BITCOIN

Daniel G.

February 7, 2023.



TOPICS

Part I: Data Structure

Part II: Bitcoin Overview

Part III: Bitcoin Mining

Part IV: Proof-of-X

PART I
DATA STRUCTURE

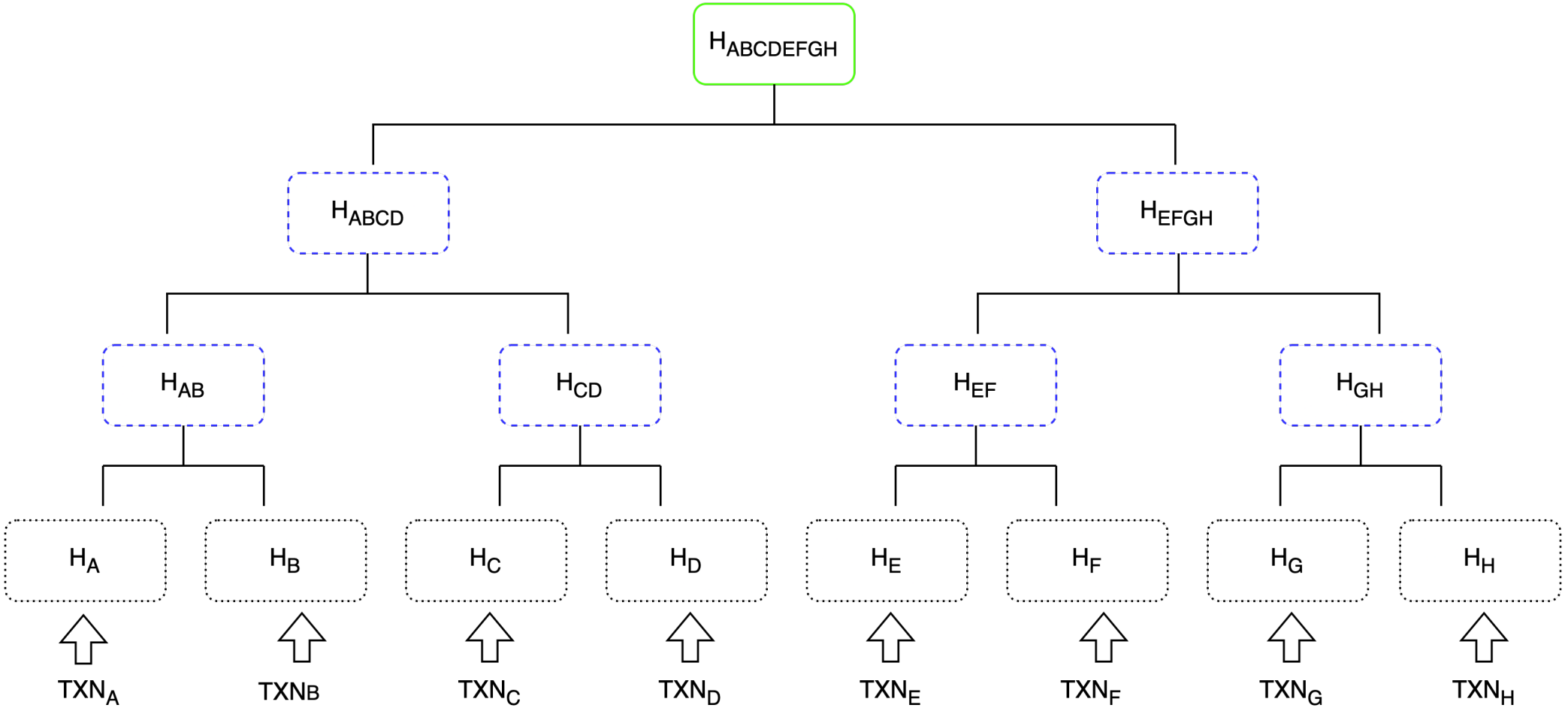
THE MERKLE TREE

- First introduced by Ralph Merkle in 1979, a Merkle tree (also known as Binary Hash Tree) is a tree data structure where each non-leaf node is a hash of its child nodes.
- Data elements are arranged into a balanced binary tree and the cryptographic hash (such as SHA) of an element with its sibling is computed.
- These hashes are then arranged in pairs and hashed together to create the next level of the tree. This continues until only one element remains at the top of the tree (called the Merkle root).
- It is primarily used to efficiently summarize and verify the integrity of large sets of data

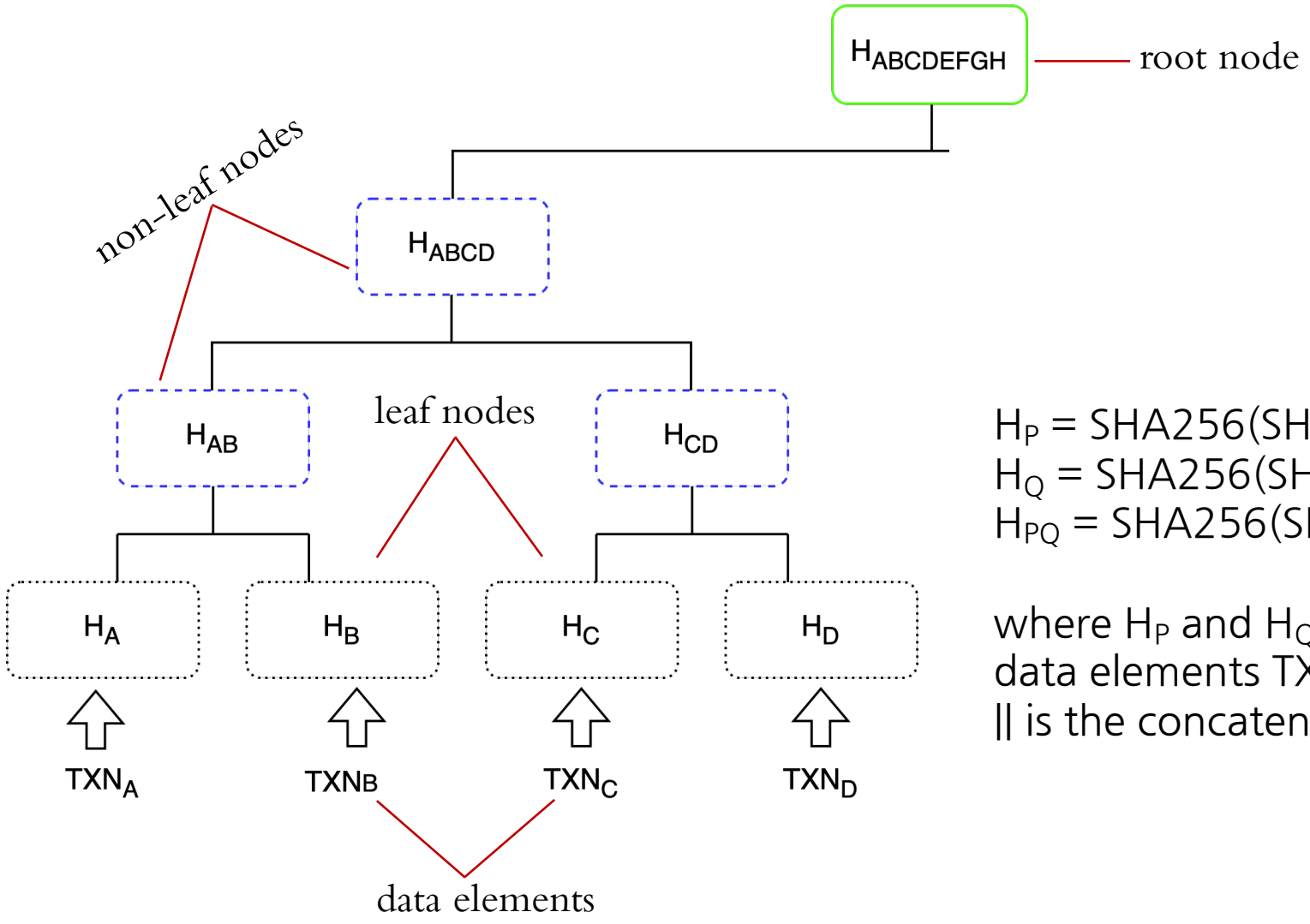
THE MERKLE TREE CONT...

- Cryptographic hash algorithms play a central role in the construction of a Merkle Tree.
- Bitcoin, for instance, uses a Secure Hash Algorithm (SHA-256) that has the following properties:
  - ❖ Collision resistance
  - ❖ Preimage resistance
  - ❖ Deterministic
  - ❖ Avalanche effect

THE MERKLE TREE CONT...



THE MERKLE TREE CONT...



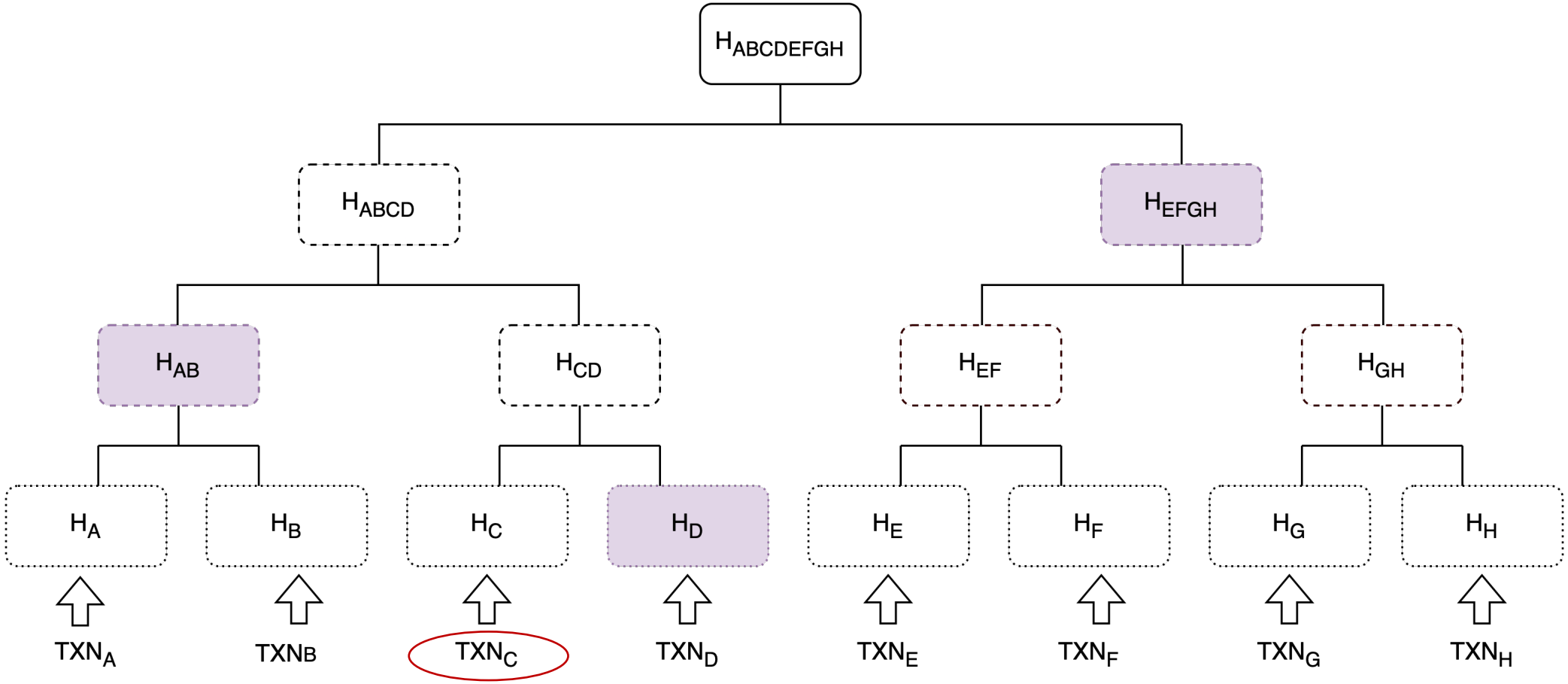
$H_P = SHA256(SHA256(TXN_P))$,
$H_Q = SHA256(SHA256(TXN_Q))$,
$H_{PQ} = SHA256(SHA256(H_P \| H_Q))$,

where H_P and H_Q are the hash of the P^{th} and Q^{th} data elements TXN_P and TXN_Q, resp.
$\|$ is the concatenation operator.

THE MERKLE TREE CONT...

- To prove whether a specific data element is included in a block, a node only needs to produce $\log_2^{(N)}$ 32-byte hashes, constituting an authentication Merkle path connecting the specific data element to the root of the tree.
- From the previous figure, a node, for instance, can prove that a data element C (TXN_C) is included in the block by producing a Merkle path that is only three 32-byte hashes long (96 bytes total).
- The path consists of HD, HAB and HEFGH.

THE MERKLE TREE CONT...



THE MERKLE TREE CONT...

- Merkle trees are used in bitcoin to summarize all the transactions in a block, producing an overall digital fingerprint of the entire set of transactions, providing a very efficient process to verify membership.
- Benefits of using Merkle tree for blockchain are:
  - ❖ Efficient mapping of arbitrarily large amounts of data and easy identification of changes should they occur.
  - ❖ Immutability of transactions in a block.
  - ❖ Significant improvement of storage and processing
- Merkle trees can also be used for other applications such as in certificate transparency and file integrity protection.

THE MERKLE TREE CONT...

Number of Transactions	*Approx. Block Size (KB)*	*Path Size (Hashes)*	*Path Size (Bytes)*
16	4	4	128
32	8	5	160
64	16	6	192
128	32	7	224
256	64	8	256
512	128	9	288
1024	256	10	320
2048	512	11	352
4096	1024 = (1MB)	12	384

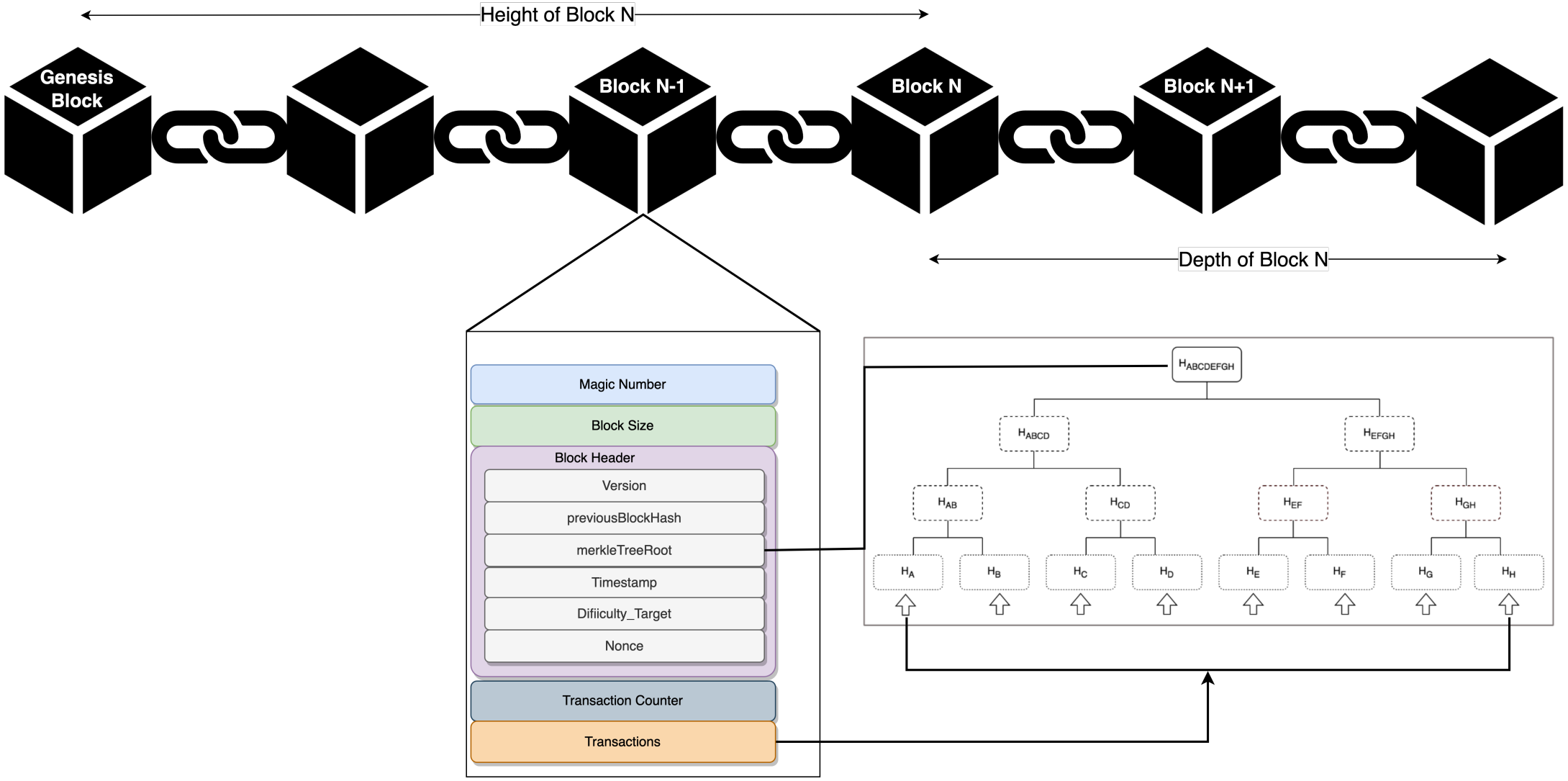
An illustration of Merkle Tree efficiency

