Crypto news?

Blockchain Structure

Happy Groundhog Day!

What is blockchain?

- A blockchain is an append-only data structure that records information in history
    - Append-only - items can be added, but cannot be changed or removed at any time
- A blockchain is split into **blocks** that **chain** together to make the entire structure, hence the name

Why is an append-only blockchain important for Bitcoin?

- To prevent fraudulent transactions
- Prevent people from creating their own coins
- Keep people from reversing transactions or double-spending their currency

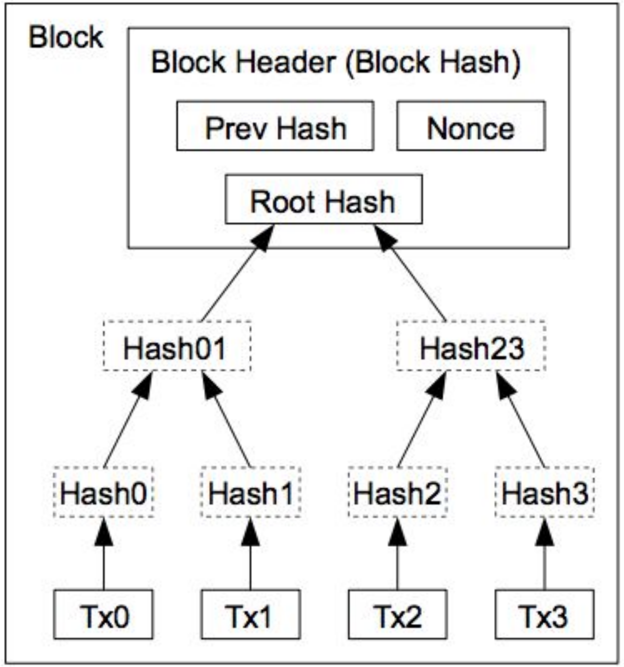
How does bitcoin’s blockchain achieve append-only?

- SHA-256
    - SHA-256 is a powerful one-way hash function used to give a unique identifier to each block on the blockchain
    - One-way - easy to hash x to y, but **nearly** impossible to retrieve the original value x from given hash value y
- Key properties of SHA-256
    - Pre-image resistance - given hash value h, difficult to find data d such that hash(d) = h
    - Collision resistance - given data d_1, difficult to find data d_2 such that hash(d_1) = hash(d_2) (also known as second-preimage resistance)
- Bitcoin uses SHA-256 applied twice to achieve this

Why use SHA-256?

- Security - It’s nearly impossible to break SHA-256, which keeps transactions safe and secure on the network
- Difficulty - takes a lot of computing power to find the right hash for a block, since similar inputs give vastly different hashes
- Verification - anyone can verify the validity of a block by simply re-computing the hash of that block
    - Keeps network efficient and secure because running the block through SHA-256 with the published information is easy and quick.
    - Why does this contribute to network efficiency?

What are the components of a Bitcoin block?



Transactions Hashed in a Merkle Tree

What are the components of a Bitcoin block?

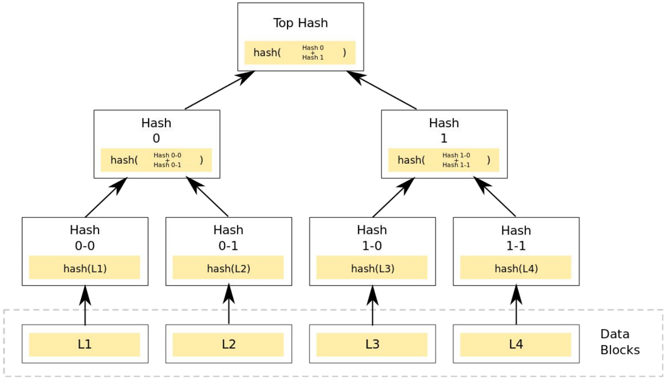
- Three main parts
    - There are other smaller parts of metadata that we won’t discuss here
- Previous block hash
- List of Transactions
- Nonce (also known as the “winning” nonce)

What are the components of a Bitcoin block?

- Previous block hash
  - The previous block will have a hash that has been published to the network
  - Why does including the previous hash increase security?
  - How to choose a previous block hash?
    - More on this next week when we discuss miners

What are the components of a Bitcoin block?

- List of Transactions
  - Arguably most important part of the block (and the whole reason people are using bitcoin)
  - Transactions are included in the block
    - Merkle Tree - hash-based binary tree of transactions
    - Including every transaction would take up a huge amount of space - how to fix this?
    - Pruning - does bitcoin actually do this?

