How Does Bitcoin Work

This set of slides was developed based on the reference books as listed in the course document.

Problem: Create Accounts Need Permission

If you want to create an account, it is the administrator's job to assign you an unused account number then set you up with some sort of username (and password).

Later, when you ask the administrator to make a payment on your behalf, the administrator knows it is really you making the request.

Is there a way you can open an account without having to ask administrator's permission? (i.e. no need to use a centralized approach)

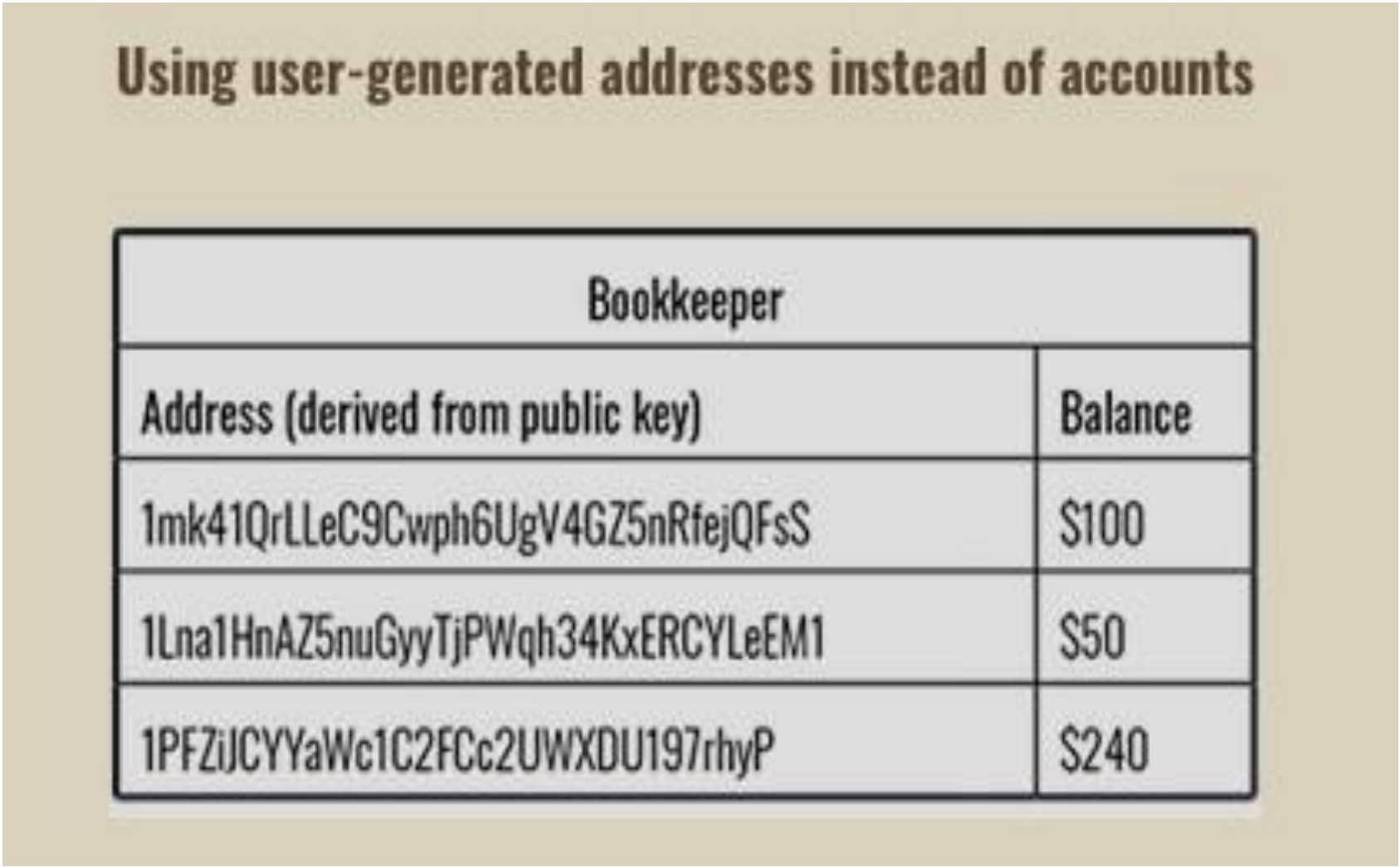
Solution: Use Public Key as Account Number

By using public keys as account numbers, anyone can create their own accounts with their own computer without having to ask an administrator for an account number.