PART II
OVERVIEW OF BITCOIN

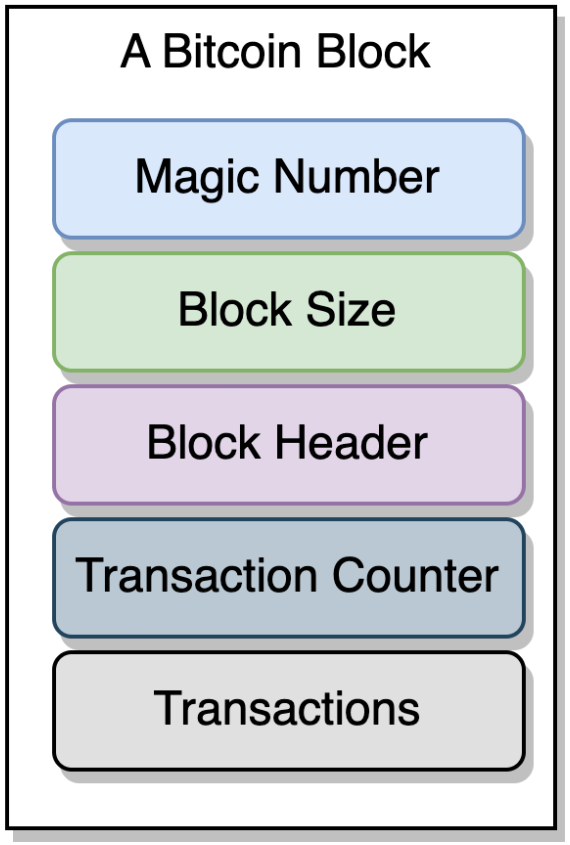
BITCOIN: OVERVIEW

- Bitcoin is a decentralized digital currency that allows peer-to-peer transactions without the need for intermediaries.
- The first Bitcoin specification was published by Satoshi Nakamoto in 2009.
- Bitcoin is composed of three core elements:
  - ***Network***: composed of nodes that operate on peer-to-peer system, where each node has a copy of the ledger and works together to validate transactions and add them to the blockchain.
  - ***Ledger***: a decentralized, tamper-proof database that contains a record of all Bitcoin transactions in a block that hold a reference to the previous block, creating a chain of blocks.
  - ***Cryptocurrency***: a digital currency, called bitcoin, that operates on a public ledger, in which anyone can see the transactions made.

BITCOIN: OVERVIEW CONT...

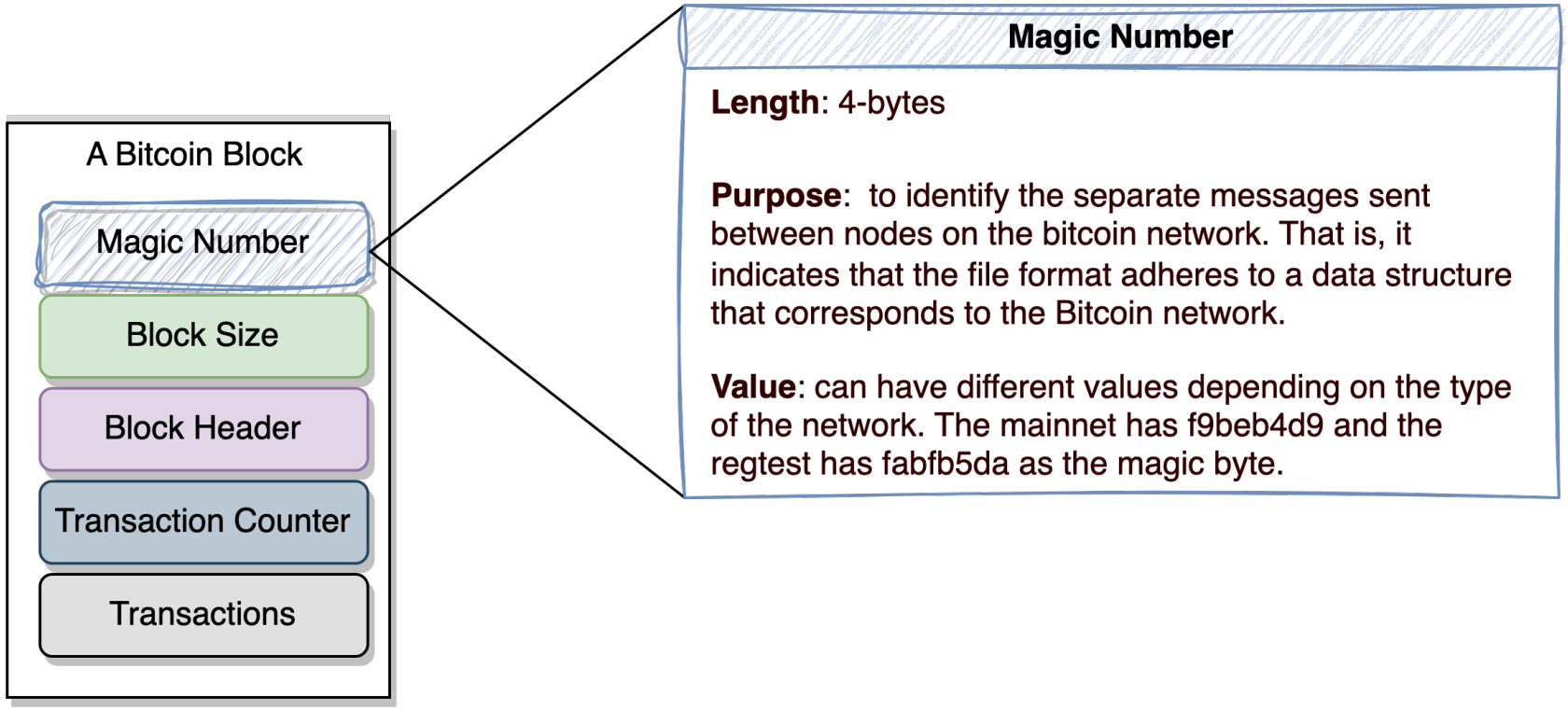


BITCOIN: BLOCK STRUCTURE

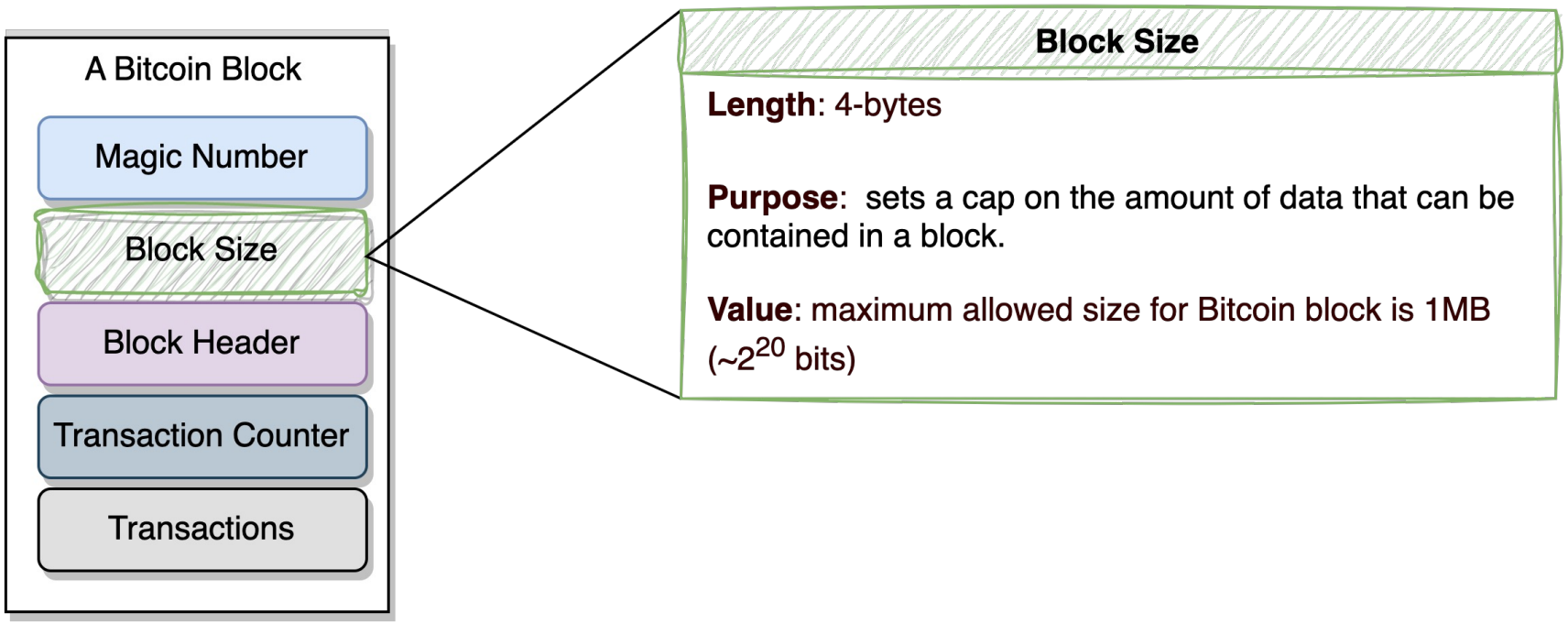


- A Bitcoin blocks are chained together (where the current block adds the hash of the previous block) to form a blockchain.
- The Bitcoin block is constructed from five elements: magic number, block size, block header, transaction counter and transactions.
- A block is primarily identified by the hash of the block header, which consists of a reference to a previous block hash, information related to mining, and the Merkle Tree root.

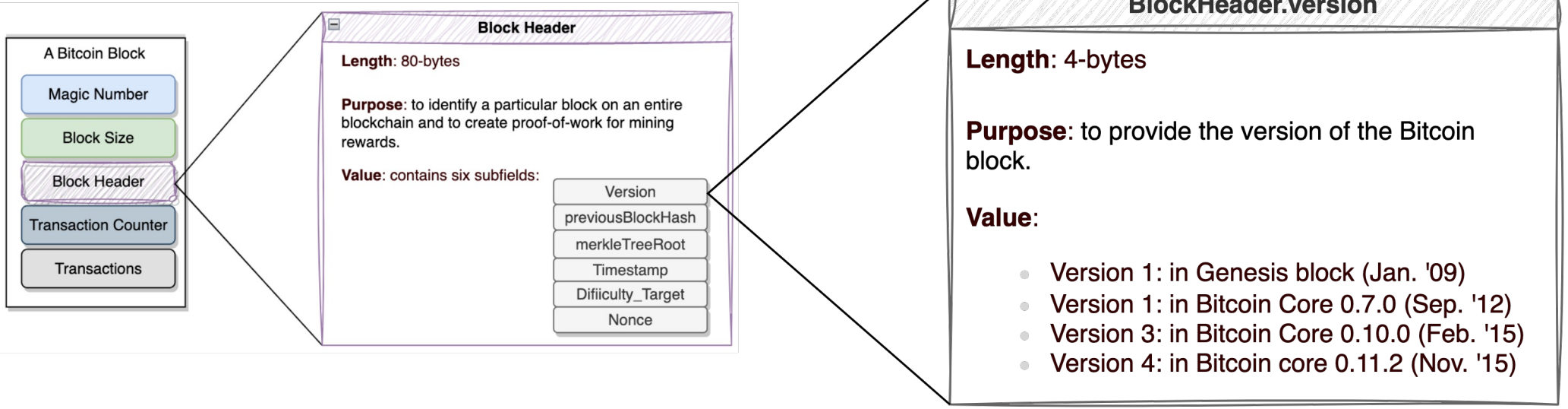
BITCOIN: BLOCK STRUCTURE CONT...



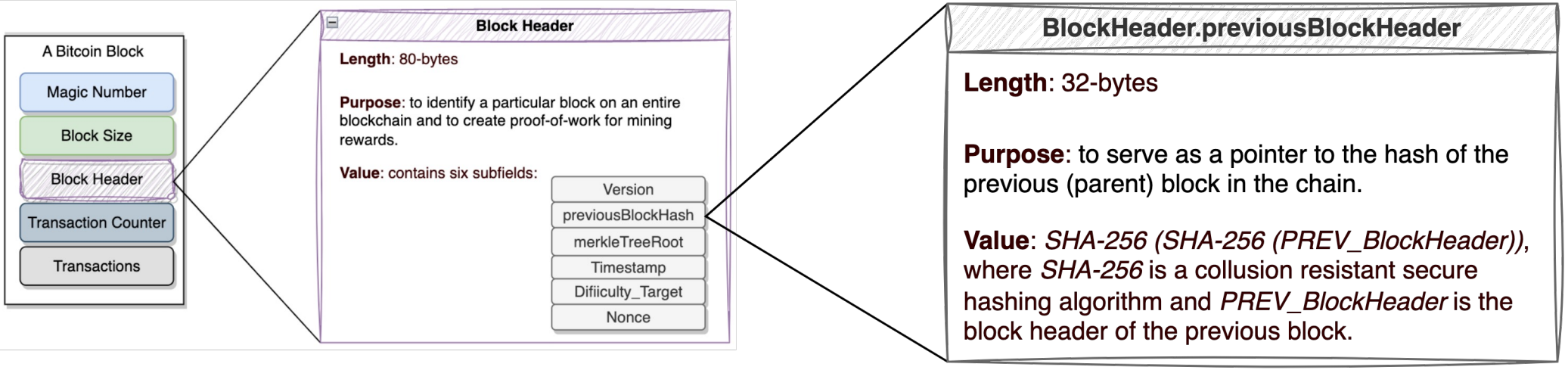
BITCOIN: BLOCK STRUCTURE CONT...



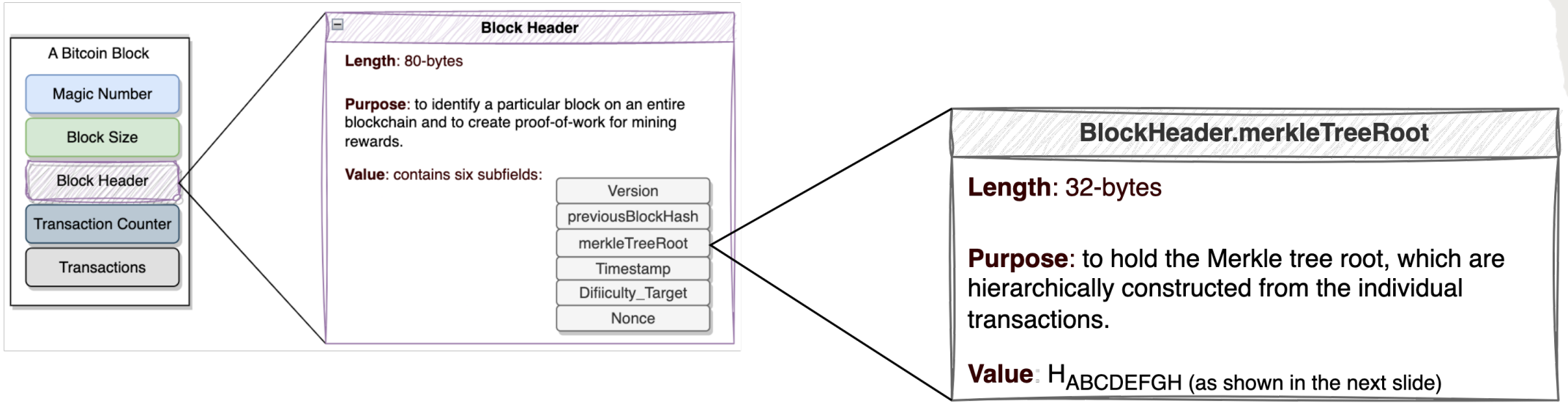
BITCOIN: BLOCK STRUCTURE CONT...