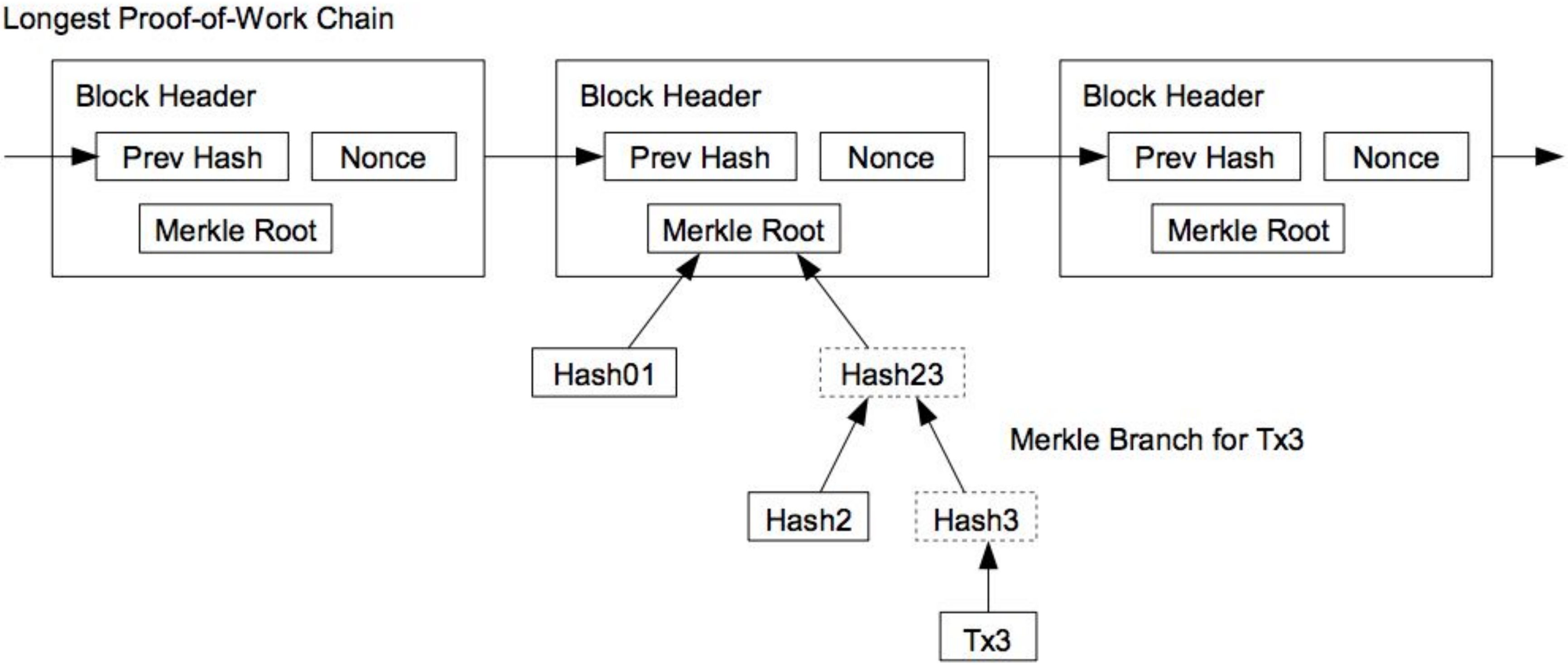


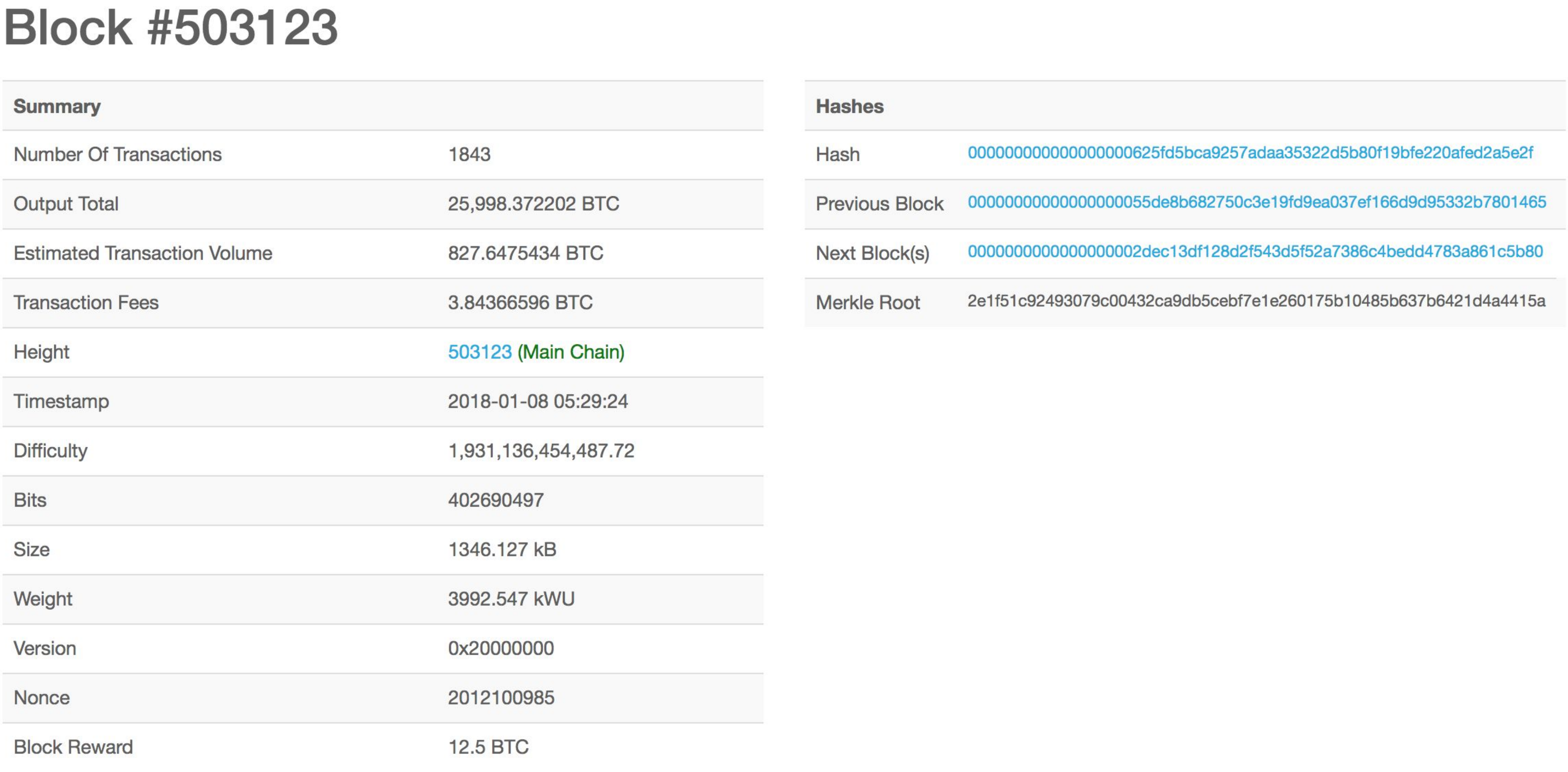
What are the components of a Bitcoin block?

- Nonce
  - 32-bit integer that is included in the block
  - The nonce will hash together with the previous block hash and merkle root to create the block's hash
  - A block hash is valid if it is less than a certain target, which usually means starting with a certain number of 0-bits (i.e., the hash must look like 0x000000023FB23..., **not** 0x12FD23A123...)
    - Small exceptions to this
  - If the block has the correct hash, we call that nonce the "winning" nonce
  - How hard is it to find a nonce?
  - Nonces will come up more next week when we talk about proof-of-work

How does Bitcoin use blocks to create a ledger?

- Each block is linked to the block immediately before it using the previous block hash
    - Allows for the creation of a chain of confirmed transactions, which we call the ledger
    - Allows storage of a shortened version of the blockchain
        - Why?
- What does Bitcoin's blockchain look like?


Longest Proof-of-Work Chain
Block Header
Prev Hash
Nonce
Merkle Root
Block Header
Prev Hash
Nonce
Merkle Root
Block Header
Prev Hash
Nonce
Merkle Root
Hash01
Hash23
Merkle Branch for Tx3
Hash2
Hash3
Tx3



Block #503123

Summary	
Number Of Transactions	1843
Output Total	25,998.372202 BTC
Estimated Transaction Volume	827.6475434 BTC