So you create an account by

- picking a random number (your private key) and
- doing some maths on it to get your *public key.* Then in **Bitcoin and most other cryptocurrencies**,
- mathematically derives account numbers from public keys (not public keys themselves), and
- are called **addresses**.

Addresses

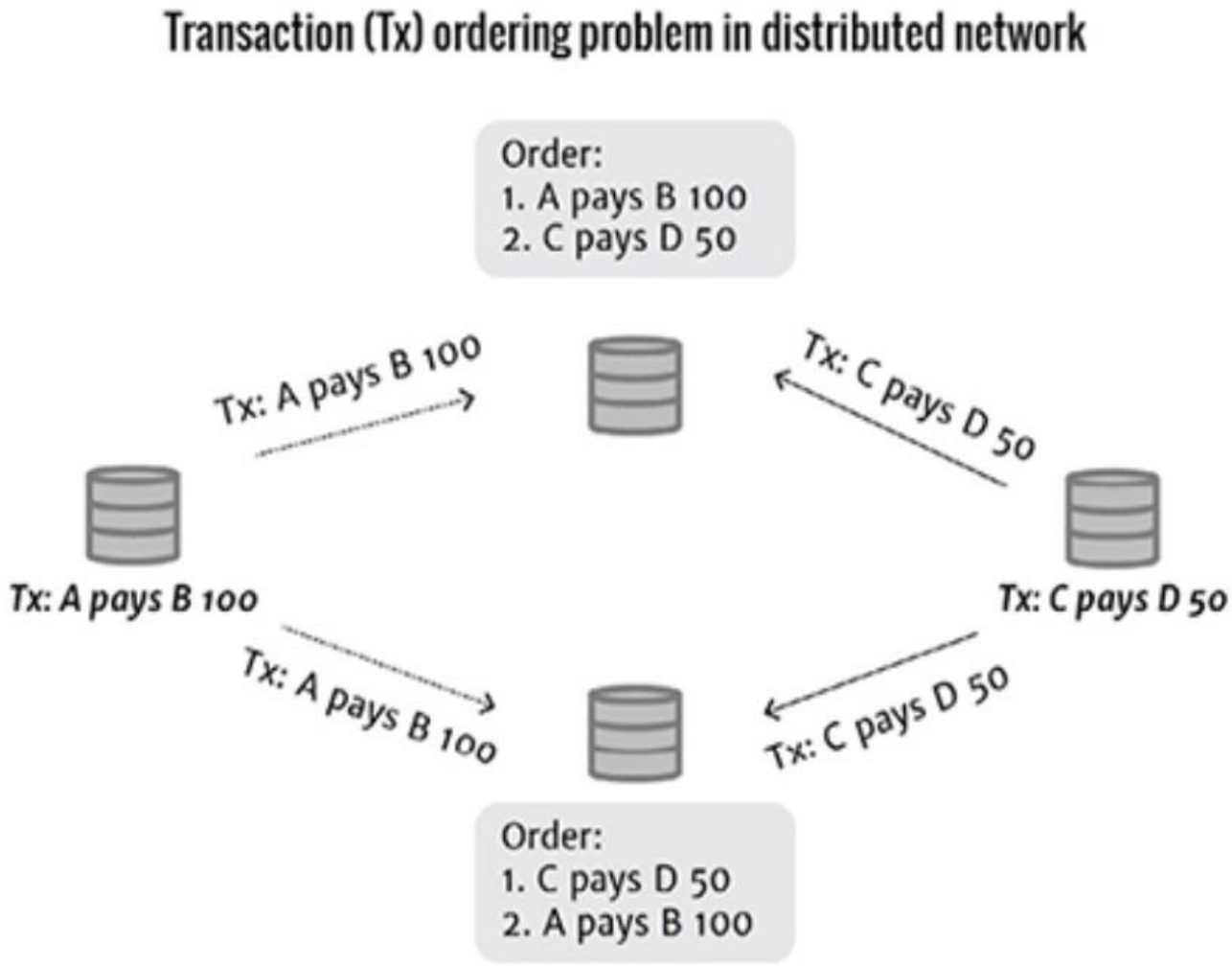

Using user-generated addresses instead of accounts
Bookkeeper
Address (derived from public key)
Balance
1mk41QrLLeC9Cwph6UgV4GZ5nRfejQFsS
$100
1Lna1HnAZ5nuGyyTjPWqh34KxERCYLeEM1
$50
1PFZiJCYYaWc1C2FCc2UWXDU197rhyP
$240

Authentication Problem Also Solved

Public/private key pairs also solve the authentication problem:

- You don’t have to log in to prove that you are the account holder.
- When *sending a payment instruction* you *digitally sign the transaction* with your *private key*, and this signature proves to the administrator that the instruction is indeed coming from you, the account holder.
- You can *create and sign the transaction offline* without being connected to any network. When you broadcast the signed transaction to the administrator, all the administrator has to do is check that the digital signature is valid for the respective account number,
- No need to maintain a list of usernames and passwords for you and all transacting parties.