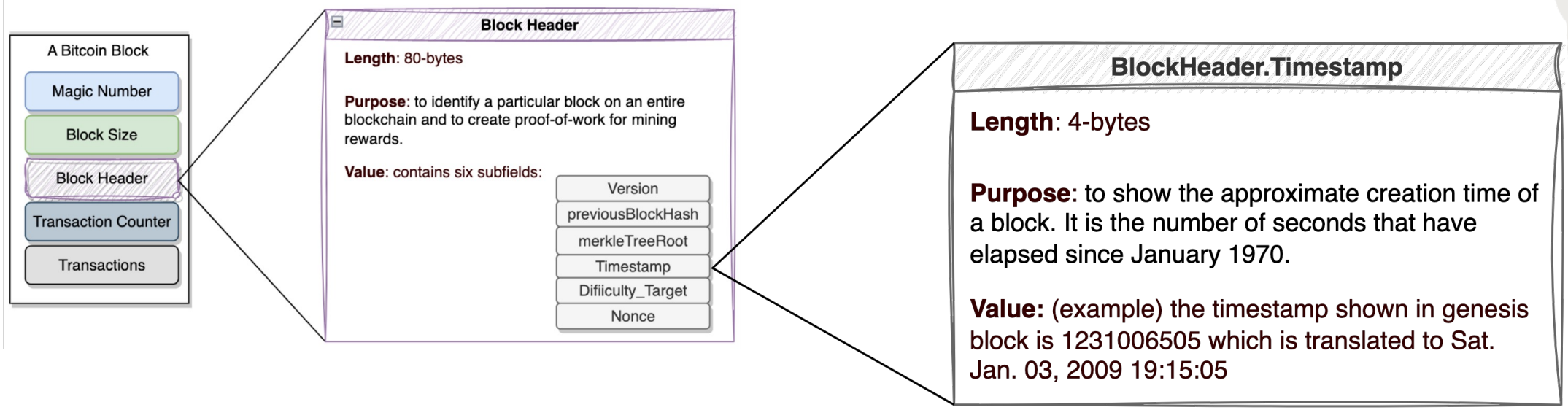
BITCOIN: BLOCK STRUCTURE CONT...



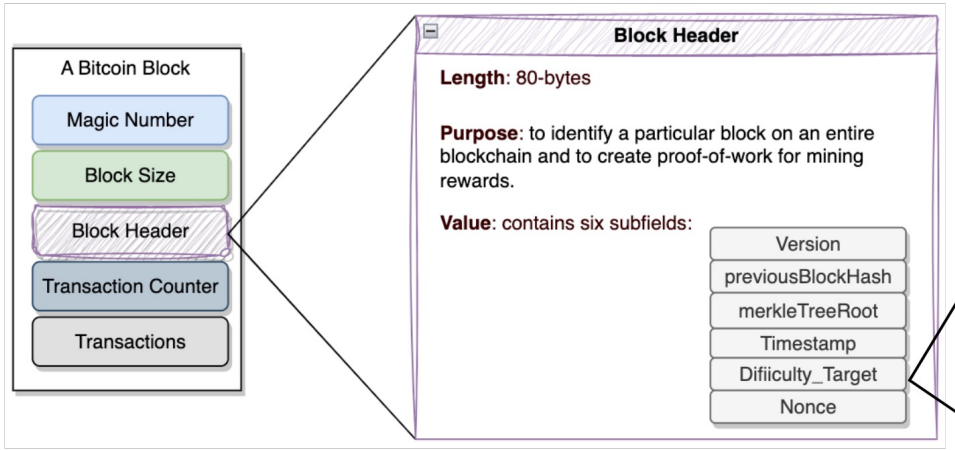
BITCOIN: BLOCK STRUCTURE CONT...



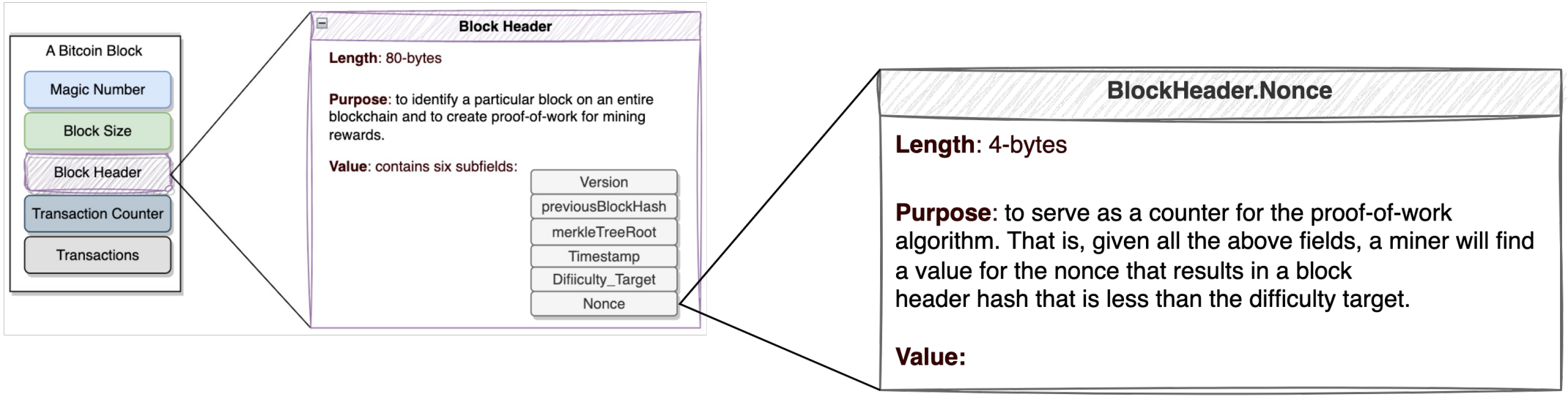
BITCOIN: BLOCK STRUCTURE CONT...



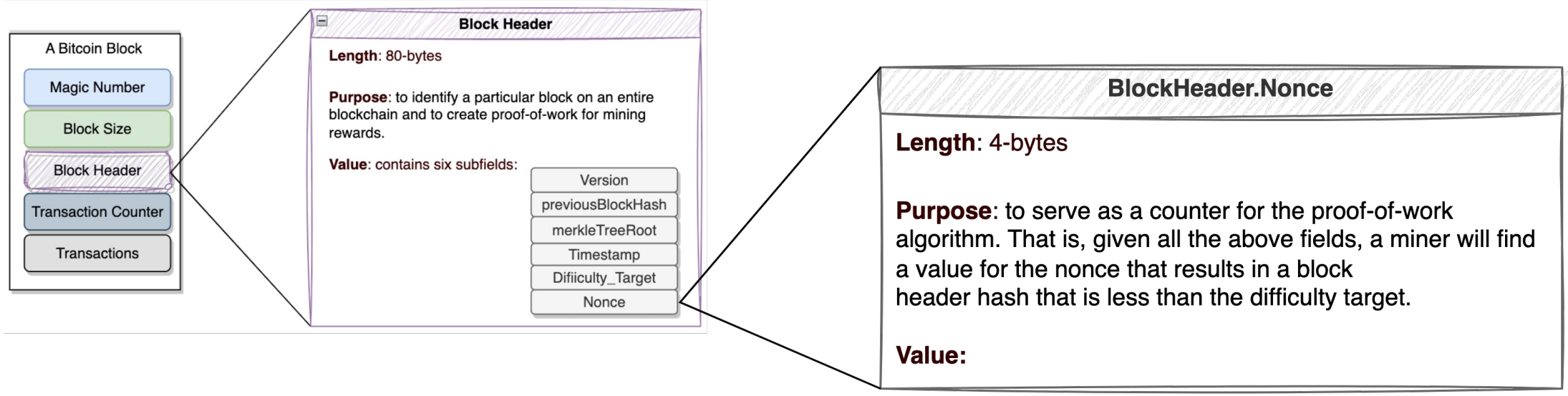
BITCOIN: BLOCK STRUCTURE CONT...



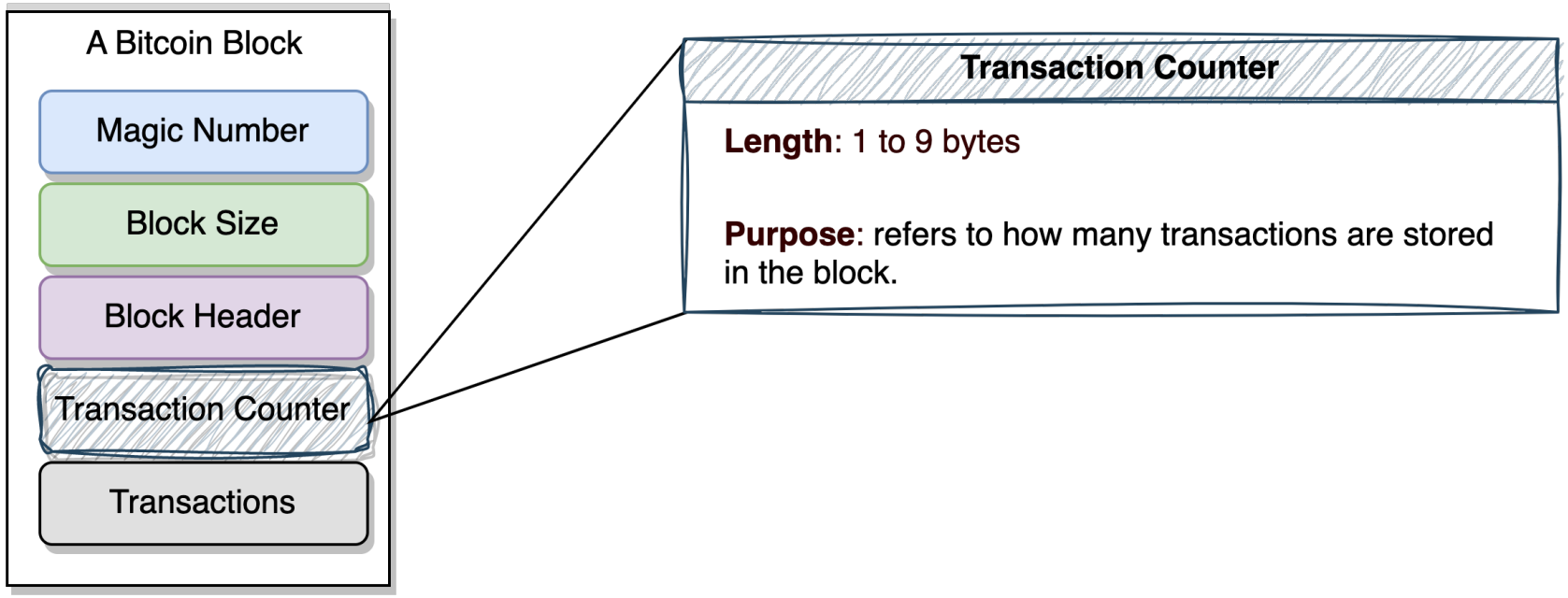
BITCOIN: BLOCK STRUCTURE CONT...



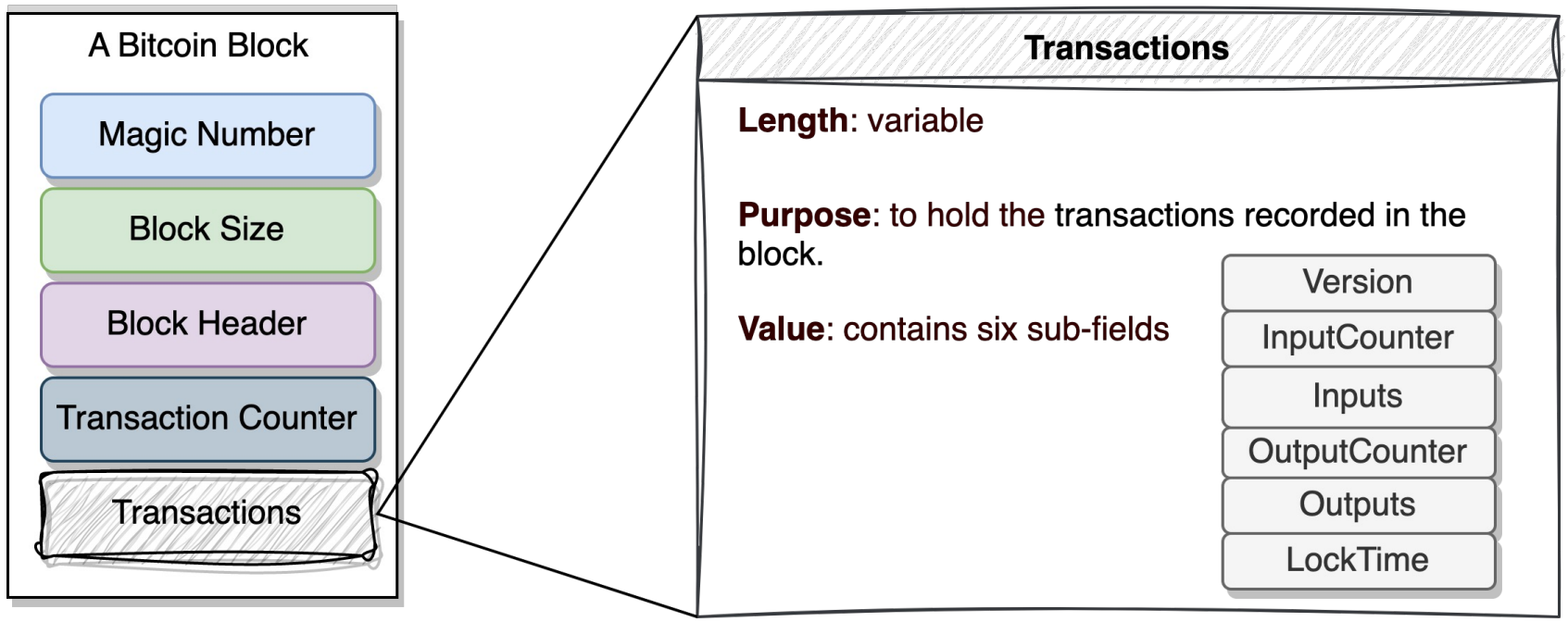
BITCOIN: BLOCK STRUCTURE CONT...



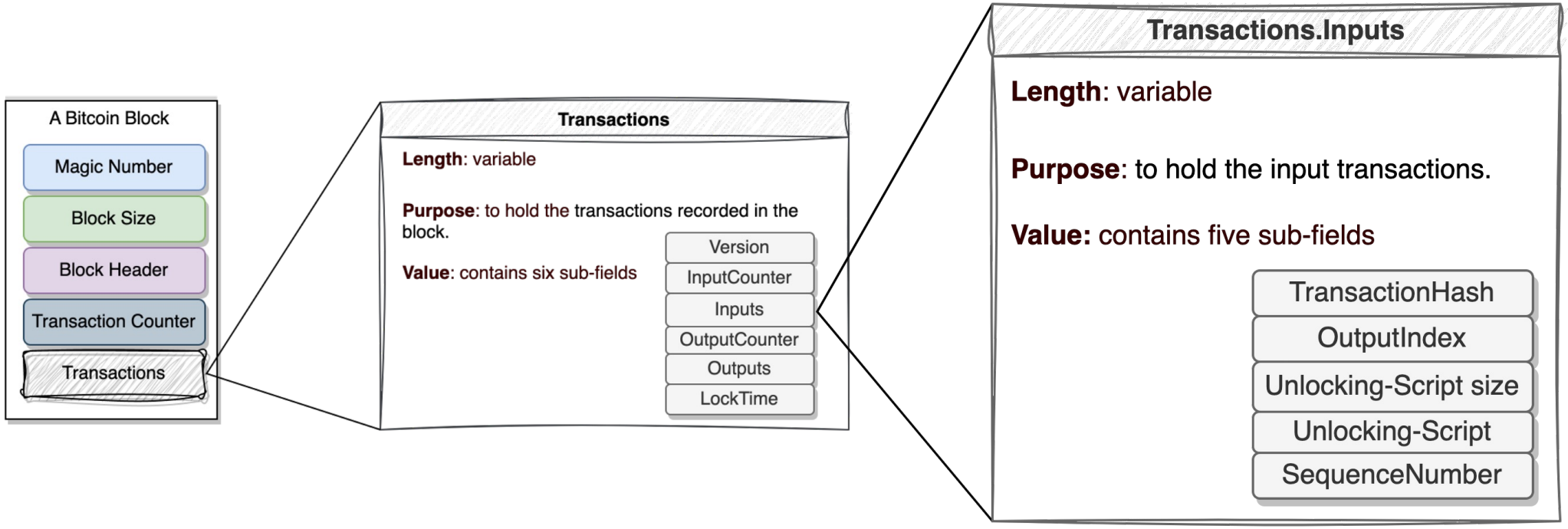
BITCOIN: BLOCK STRUCTURE CONT...



