**Bitcoin:**

Digital currencies have been around for decades, but only over the past 7 or so years has one really taken off: Bitcoin. Bitcoin enables digital payments between untrusted parties in a fully decentralized way, meaning with no central authority involved (no banks, credit card companies, governments, etc.).

Why use Bitcoin? One prosaic but representative example is transferring money internationally. Through Bitcoin, you can accomplish the same goal, saving an order of magnitude in both time and cost.

The basic element of Bitcoin is a Transaction.

**A Bitcoin Transaction**

1. One or more senders.

2. One or more receivers.

3. The amount of BTC (Bitcoins) transferred from each sender to each receiver.

4. A proof of ownership of the coins being transferred, in the form of a pointer back to the most recent transactions involving the transferred coins

5. A transaction fee, paid by the sender to the authorizer of the transaction.

A transaction is valid if:

1. It has been cryptographically signed by all of the senders. (This can be verified using the senders’ public keys.)

2. The senders really do own the coins being transferred.

**Blocks**

Transactions are added to the ledger in groups (rather than one-by-one), known as blocks. Specifically, a block has the following ingredients:

1. One or more transactions.

2. A hash of the previous block.

3. A nonce. (I.e., a bunch of bits that can be set arbitrarily.)



Blocks typically have on the order of 1000–2000 transactions (there is a 1 MB cap on the block size). The second ingredient imposes a natural linked list-type structure on the ledger, with the predecessor of block b2 being block b1 whose hash matches the hash stored in b2.

How do new blocks get added to the blockchain? Who can do it? Why should they bother? How can we make sure that everybody agrees on the contents of the blockchain? The two-prong solution to these questions is brilliant in its elegance.

1. Any user can authorize a block. Bitcoin incentivizes users to do authorizations through explicit monetary rewards (in BTC, naturally).

2. To avoid anarchy, authorizing a new block of transactions involves proof of work, meaning that the authorizer must solve a computationally difficult puzzle.

**The Computationally Difficult Puzzle**

A block b (consisting of transactions, the hash of the previous block, and the nonce) is valid if h(b) is sufficiently close to 0, where h is a pre-agreed upon hash function (currently, SHA-256, so h(b) is 256 bits). Thus, finding a valid block involves inverting a one-way/cryptographic hash function. By “sufficiently close to 0,” we mean that the leading £ bits of h(b) should all be 0, where £ is a parameter. Note that £ provides a knob to control the difficulty of the problem: the bigger the choice of £, the harder the problem.5 How is £ chosen? To keep the rate of valid block creation roughly constant, averaging around one block every ten minutes. The parameter £ is re-calibrated roughly every two weeks, based on the average time between blocks during this window. (So if the average time between blocks has dropped to 9 minutes, it’s time to increase £.) Why 10 minutes? Because to maintain the property that all miners are on the same page about what the current blockchain is, it seems essential to have block creation happening on a time scale slower than the latencies in the peer-to-peer network.

**Block Rewards and Bitcoin Mining**

Bitcoin mining refers to the process of finding new valid blocks. The intended behavior for a bitcoin miner is: choose a subset of the outstanding transactions that it knows about (e.g., the ones with the highest transaction fees), insert the hash of the current last block of the blockchain, arbitrarily set the bits in the nonce, and hope that the resulting block b is valid. Since h is a cryptographic hash function, the accepted belief is that there is no algorithm for finding a valid block that is smarter or faster than random guessing or exhaustive search. Assuming that SHA-256 acts like a random function, for any given block b, the probability that b is valid is 2−£. This means that the expected number of tries necessary to find a valid block is 2£. In practice, Bitcoin miners typically search for a valid block by fixing the set of transactions and varying the nonce until the block becomes valid. When a new valid block has been found, the miner is supposed to immediately broadcast it across the entire peer-to-peer network, so that it gets appended to the blockchain (and the miner can collect its reward, see below). If someone else announces a new valid block first, then the miner restarts this procedure, now using only transactions not already authorized by the new block, and using the hash of the new block.

1. A flat reward that does not depend on the contents of the block (other than it is valid). When Bitcoin debuted this reward was 50 BTC, but the protocol dictates that this amount gets cut in half every four years.

2. The sum of the transaction fees of the transactions in the block. Currently, transaction fees are non-zero but typically constitute only a few percent of the overall reward.8

**Forks**

Once in a while, two different bitcoin miners will discover valid blocks at roughly the same time. This results in a fork in the blockchain (Figure 2), where two valid blocks, each with its own set of transactions, point to the same previous block. There needs to be a mechanism for deciding which branch of the fork is the “right” one, for two reasons: (i) so that everybody knows which transactions have been authorized; and (ii) so that bitcoin miners know which block they should be trying to extend.

The Bitcoin protocol specifies the intended behavior when there’s a fork: a user should regard the longest branch as the valid one, breaking ties according to the block that it heard about first. When there is a fork as in Figure 2, it is completely possible that different users will have different opinions about which branch is the valid one (user 1 may have heard about b2 before b3, while user 2 heard about b3 before b2).

Eventually, some bitcoin miner will authorize a new block, which extends only one of the branches (depending on which hash the miner put in the new block). If there were previously branches with equal length, then this new block will break the tie and create a chain longer than any other. At this point, all users will again have a consistent view of the blockchain (as the longest chain). When this happens, blocks not on this longest chain are “orphaned,” and the transactions in them are not regarded as authorized. (Some of them may be authorized instead by a block on the longest chain.) Similarly, the creators of these orphaned blocks do not get any reward for them.

Blocks will occasionally get orphaned even when all miners are obediently following the protocol. Thus, one should not regard a transaction as authorized until it has been included in a block on the blockchain, and also been extended by another block

**Sybil Attacks**

Bitcoin users are identified with public keys. It’s not hard to generate many public keys, so many Bitcoin “users” might actually correspond to the same person. Deliberately creating multiple identities in a system is often called a Sybil attack. Sybil attacks plague many systems, but remarkably, they cause no issues in Bitcoin. Your influence in Bitcoin is determined entirely by the amount of computational power that you wield—the number of identities is irrelevant.