Transactions are not Directly Recorded in Bitcoin



Block Creation in Bitcoin

Transactions are recorded by grouping them in batches called **blocks**.

Blocks are created much less frequently than transactions, so it is more likely that a block reaches all bookkeepers in the network before another one is created.

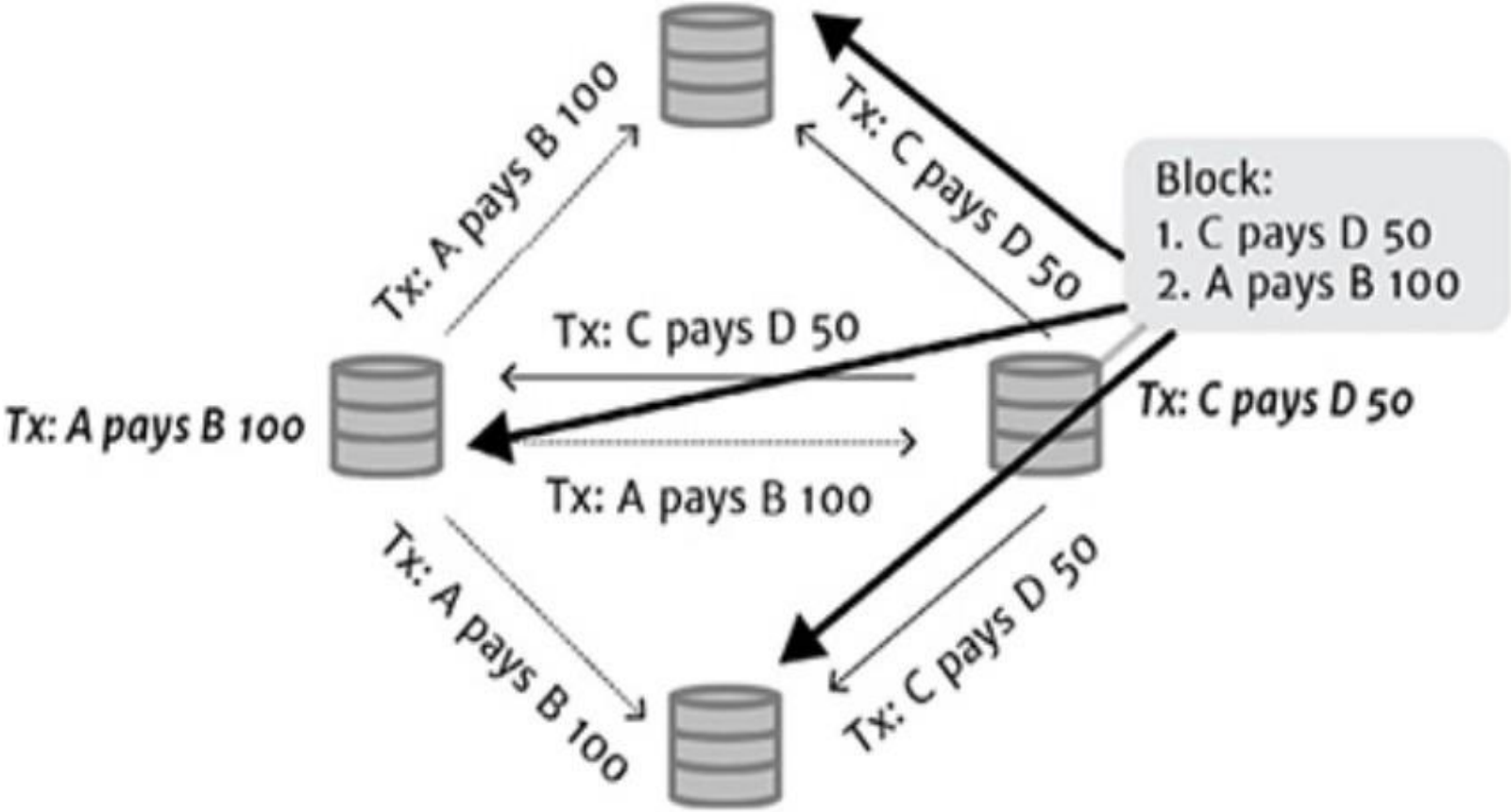
In Bitcoin, *blocks are created every 10 minutes* on average. Different cryptocurrencies have different block creation target times.