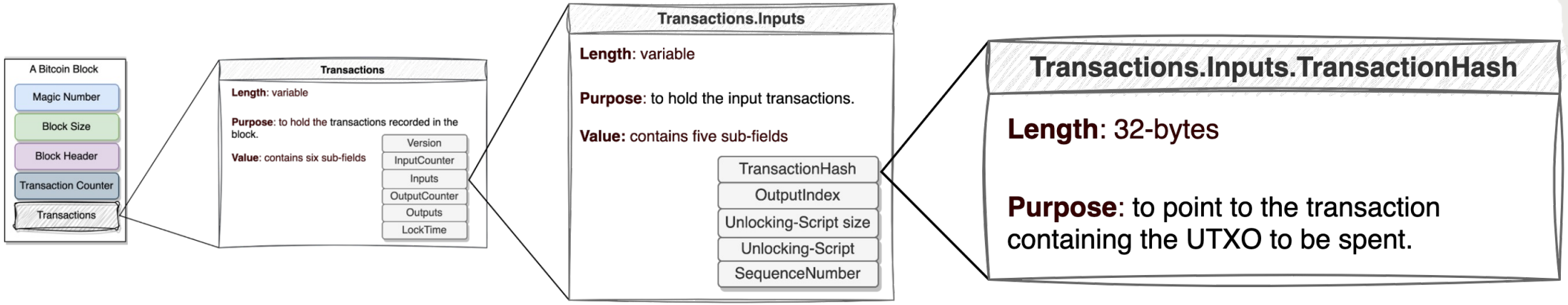
BITCOIN: BLOCK STRUCTURE CONT...



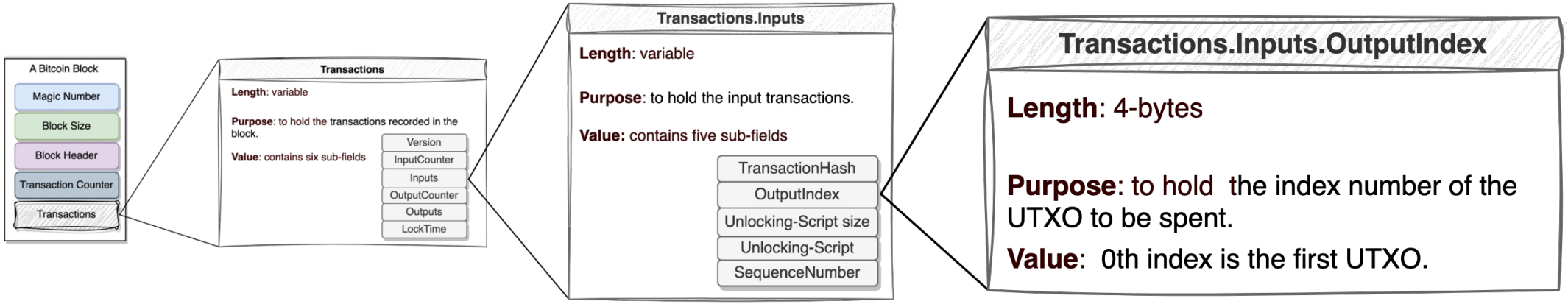
BITCOIN: BLOCK STRUCTURE CONT...



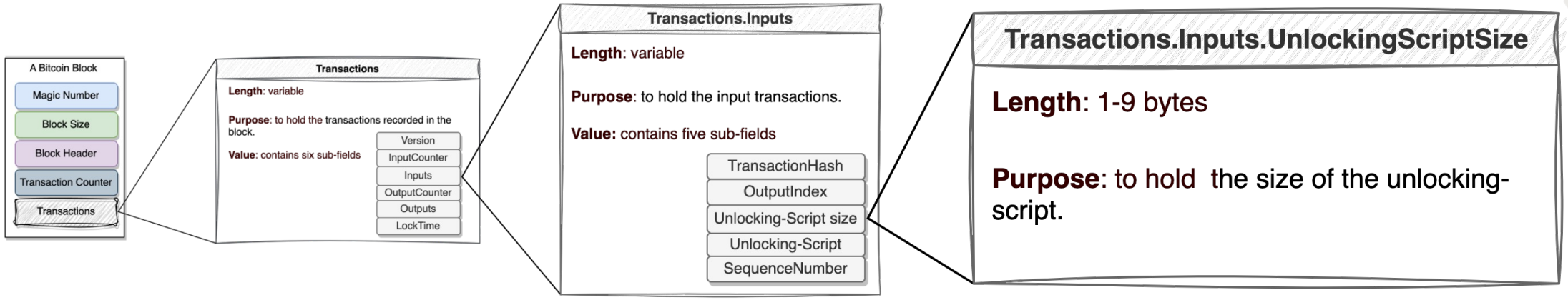
BITCOIN: BLOCK STRUCTURE CONT...



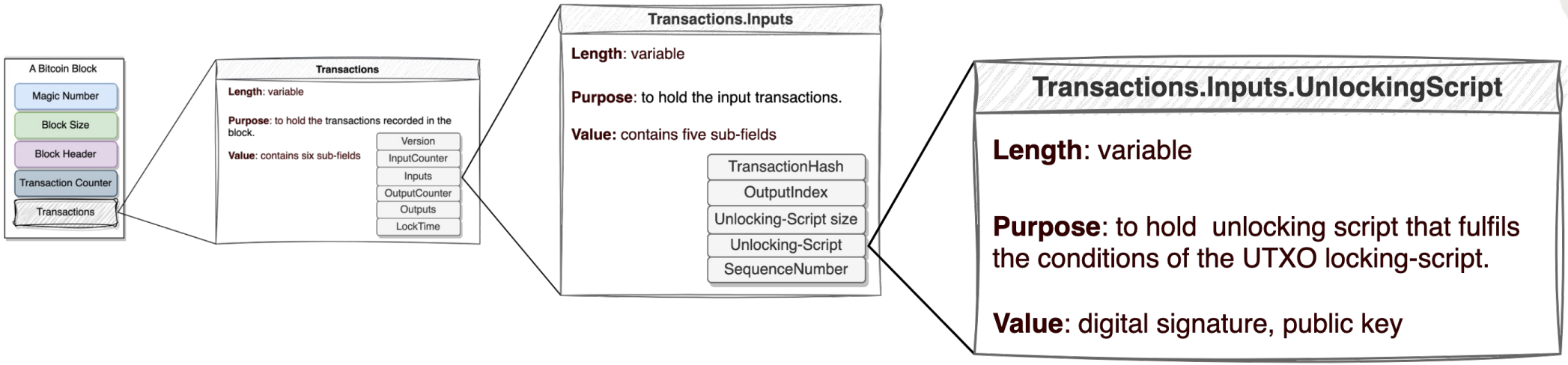
BITCOIN: BLOCK STRUCTURE CONT...



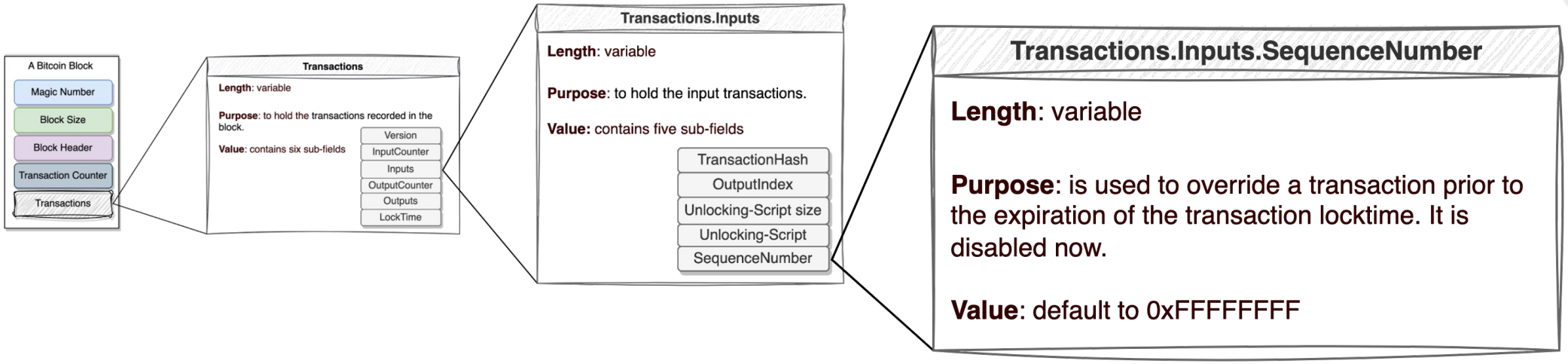
BITCOIN: BLOCK STRUCTURE CONT...



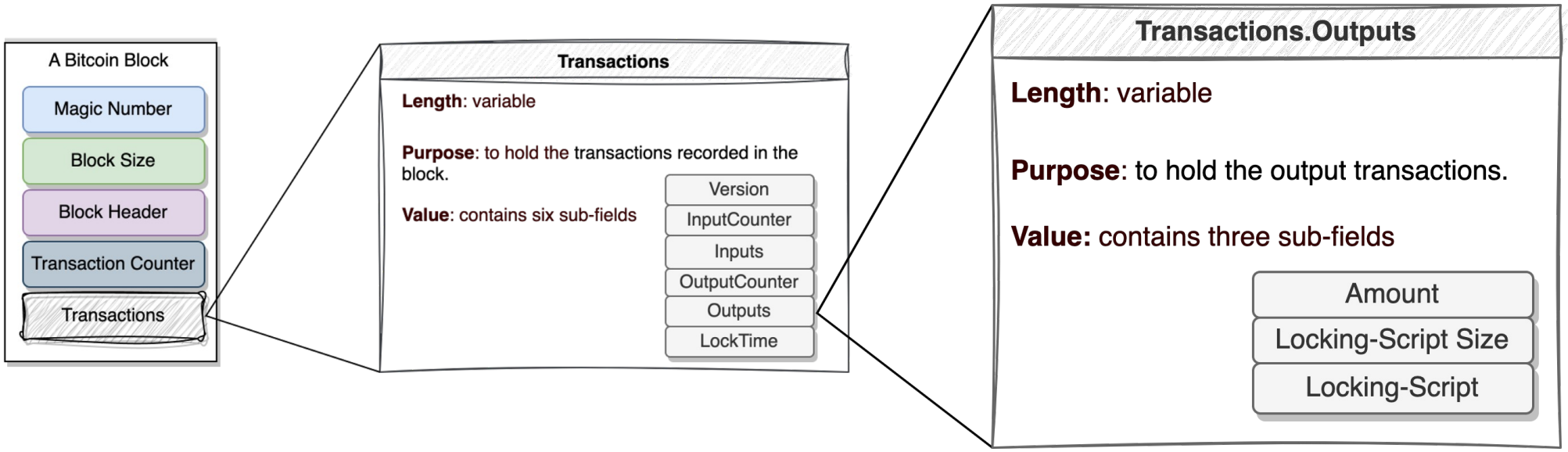
BITCOIN: BLOCK STRUCTURE CONT...



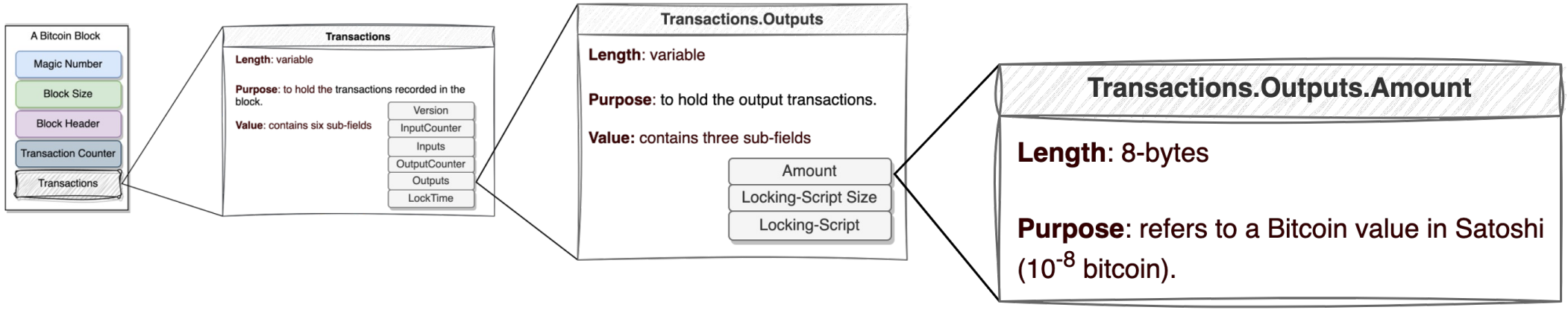
BITCOIN: BLOCK STRUCTURE CONT...



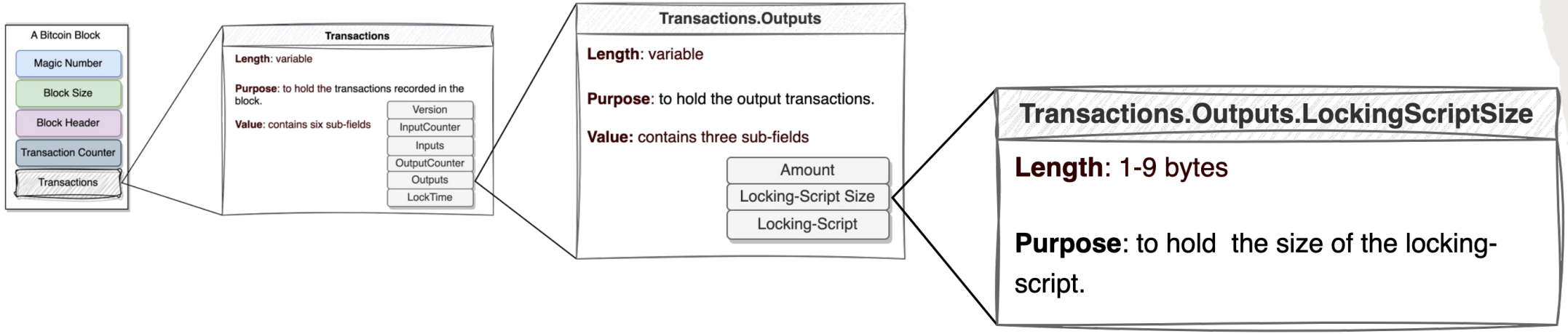
BITCOIN: BLOCK STRUCTURE CONT...



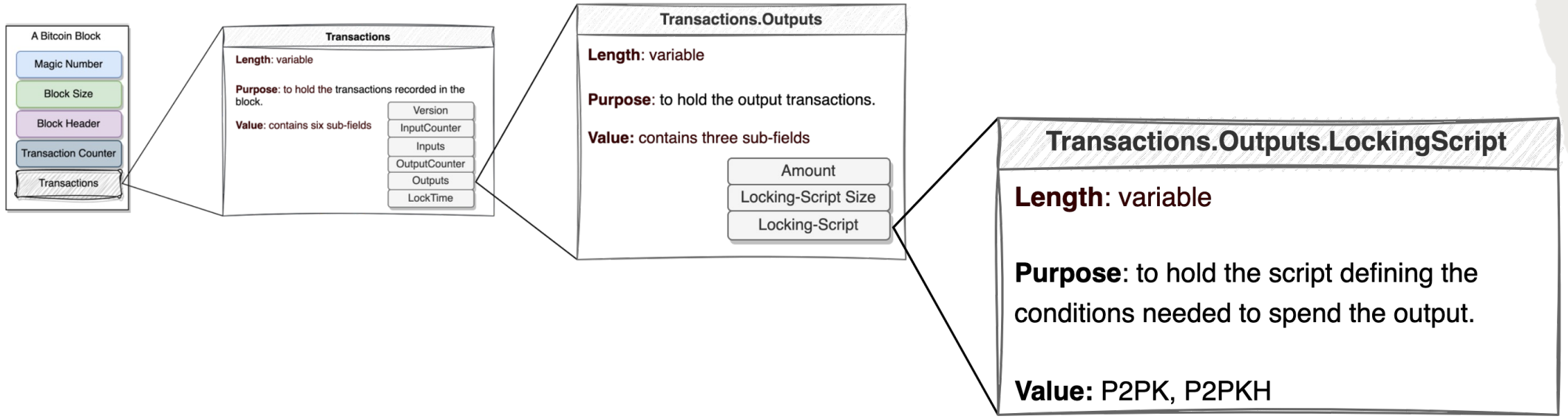
BITCOIN: BLOCK STRUCTURE CONT...