Transaction Fees	3.84366596 BTC
Height	503123 (Main Chain)
Timestamp	2018-01-08 05:29:24
Difficulty	1,931,136,454,487.72
Bits	402690497
Size	1346.127 kB
Weight	3992.547 kWU
Version	0x20000000
Nonce	2012100985
Block Reward	12.5 BTC

Hashes	
Hash	000000000000000000625fd5bca9257adaa35322d5b80f19bfe220afed2a5e2f
Previous Block	00000000000000000055de8b682750c3e19fd9ea037ef166d9d95332b7801465
Next Block(s)	0000000000000000002dec13df128d2f543d5f52a7386c4bedd4783a861c5b80
Merkle Root	2e1f51c92493079c00432ca9db5cebf7e1e260175b10485b637b6421d4a4415a

Blockchain Structure Summary

- A valid block consists of three main parts, the previous block’s hash, the list of transactions, and the ‘winning’ nonce
- Blocks, and the transactions that they contain, are verified and confirmed using the SHA-256 hashing algorithm
- These confirmed blocks are strung together in a deliberate structure to form a ledger known as the ‘blockchain’

Codelab 1

- Codelab 1 out tonight
- Due 11:59 PM on 02/16
- Project is to build a valid block based on a previous block hash and list of transactions
- You will need to build a merkle tree in order to do this!
- Valid block hashes will need 2 leading zeros, which can be achieved by using a nonce between 1 and 100,000 (in the case of this codelab)
- More details will be on the github page when we release the project

Questions?