Block Creation in Bitcoin

A bookkeeper now performs two functions:

- Validating and propagating ‘pending’ transactions
- Validating, storing, and propagating blocks of transactions

Block Creation in Bitcoin



Proof-of-Work

In Bitcoin, anyone is able to create blocks and send them around the network. But then how do we *control the speed* at which blocks are created?

Answer: **Proof-of-Work**

> The is extremely elegant. All block-creators have to play and win at a game of chance, a game that in aggregate, over the whole network, takes some specific amount of time to play (say 10 minutes on average).

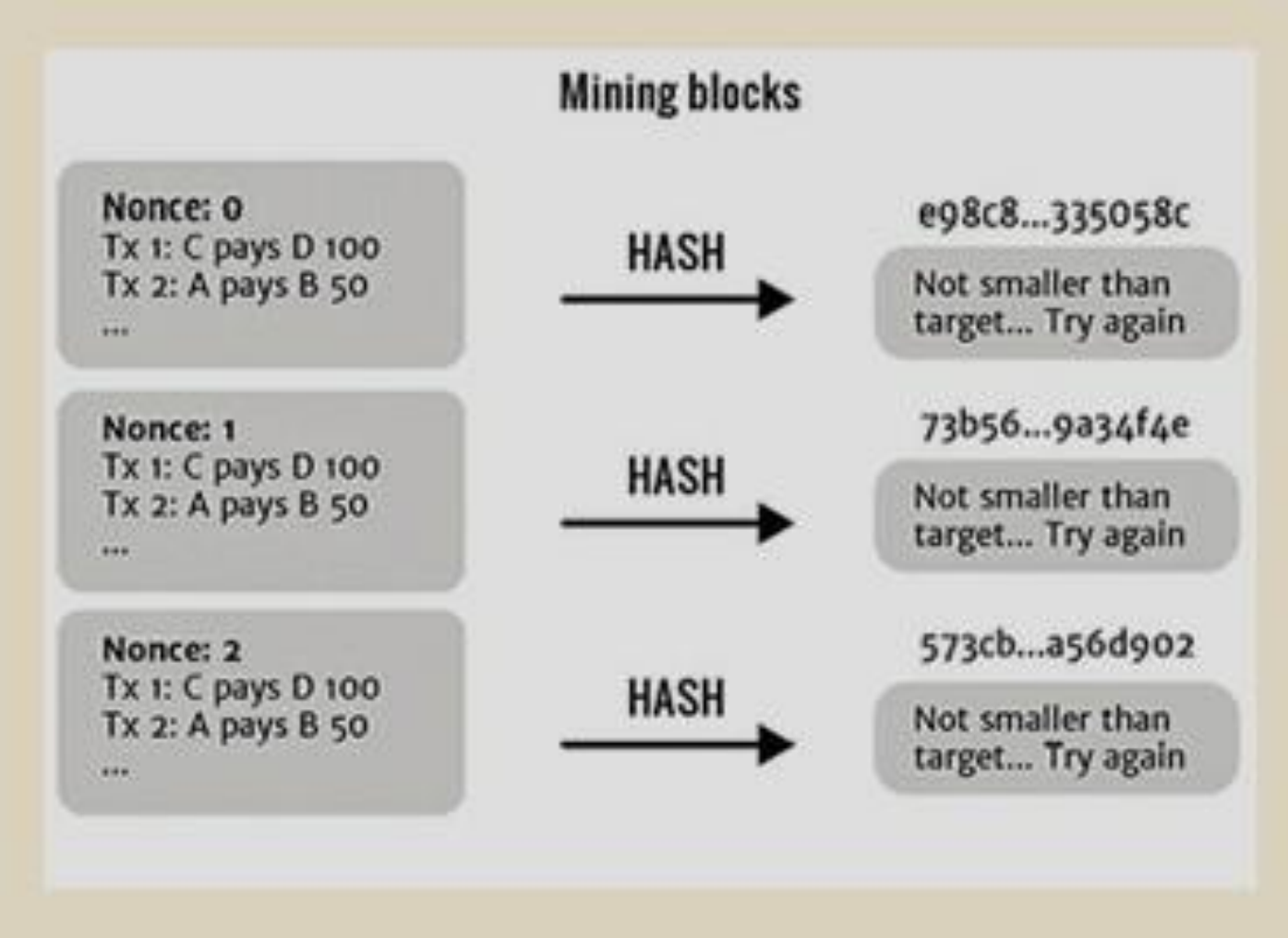
Proof-of-Work

In the game of chance, each block-creator

- takes a bunch of transactions that they know about, but which have not yet been included in any previous blocks, and builds a block out of them, in a specific format
- then calculates a cryptographic hash from the block's data
- if the *hash* of the block is *smaller than a target number*, then this block is considered a valid block which all bookkeepers should accept
  - if not, *change nonce* and try again