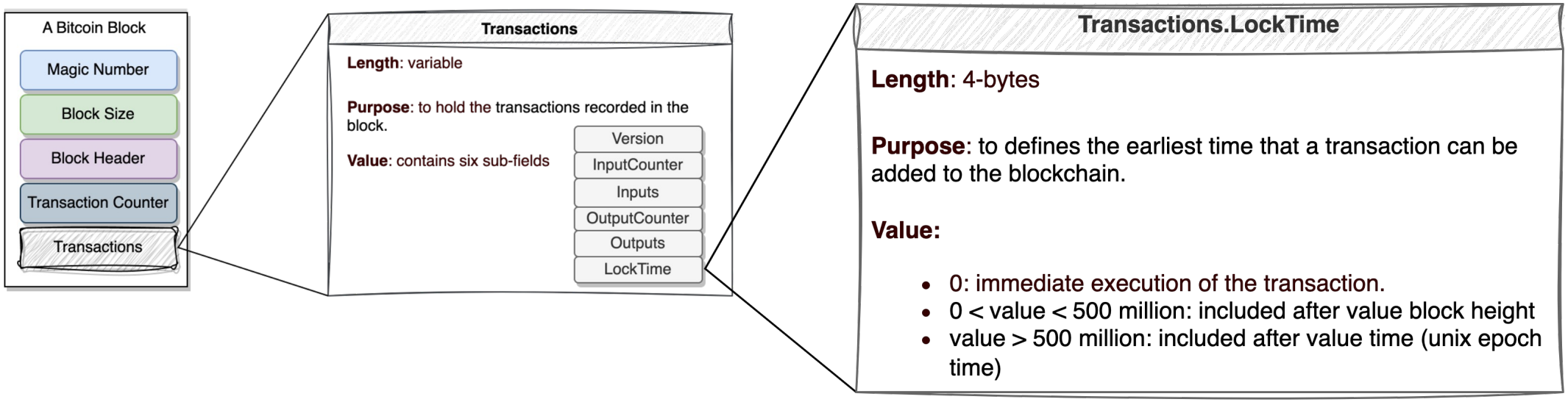
BITCOIN: BLOCK STRUCTURE CONT...



BITCOIN: BLOCK STRUCTURE CONT...



BITCOIN: BLOCK STRUCTURE CONT...



BITCOIN: STORAGE

- Blocks are stored in files *on disk*, and to preserve immutability, each block is linked to another forming a chain.
- Block indexes (in a key-value store called LevelDB) are stored *on disk* to facilitate search for blocks and transactions.
- Moreover, state-related information (like Unspent Transaction Output – UTXO) are stored *on disk* in a LevelDB storage called chainstate.
- Merkle trees are constructed and stored *in memory* (on demand) by a Bitcoin node.
- On a start-up, the entire block indexes in the LevelDB database are loaded *in memory* for efficiency.
- A UTXO set, that stores all unspent outputs, is accessed frequently during block validation. To reduce this access time, the required unspent coins are loaded in memory cache.

BITCOIN: TRANSACTIONS

- Transactions are data structures that encode the transfer of value between participants in the Bitcoin system.
- Transactions are the most important part of the bitcoin system – everything else in Bitcoin is designed to ensure that transactions can be created, propagated, validated, and finally added to the global ledger.
- As shown in the previous slides, a Bitcoin transaction contains version, input counter, input, output counter, output and lock time.
- The input and output elements are composed of other data structures like transaction hash, amount, locking and unlocking scripts, etc.

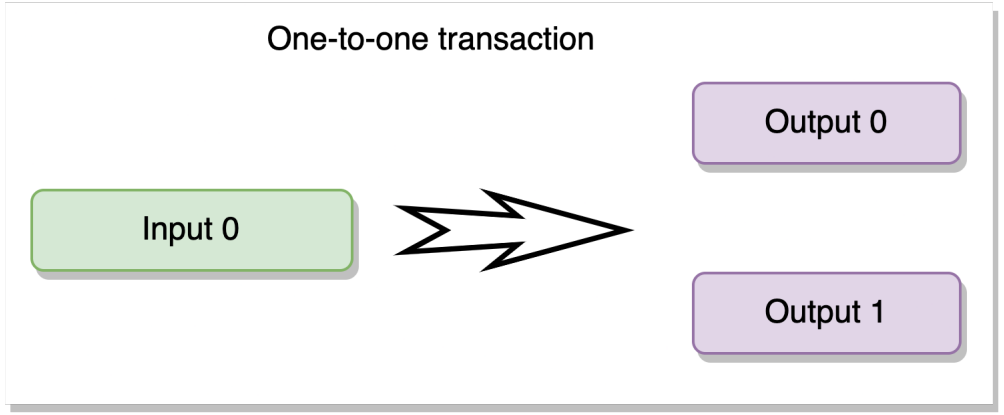
BITCOIN: TRANSACTIONS. CONT...

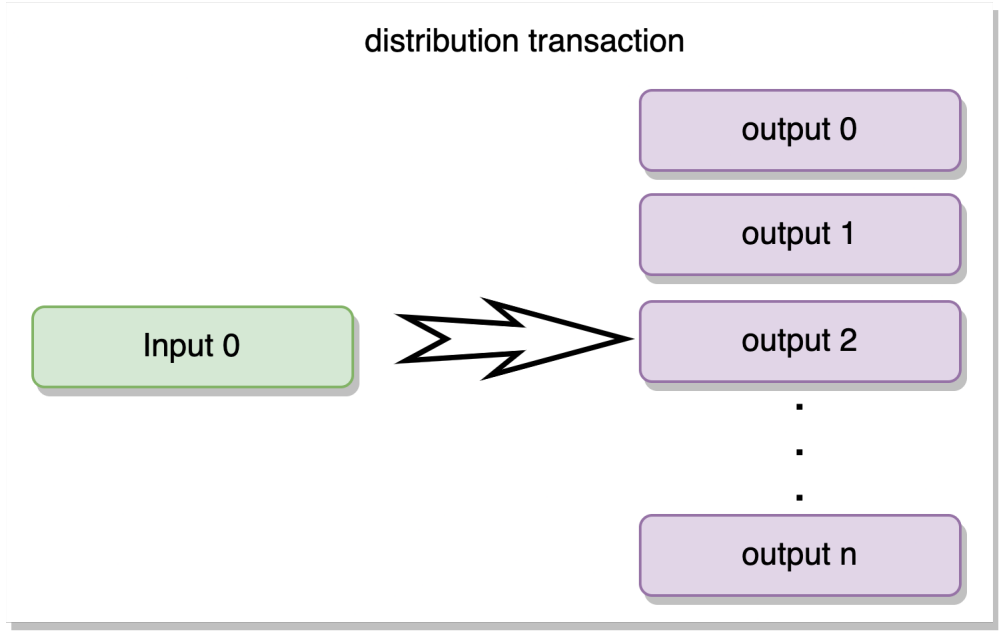
- The fundamental building block of a Bitcoin transaction is unspent transaction output (UTXO).
- UTXO are indivisible chunks of bitcoin currency locked to a specific owner.
- Whenever a user receives bitcoin, that amount is recorded within the blockchain as a UTXO.
- A user's bitcoin is normally scattered as UTXO amongst numerous transactions in different blocks.
- There are no accounts or balances in bitcoin, there are only unspent transaction outputs (UTXO) scattered in the blockchain.

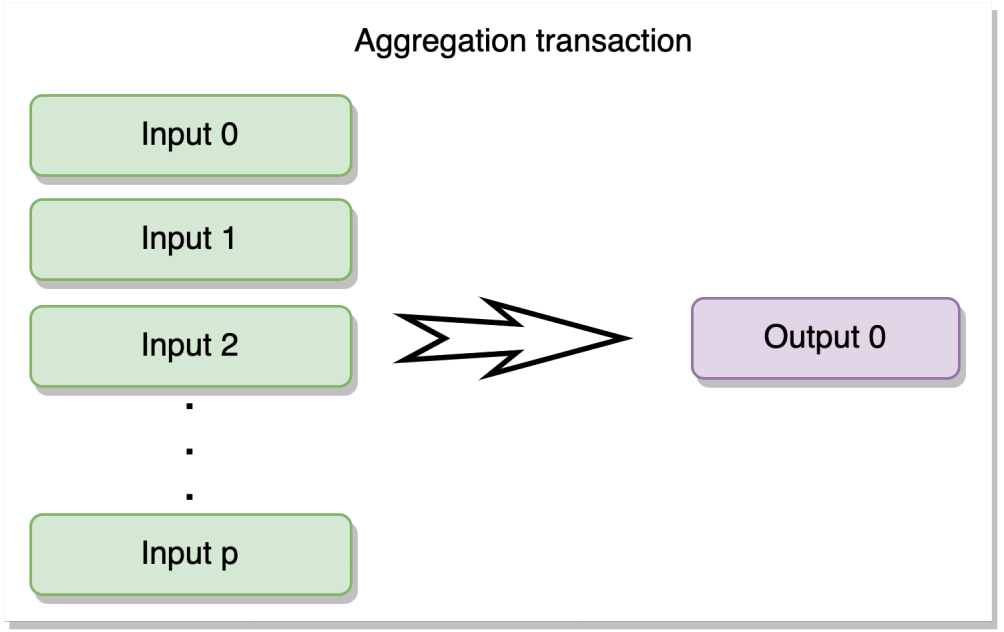
BITCOIN: TRANSACTIONS. CONT...

- A compilation of all the UTXOs at any given time is known as the *UTXO set*, which is constantly maintained by each Bitcoin (full) node.
- The UTXO consumed by a transaction are called *transaction inputs*, while the UTXO created by a transaction are called *transaction outputs*.
- Transactions consume UTXO *unlocking* it with the signature of the current owner and create UTXO *locking* it to the bitcoin address of the new owner.
- If a UTXO is larger than the desired value of a transaction, it must still be consumed in its entirety and change must be generated in the transaction.
- The exception to the output and input chain is a special type of transaction called the coinbase transaction, which has no input but only output.

BITCOIN: TRANSACTIONS. CONT...







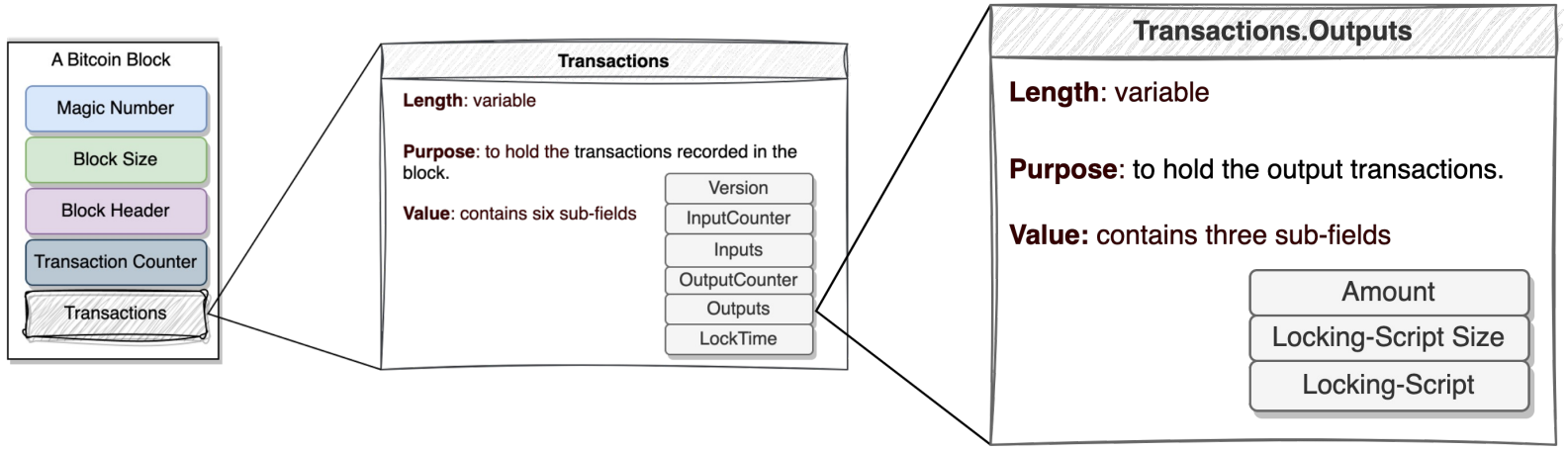
BITCOIN: TRANSACTIONS. CONT...

- The UTXO approach, compared to account-based approach, has some advantages concerning privacy and scalability.
- Unlike account-based approach that maintains the user's balance and keeps explicit association between the transaction and accounts, states cannot be linked to an individual user in the UTXO approach.
- Since it is possible to process multiple UTXOs at the same time, it enables parallel transactions and encourages scalability innovation.
- However, the UTXO approach is more complex and less efficient especially with regard to computation and storage.

BITCOIN: TRANSACTIONS. CONT...

Transaction Output

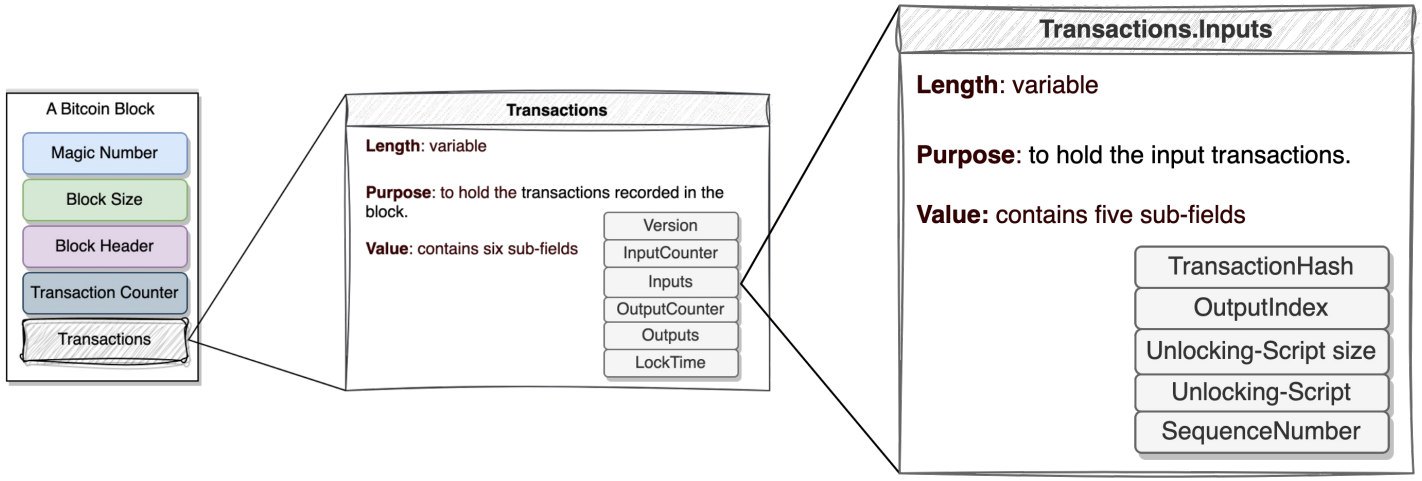
- Consists of two parts: amount and locking script (scriptPubKey)
- Transaction outputs associate a specific amount (in satoshis) to a specific locking-script that defines the condition that must be met to spend that amount.
- Usually the locking script locks the output to a specific bitcoin address, thereby transferring ownership of that amount to the new owner.



BITCOIN: TRANSACTIONS. CONT...

Transaction Input

- Consists of transaction hash, unlocking script (sigPubKey), and sequence number.
- Transaction inputs are basically the pointers to UTXO.
- The unlocking script is usually a signature proving ownership of the bitcoin address that is in the locking script.



BITCOIN: TRANSACTIONS. CONT...

