2017/09/17 05:30
Hash Rate TH/s: 8,072,925

Total hashing power, in gigahashes per second, over two years

Mining: By pool of Miners

Mining: By pool of Miners

> “ Since Solo Mining has become impossible with the increasing difficulty in the Bitcoin network, so miners have started mining in the Pools. ”

Mining: By pool of Miners

Mining pools coordinate many hundreds or thousands of miners, over specialized pool-mining
protocols

The pool server will periodically make payments to the miners' bitcoin addresses, once their share of the rewards has reached a certain threshold

Successful blocks pay the reward to a pool bitcoin address, rather than individual miners

The individual miners configure their mining equipment to connect to a pool server, after creating an account with the pool

Their mining hardware remains connected to the pool server while mining, synchronizing their efforts with the other miners

The pool miners share the effort to mine a block and then share in the rewards

Pooled mining: Difficulty of Targets

- The mining pool sets a lower difficulty target for earning a share, typically more than 1,000 times easier than the bitcoin network’s difficulty so as all the solo miners are able to mine and earn bitcoins.

- When someone in the pool successfully mines a block, the reward is earned by the pool and then shared with all miners in proportion to the number of shares they contributed to the effort

Pooled mining: Difficulty of Targets

How does a mining pool measure the individual contributions, so as to fairly distribute the rewards, without the possibility of cheating?

The answer is to use bitcoin's proof-of-work algorithm to measure each pool miner's contribution, but set as a lower difficulty so that even the smallest pool miners win a share frequently enough to make it worthwhile to contribute to the pool

Pooled Mining: Types

There are several types of mining pool payout systems. The different payout systems are summarized below:

1. Pay Per Share (PPS)

- The Pay-per-Share (PPS) approach is to offer an instant flat payout for each share that is solved
- The payout is offered from the pool's existing balance and can therefore be withdrawn immediately, without
- waiting for a block to be solved or confirmed
- The possibility of cheating the miners by the pool operator and by timing attacks is thus completely eliminated
- This method results in the least possible variance for miners while transferring all risk to the pool operator
- The resulting possibility of loss for the server is offset by setting a payout lower than the full expected value

Pooled Mining: Types

There are several types of mining pool payout systems. The different payout systems are summarized below:

2. Pay Per Last N Shares (PPLNS)

- PPLNS has a higher payout. This is for people trying to mine as fast as possible
- PPLNS will give you wide fluctuations in your 24 hour payout, but for hardcore miners, the law of large numbers states you will earn more this way

Thank You

Email us – support@intellipaat.com

Visit us - https://intellipaat.com