Mining Blocks

This process is called *mining*.



Problem: Incentivising Block-creators

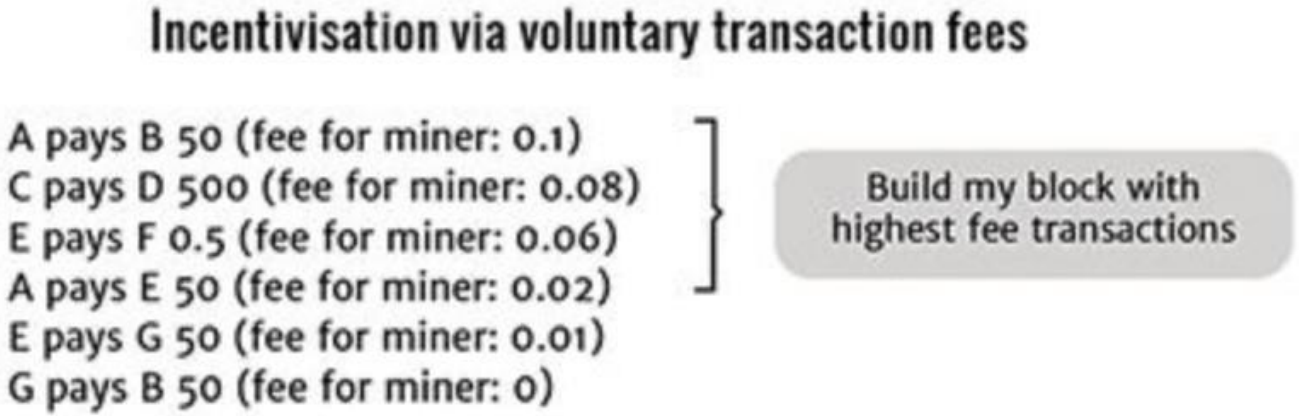
But all of this tedious hashing needs resources: computers, electricity, bandwidth… and this all *costs money*. Why should anyone bother creating blocks? How can we incentivise the block-creators to create blocks and keep the system running?
Solution: Transaction Fees

Answer: **Transaction Fees**

> The solution is to pay the block-creators for their time and resources! But *who is going to pay* them and *in what currency*?

Transaction Fees

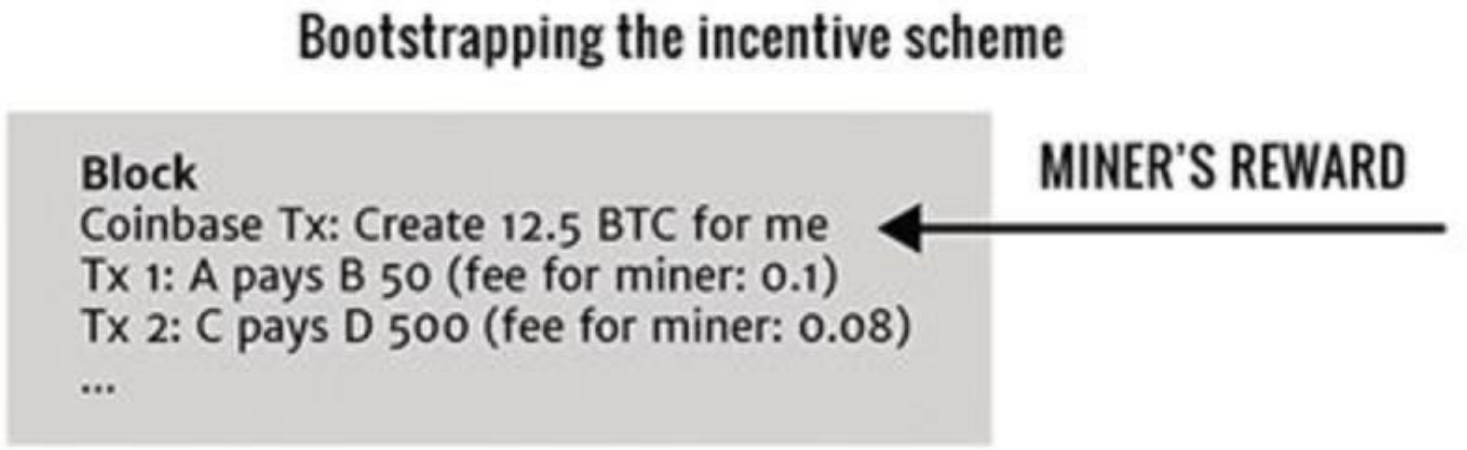
Bitcoin's solution is a *market-based* approach where people creating transactions add their own voluntary transaction fees, and the block-creators can prioritise those transactions with higher fees over those with lower fees.