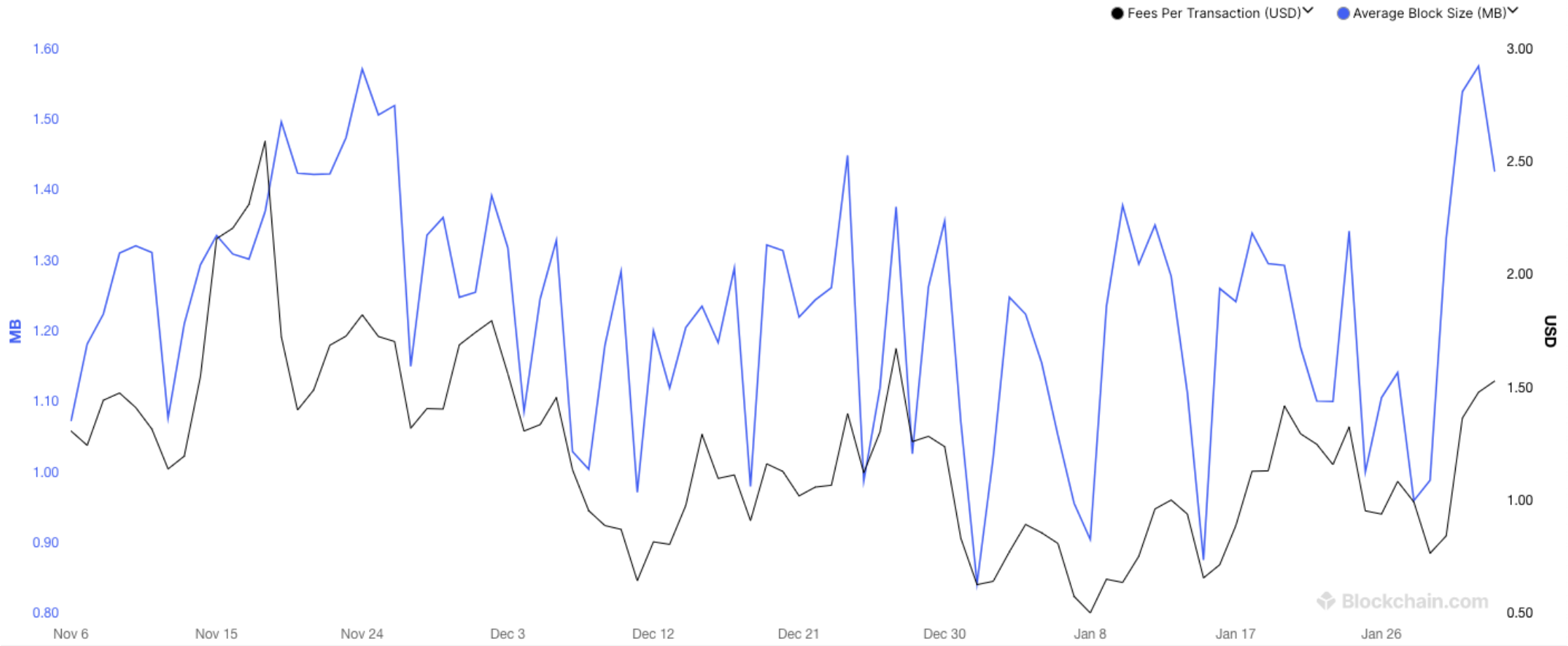
Transaction Fees

- Transaction fees play two important roles: incentive for the miners to include a transaction in the next block and disincentive against malicious transactions.
- Transaction fees are calculated based on the size of the transaction, not the value of the transaction in bitcoin. Overall, transaction fees are set based on market forces, network capacity and transaction volume.
- The data structure of transactions does not have a field for transaction fees, hence they are implied as the excess of inputs minus outputs:

$$Transaction\ fee = Sum(Inputs) - Sum(Outputs)$$

BITCOIN: TRANSACTIONS. CONT...

Transaction Fee Vs. Average Block Size



BITCOIN: TRANSACTIONS. CONT...

Transaction Scripts

- Bitcoin clients validate transactions by executing scripts: locking script (also called scriptPubKey) placed on a UTXO and unlocking script (also called scriptSig) that usually contains a signature.
- Transactions in Bitcoin are validated by executing the locking and unlocking scripts together.
- For each input in the transaction, the validation software will first retrieve the UTXO referenced by the input. That UTXO contains a locking script defining the conditions required to spend it.
- The validation software will then take the unlocking script contained in the input that is attempting to spend this UTXO and execute the two scripts.

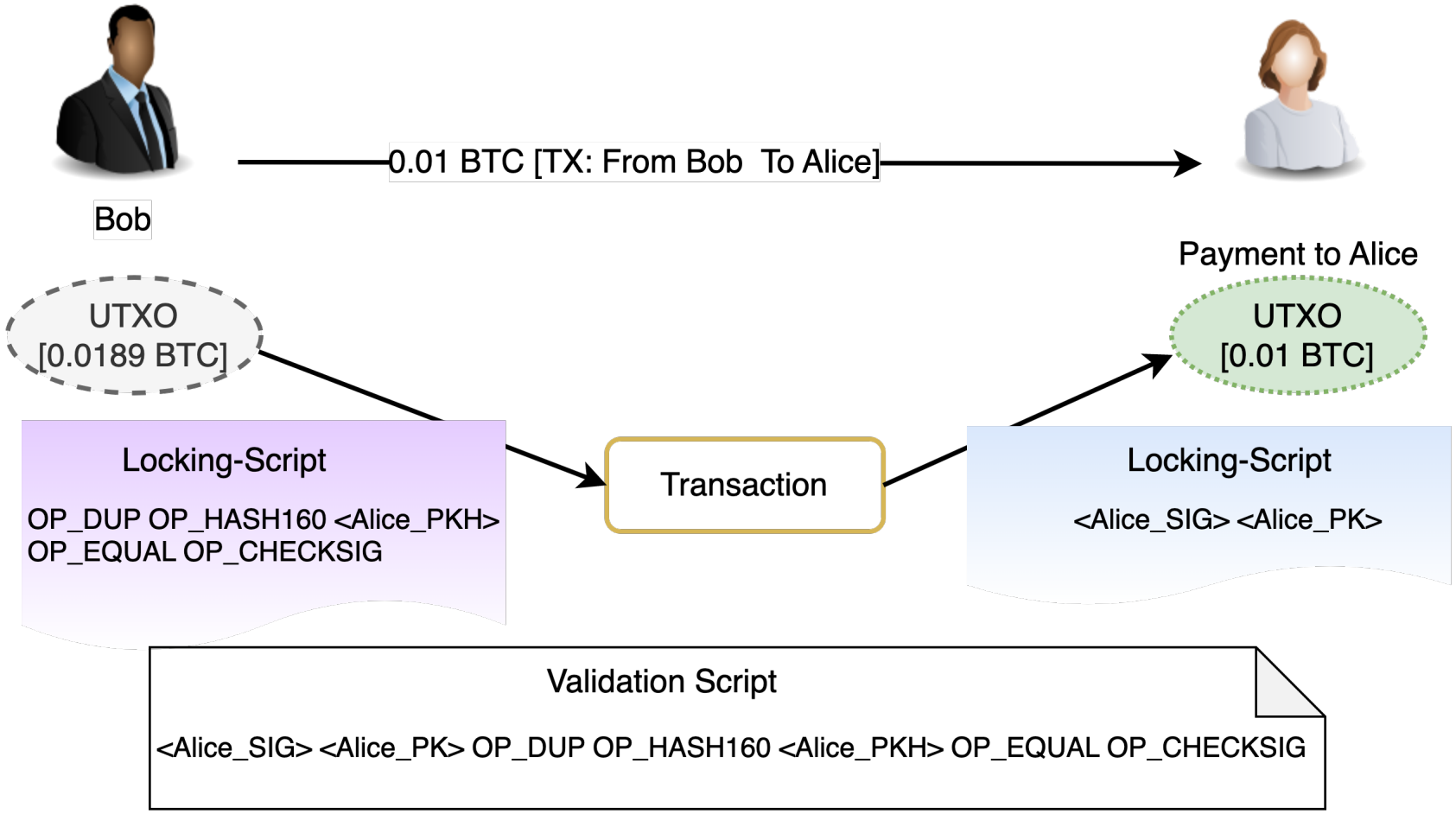
BITCOIN: TRANSACTIONS. CONT...

Transaction Scripts (cont...)

- Most transactions processed through the Bitcoin network are based on a script called Pay-to-Public-Key-Hash (P2PKH). Such scripts are of the form "Alice pays Bob".
- Bitcoin transaction are not limited to P2PKH scripts only. There are four more standard types of transaction scripts: Pay-to-Public-Key (P2PK), Multi-Signature, Pay-to-Script-Hash (P2SH) and Data Output (OP_RETURN).

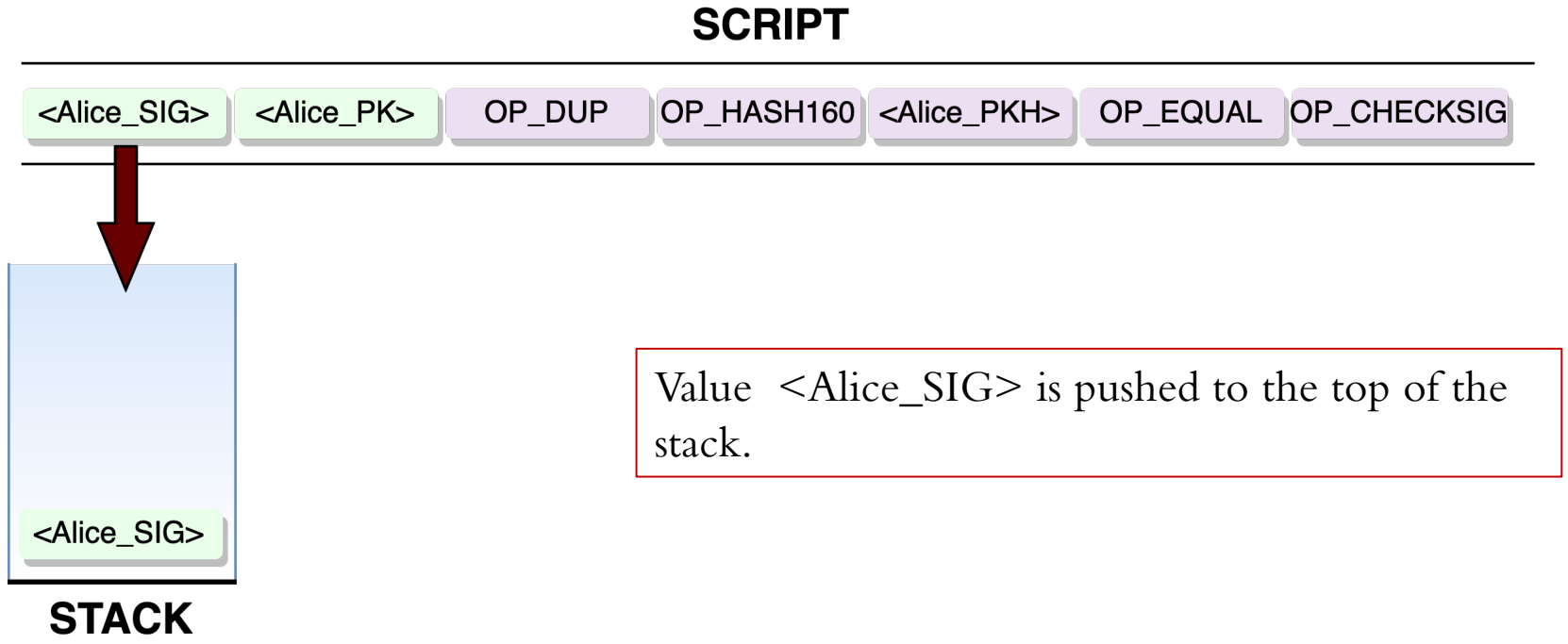
Transaction Scripts (cont...)

- ***P2PKH***



Transaction Scripts (cont...)

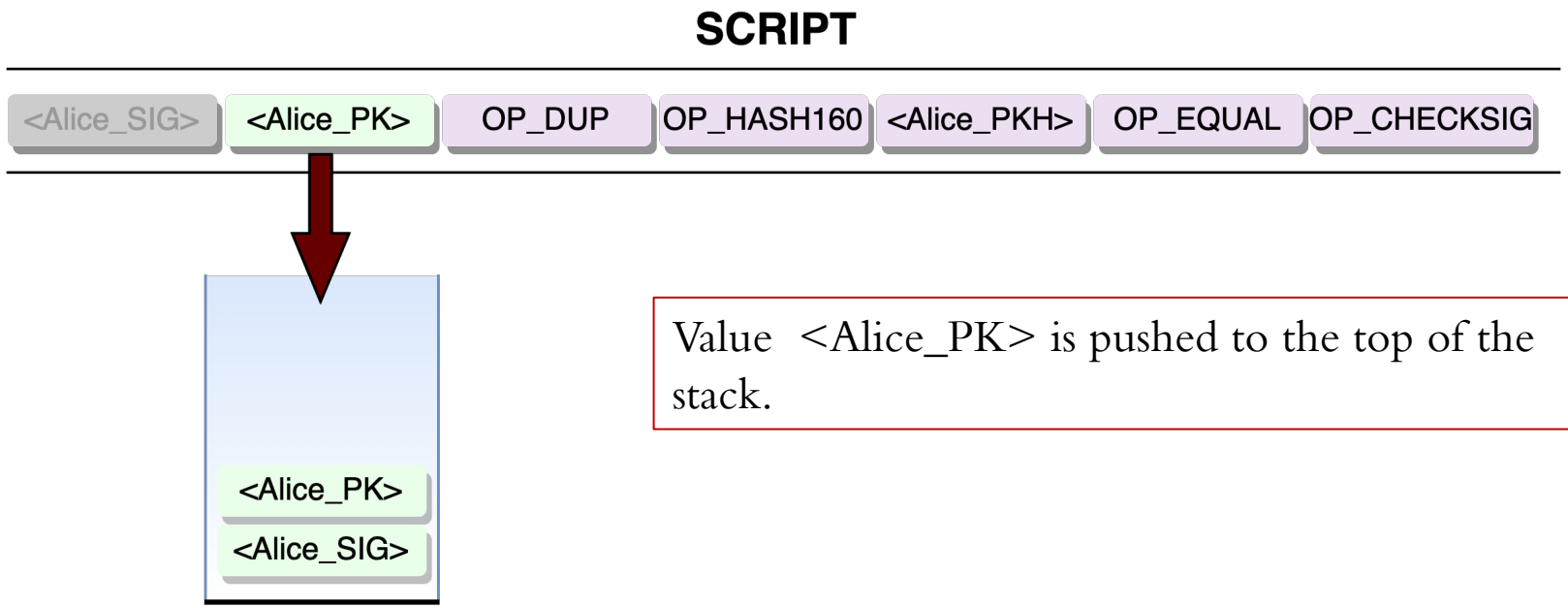
- ***P2PKH***



Value <Alice_SIG> is pushed to the top of the stack.

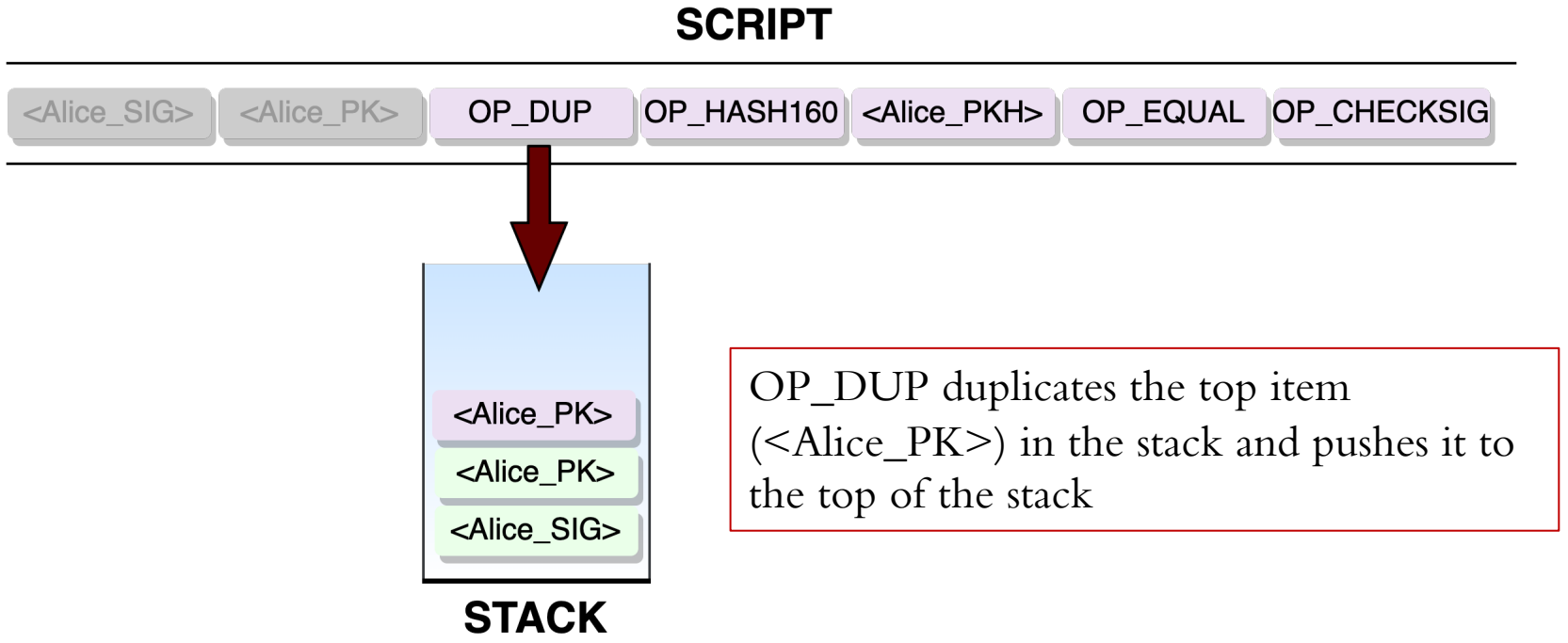
Transaction Scripts (cont...)