Fees tend to go up in times where there are many transactions queuing up to get into blocks, and down again in times with fewer transactions.

Block Reward

The very first transaction in a block is called the *coinbase* transaction. This coinbase transaction is special because it is the *only transaction that creates bitcoins*. All *other* transactions *move bitcoins* between addresses



The block-creator can create a transaction that pays any address (usually themselves) any number of bitcoins, up to a limit specified by the Bitcoin protocol.

They receive valuable BTC in return for spending resources doing the tedious hashing to create valid blocks.

Block Reward

This limit of reward:

- was 50 BTC per block in 2009 and
- reduces by half every 210,000 blocks, which at 10 minutes per block, is about every 4 years.
- *is **12.5 BTC** in mid-2018*,
- will be further reduced in around May 2095 (with the next reduction to occur on block 630,000)

These block rewards

- have created around 17 million bitcoins to date, and
- will at the maximum create about 21 million bitcoins (the last of which should be created a little before the year *2140)*

Unless the rules change. This block reward is the mechanism that keeps block-creators creating blocks.

Blocks Chained Together

A block chain[96] where each block includes the hash of the previous block, rather than a sequential block number.

Problem: Creating Blocks Too Fast

With more people throwing more hashing power (i.e., computers) at the block creation process, blocks would be created faster and faster. This is not good.

When blocks are created slowly, so that the bookkeepers have a better chance of staying in consensus.

Creating Blocks Too Fast

With more people throwing more hashing power (i.e., computers) at the block creation process, blocks would be created faster and faster. This is not good:

- Because when blocks are created slowly, the bookkeepers have a better chance of staying in consensus.

Solution: Difficulty

The network needs to *self-correct and slow down* if blocks are created more quickly than the target of one block every ten minutes.

The answer lies in changing the *target number* for the *hash* calculation:

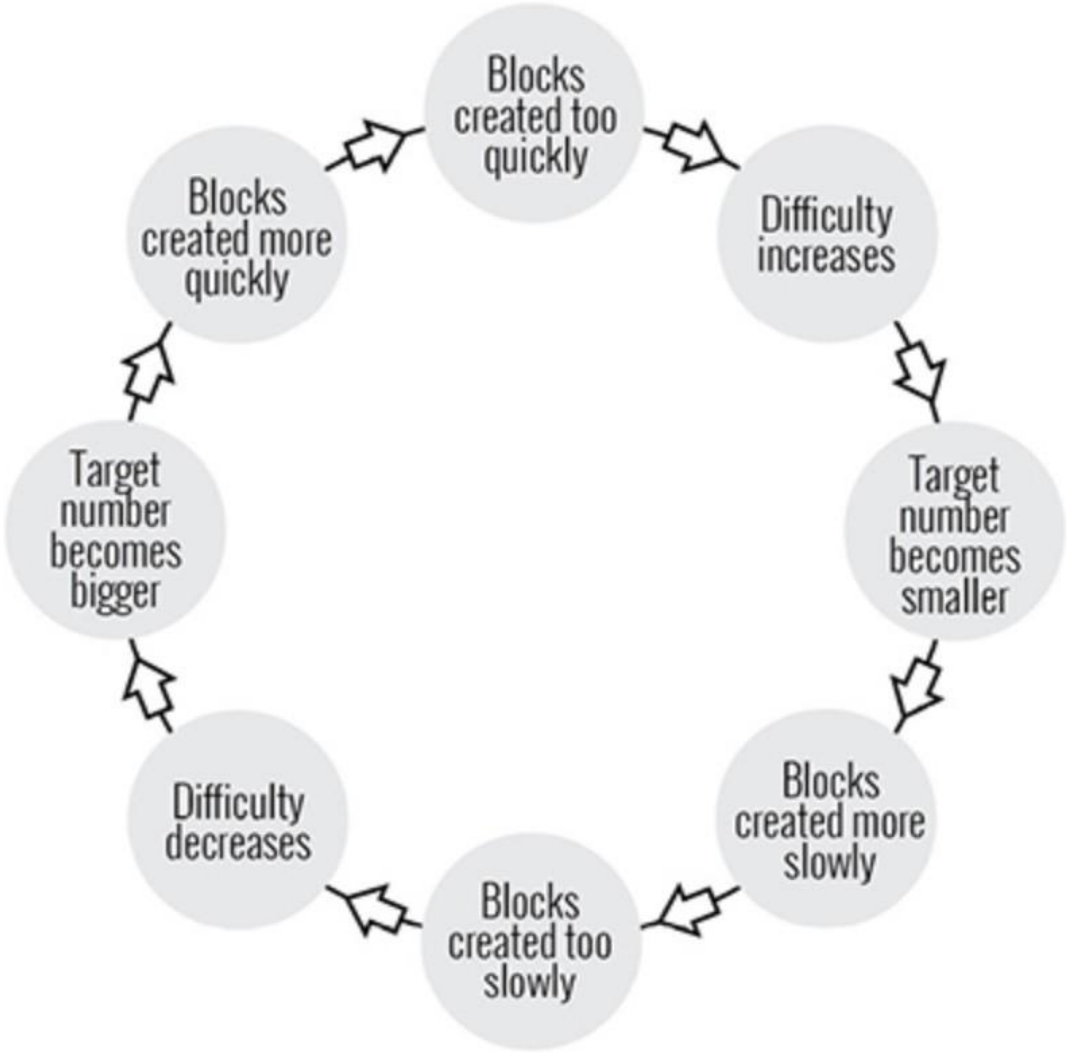
- Variations in this target number can make it easier or harder for the network, in aggregate, to find hashes that fall below this number.

Solution: Difficulty

In Bitcoin, the *target number* is:

- mathematically calculated from a number called the 'difficulty'
- changed every 2016 blocks (which takes about two weeks at ten minutes per block)
- changed according to a *formula* that uses the elapsed time it took to mine the previous 2016 blocks:
  - The faster the previous 2016 blocks were created, the more the difficulty increased
- and the difficulty are *inversely related*, so as difficulty increases, the target number becomes smaller, making it harder and therefore slower to find valid blocks.

Difficulty is Self-balancing



Solution: Difficulty is Self-balancing

The network is beautifully *self-balancing*.

If more hashing or mining power is added

- blocks get created faster for a period of time until the next difficulty change,
- after which it becomes harder to find valid blocks, slowing block creation down.

If mining power leaves the network,

- blocks take longer to be found, until the next time the difficulty changes,
- when difficulty decreases, and blocks become easier to find.

And this is all done *without a central coordinator*.

Longest Chain Rule

Bitcoin also uses *the longest chain rule*.

If a miner sees two valid blocks at the same block height then they can mine on either block (usually the first seen) and would keep the other one ‘in mind’.