- ***P2PKH***



Transaction Scripts (cont...)

- ***P2PKH***

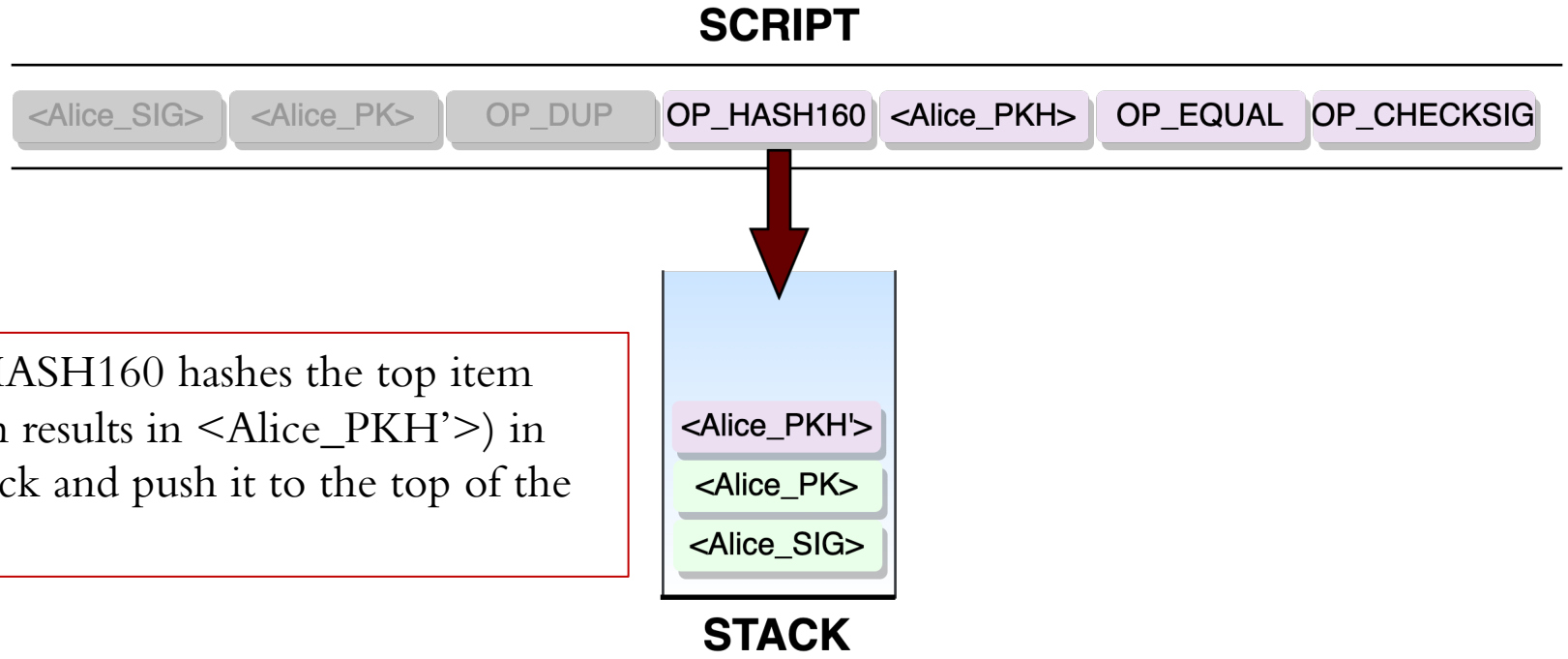


OP_DUP duplicates the top item (<Alice_PK>) in the stack and pushes it to the top of the stack

BITCOIN: TRANSACTIONS. CONT...

Transaction Scripts (cont...)

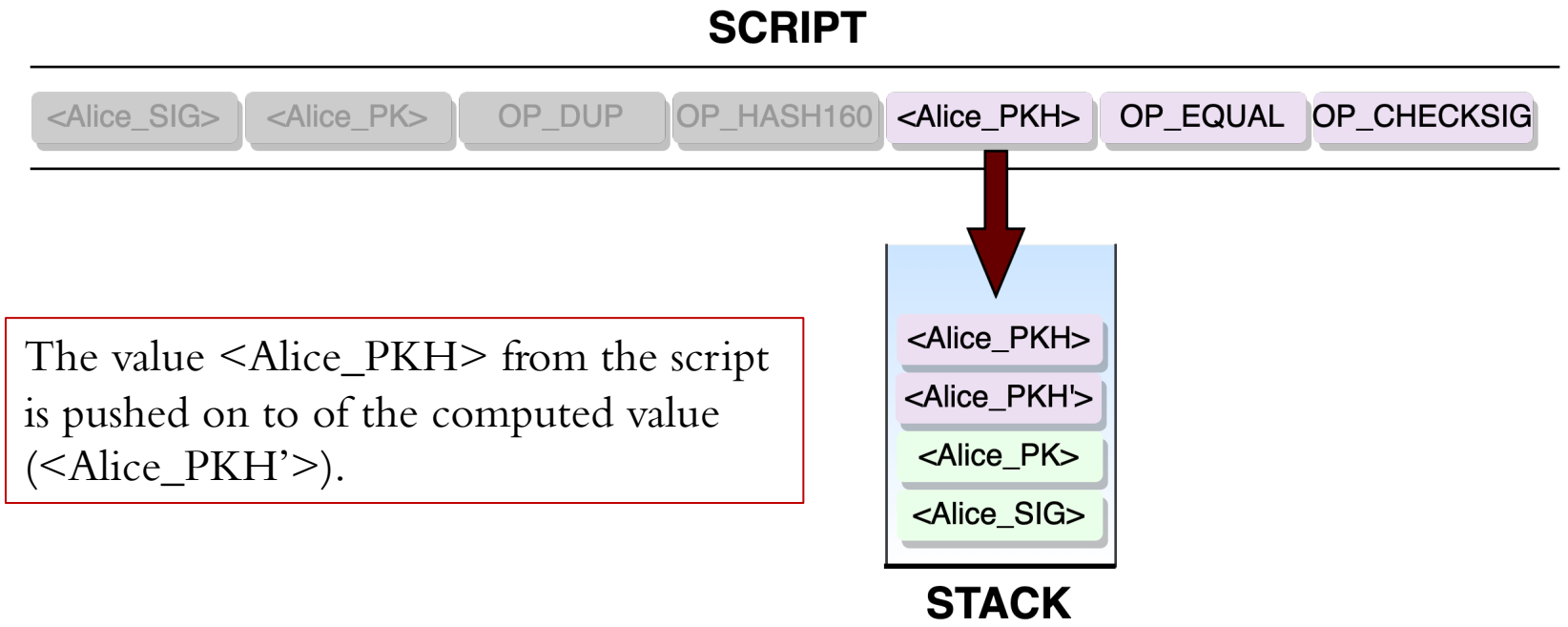
- ***P2PKH***



OP_HASH160 hashes the top item (which results in <Alice_PKH'>) in the stack and push it to the top of the stack.

Transaction Scripts (cont...)

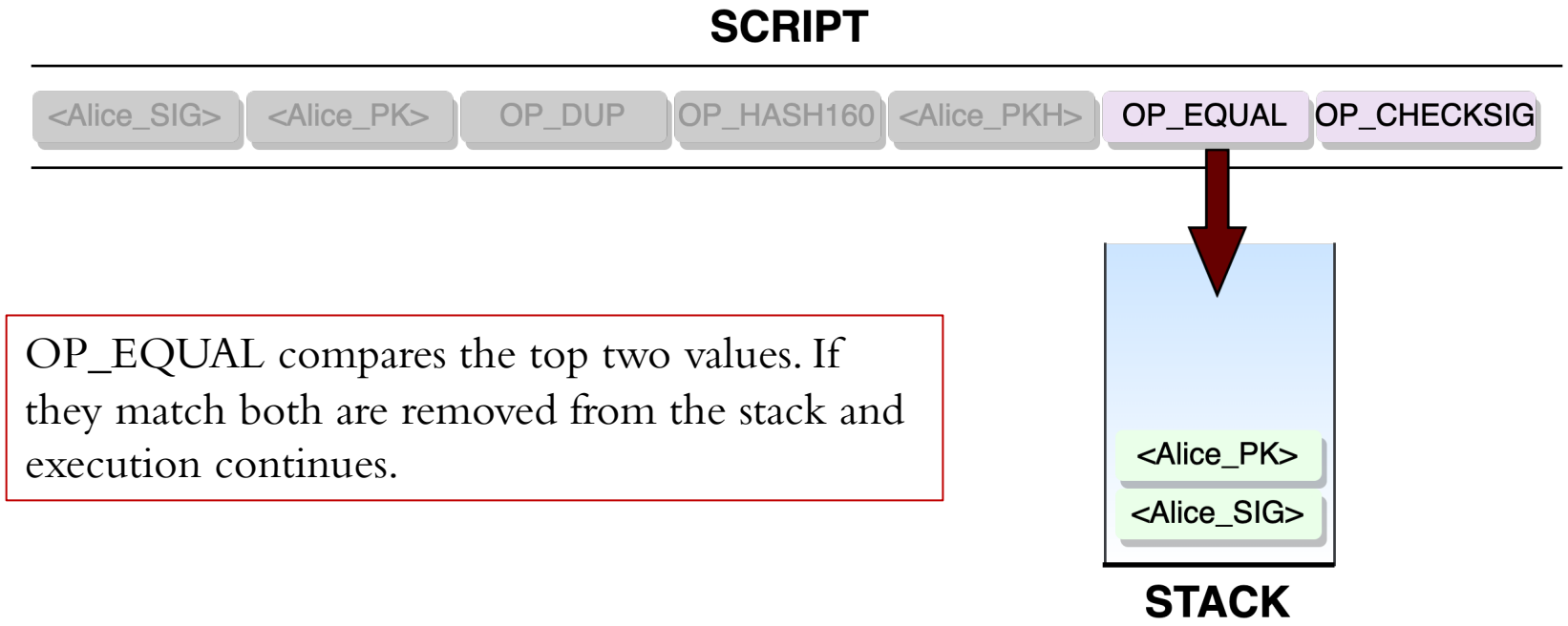
- ***P2PKH***



The value <Alice_PKH> from the script is pushed on to of the computed value (<Alice_PKH'>).

Transaction Scripts (cont...)

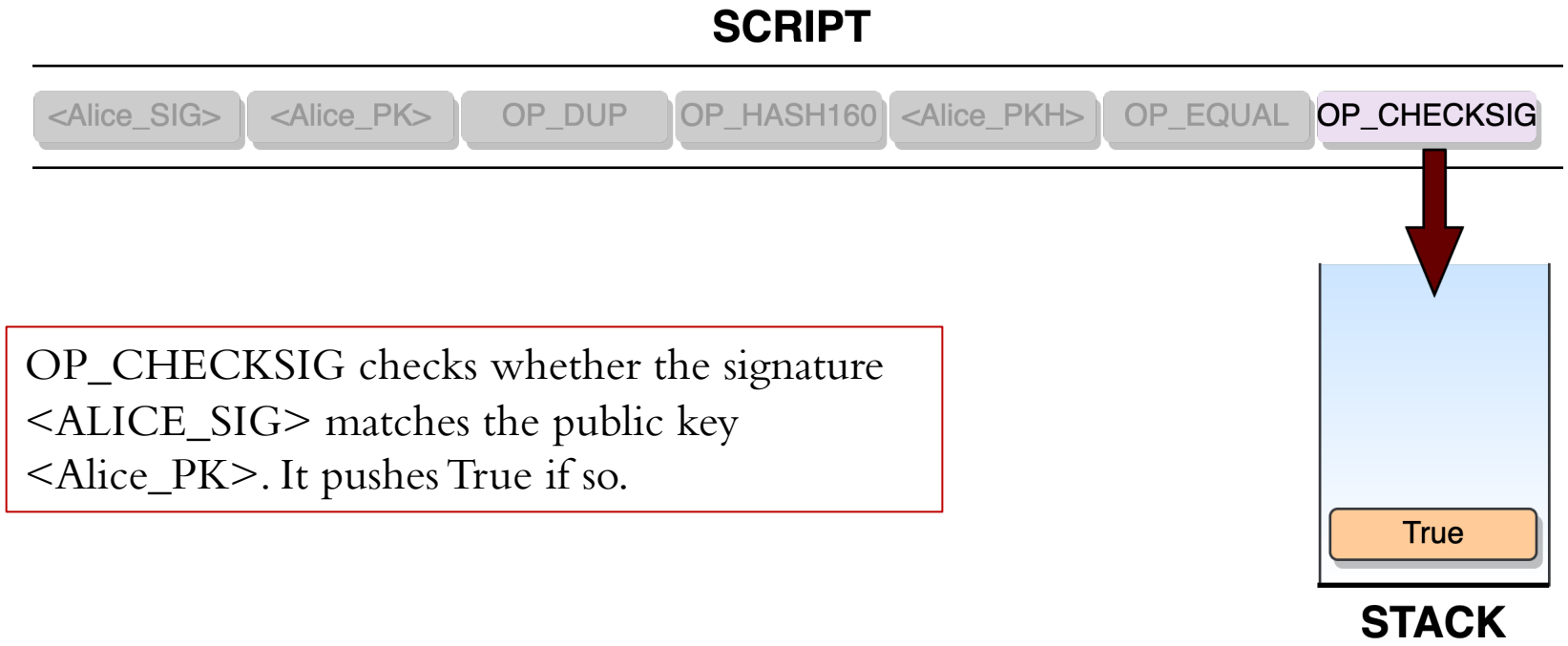
- ***P2PKH***



OP_EQUAL compares the top two values. If they match both are removed from the stack and execution continues.

Transaction Scripts (cont...)

- ***P2PKH***



OP_CHECKSIG checks whether the signature <ALICE_SIG> matches the public key <Alice_PK>. It pushes True if so.

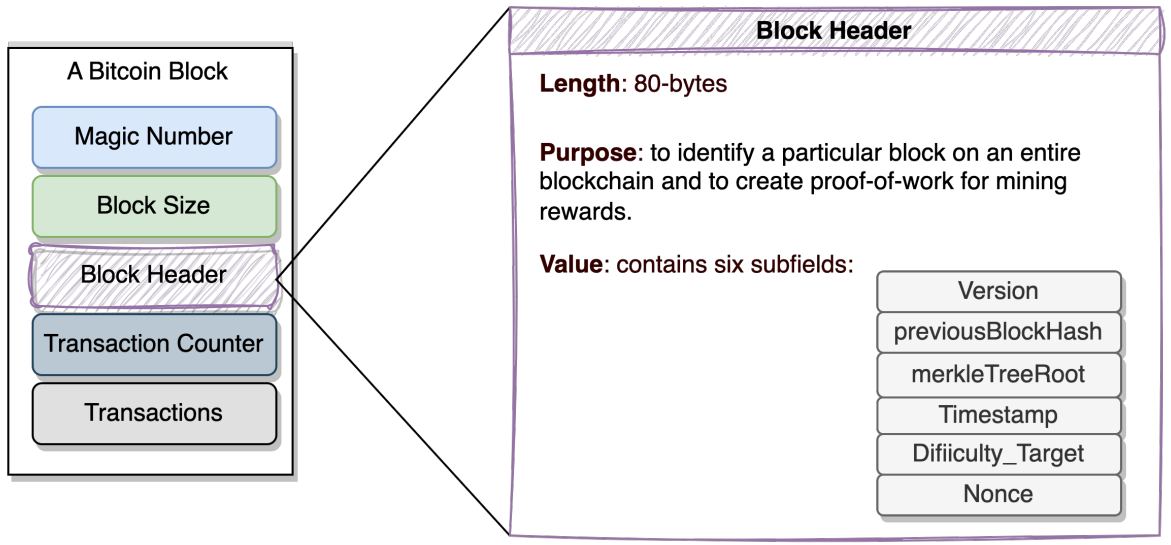
PART III
BITCOIN MINING

BITCOIN MINING

- Bitcoin mining, in general, refers to the process of adding a new bitcoin to the money supply.
- For Bitcoin to function as a currency, miners need to validate transactions and blocks. For miners to do so, however, they need to be incentivised.
- Miners perform six core tasks:
  - ❖ Listen for transactions
  - ❖ Maintain the Bitcoin blockchain and listen for new blocks
  - ❖ Assemble a candidate block
  - ❖ Find a nonce that make the candidate block valid
  - ❖ Wait and hope for the block to be accepted
  - ❖ Get the block reward and transaction fees

BITCOIN MINING CONT...

- A miner, say Alice, first compiles a set of valid transactions that she has from her pending transaction pool into a Merkle tree.
- Alice then create a block with a header that points to the previous block.