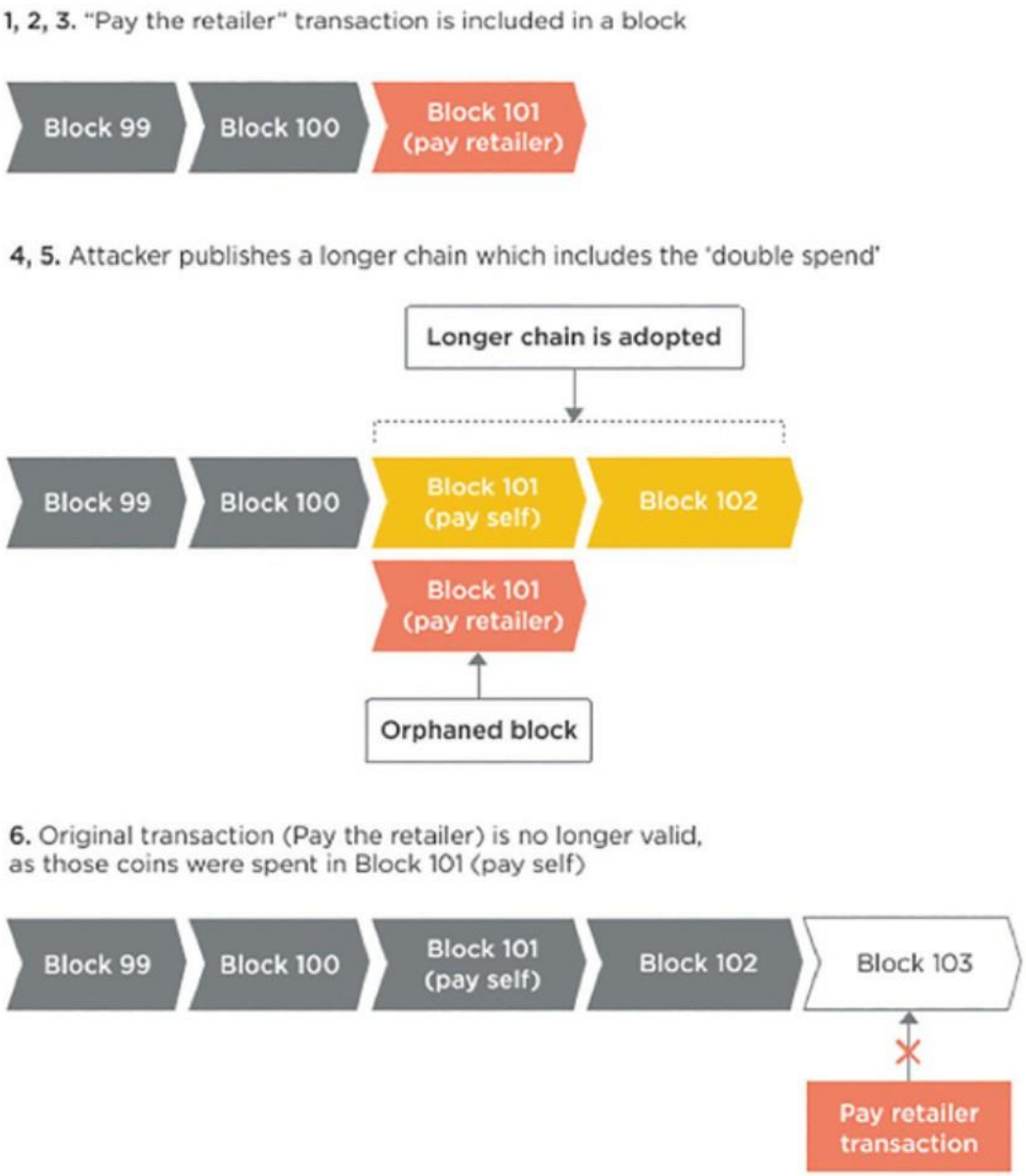
Double Spend

Although the longest chain rule seems sensible, it can be used to create mischief in a deliberate double spend. Here is how you could do it:

1. Create two transactions using the same bitcoins: one payment to an online retailer, the other to yourself (i.e., to another address you control).
2. Only broadcast the transaction that is the payment to the retailer.
3. When the payment gets added in an ‘honest’ block the retailer sees this and sends you goods.
4. Secretly create a longer chain of blocks which excludes the payment to the retailer, and replaces it with the payment to yourself.
5. Publish the longer chain. If the other nodes are playing by the ‘longest chain rule,’ then they will reorganise their blockchains, discarding the honest block containing the payment to the retailer, replacing it with the longer chain you published. The honest block is said to be ‘orphaned’ and, to all intents and purposes, does not exist.
6. The original payment to the retailer will be deemed invalid by the honest nodes because those bitcoins have already been spent in your longer, substituted, chain.

You will have received your goods but the payment to the retailer will be rejected by the network.

Double Spend



How to double spend.

Solution to Double Spend

Wait About Six Blocks:

- Therefore, common advice for people receiving bitcoins is to wait for the transaction to be a few blocks deep (i.e., to have a few blocks mined on top of it).
- This gives comfort that the transaction is settled and can't easily be unwound
- At this point the amount of mining that has to be done *to create a competing chain longer than the existing chain is enormous*
- So rational miners would prefer to dedicate their hash power towards creating legitimate blocks, receiving the block reward and transaction fees, rather than trying to subvert the network

Solution to Double Spend

To put it another way, it is deliberately hard to generate a valid block. This is a reason why people say Bitcoin's blockchain is *immutable* and cannot be changed:

- only if more than 50% of the total hash power of the network is used to re-write blocks, then it will be able to do so, because it will create blocks faster than the other, less powerful, half.
- only if under a 51% attack:
  - smaller amounts of hash power can also be used to re-write the blockchain, but with a lower probability of success
  - 51% attacks have been successfully performed on unpopular coins with few miners.