- The block header includes a 32-bit nonce field, and she keeps trying different nonces until she found the one that causes the block's hash to be under the target.
- In case Alice exhausted all the possible nonce, yet unable to find a block hash that is under the target, she can change the nonce in the coinbase transaction and try again.

BITCOIN MINING CONT...

- The mining difficult in Bitcoin changes every 2 weeks or every 2,016 blocks.
- The next difficulty is computed as

$$\frac{previousDifficulty * 2016 * 10\,min.}{average\ time\ to\ mine\ last\ 2016\ blocks}$$

- Generally, as more miners join the network and mining hardware gets more efficient, blocks are found faster and the difficulty is increased, so that it always takes about ten minutes to find a block.

BITCOIN MINING CONT...

Network Difficulty vs. Total Hash Rate



BITCOIN MINING CONT...

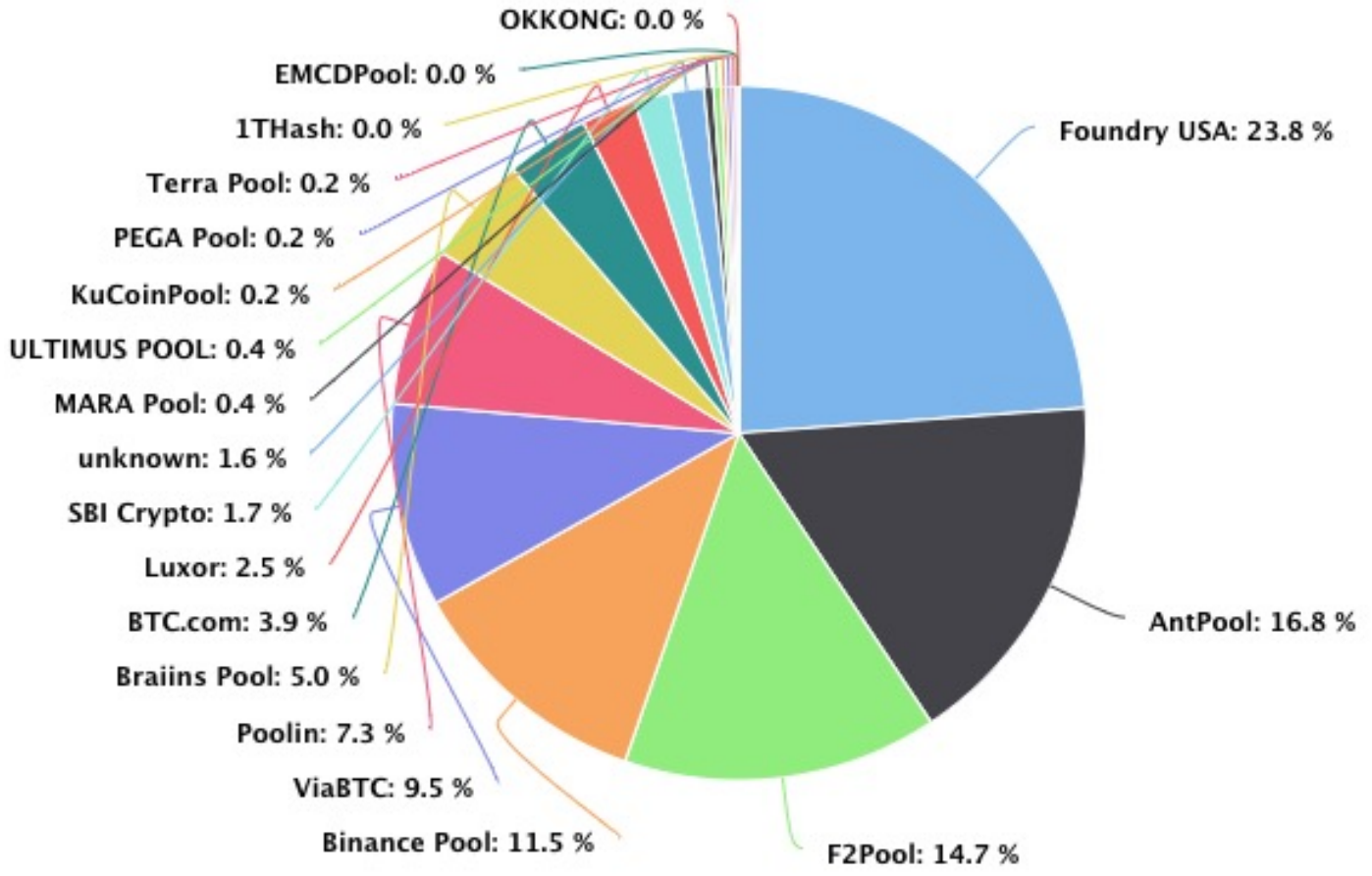
- Miners use different kinds of hardware to find the nonce. The most known types are CPU, GPU, FPGA, and ASIC.
- The hash rate of each type of processor can vary depending on the specific model and its specifications, not to mention the constantly increasing difficulty of Bitcoin.
- CPU (Central Processing Unit): Low hash rate, in the range of a few MH/s (Mega hashes per second).
- GPU (Graphics Processing Unit): Higher hash rate than a CPU, typically in the range of several hundred MH/s to a few GH/s (Giga hashes per second).

BITCOIN MINING CONT...

- FPGA (Field-Programmable Gate Array): Higher hash rate than a GPU, in the range of several GH/s to tens of GH/s.
- ASIC (Application-Specific Integrated Circuit): The highest hash rate, typically in the range of tens of GH/s to several TH/s (Tera hashes per second). ASICs are specifically designed for bitcoin mining and are much more efficient than other types of processors.
- These days, mining pools, which are groups of bitcoin miners who combine their processing power to increase their chances of finding a block and earning rewards, are extensively used.
- Mining pools bring advantages like consistent earnings and lower costs to the miners. However, pool operator dependence and centralization can't be avoided.

BITCOIN MINING CONT...

Bitcoin Mining Pool Distribution (Hashrate Share)



BITCOIN MINING CONT...

- Miners are faced with different choices and strategies to increase their profits. Some of these considerations are:
  - ❖ Which transactions to include
    - ✓ Default: transactions with higher transaction fees
  - ❖ Which block to mine on
    - ✓ Default: to extend the longest known valid chain
  - ❖ Choosing between blocks at the same height
    - ✓ Default: to build on top of the block that they heard about first
  - ❖ When to announce new blocks
    - ✓ Default: immediately
- Miners who deviate from these defaults may follow different strategies.

BITCOIN MINING CONT...

Forking Attack

- Refers to an attack on the bitcoin network that creates two or more separate versions of the blockchain, with conflicting transactions.
- This happens when two or more miners find and solve blocks at approximately the same time, creating two different chains.
- This type of attack is considered a minor risk to the bitcoin network because the network is designed to automatically choose the longest chain and abandon the other chain.
- Additionally, it would require a large amount of computational power (> 51% hashrate) and resources to carry out such an attack, and the likelihood of success is low.

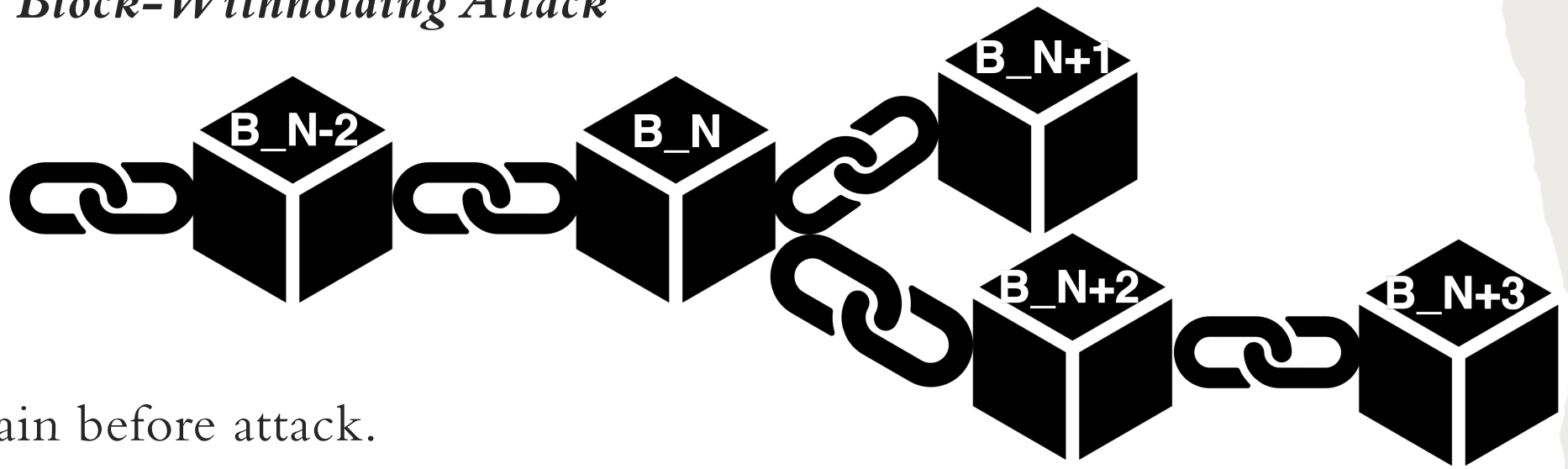
BITCOIN MINING CONT...

Temporary Block-Withholding Attack

- Refers to a type of attack on the bitcoin network that involves a miner intentionally withholding blocks from being broadcast to the rest of the network.
- The miner withholds the block it mined in hopes of finding two blocks in a row. If the miner succeeds, he will be ahead of the public blockchain by two secret blocks.
- The attacker can then broadcast the secret blocks, effectively forcing the rest of the network to switch to the attacker's version of the blockchain, which includes the attacker's desired transactions.
- Temporary block-withholding attacks are considered a minor risk to the bitcoin network, because it would increase the rate of near-simultaneous block announcements.

BITCOIN MINING CONT...

Temporary Block-Withholding Attack



- (***B_N***) Block chain before attack.
- (***B_N+2***) Attacker mines a block, withholds it, starts mining on top of it.
- (***B_N+3***) Attacker finds a second block before the rest of the network, continues to withhold blocks.
- (***B_N+1***) Non-attacker finds a block and broadcasts it. In response, the attacker broadcasts both his blocks, orphaning the non-attacker's block and wasting the mining power that went into finding it.

BITCOIN MINING CONT...

Blacklisting and Punitive Forking

- If miners, for any reason, want to blacklist a transaction from a particular address, there is no effective way in bitcoin since other miners may still be interested to include the transaction in their block.
- A miner with majority hash power may, however, use punitive forking, announcing that he'll refuse to work on a chain containing a transaction originating from this address.
- Other miners would probably exclude transactions from this address as well, since doing otherwise would simply cause their blocks to be elided in forks.

BITCOIN MINING CONT...

Feather Forking

- Refers to an attack that involves using much less than 50% of the hash power to influence the network (refuse transactions from a blacklisted address).
- To blacklist transactions from some address X, the miner will announce that he will refuse to mine directly on any block with a transaction from X (but he'll concede after n confirming blocks).
- The chance of pruning an offending block is α^2 (the miner's hash rate squared)
- For other miners, including a transaction from X induces an α^2 chance of losing a block, and if this value is greater than the transaction fee, it may be safer to join in on the blacklist.

PART IV
PROOF-OF-X

PROOF-OF-X

Requirements for Cryptographic Puzzles

- Mining puzzles need to be *quick to verify*, because every node (including SPV clients) on the network validates every puzzle solution.
- The difficulty of the puzzle should be *adjustable over time* as new users enter the network with increasing amounts of hash power.
- The chance of winning a puzzle solution in any unit of time should be *roughly proportional to the hash power used*.

PROOF-OF-X

Proof-of-Work

- It is a consensus mechanism used in the Bitcoin network to confirm transactions and produce new blocks to the blockchain.
- It involves solving a complex mathematical problem, known as a hash puzzle, which requires a significant amount of computational power. Once a miner solves the puzzle, they can add the block of confirmed transactions to the blockchain and receive a reward in the form of newly minted bitcoins.
- The solution to the puzzle is proof that the miner has done a certain amount of work, hence the name proof-of-work.

PROOF-OF-X

Proof-of-Work (cont.)

- The system is designed to ensure that adding a new block to the blockchain is a slow and costly, making it difficult for any one miner or group of miners to control the network.
- The steps involved in proof-of-work mechanism are:
  - ❖ A miner collects transactions from the unconfirmed transaction pool (mempool) and creates a new block.
  - ❖ The miner then computes a cryptographic hash of the block header, which includes the previous block hash, the transaction data, and a nonce.
  - ❖ The miner repeatedly modifies the nonce value in the block header and calculates the hash until it finds a hash that meets a certain difficulty requirement.

PROOF-OF-X

Proof-of-Work (cont.)

- The first miner to find a hash that meets the difficulty requirement announces the solution to the rest of the network.
- Other miners verify the solution by checking that the hash is below the target value and that all the transactions in the block are valid.
- If the block is valid, the miner who found the solution is rewarded with newly minted bitcoins and all the transactions in the block are confirmed and added to the blockchain.
- The new block's hash is added to the end of the blockchain, making it permanent and unalterable.
- It also incentivizes miners to participate in the network and confirm transactions, thereby maintaining the integrity and security of the blockchain.

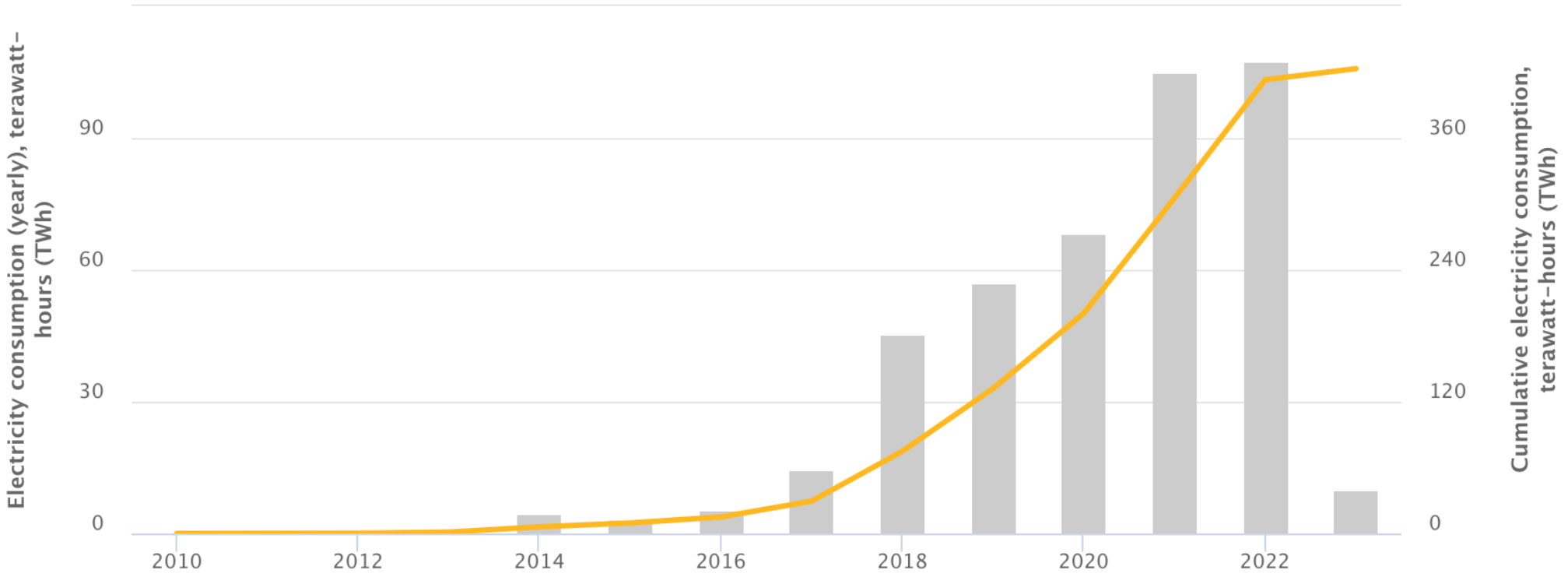
ASIC-Resistant Puzzles

- The objective is to device a puzzle that reduces the gap between the ASIC hardware and general-purpose computers can do. That is, limiting the advantage ASIC has, concerning solving a particular puzzle, to an order of magnitude or less.
- One solution could be to design a puzzle that require a large amount of memory (in addition or instead of large CPU time), called memory-hard puzzles. These puzzles make it more difficult for ASICs to reduce the time required to solve the puzzle.
- Memory-bound puzzles, on the other hand, are those that require a large amount of memory to solve, but the time required to solve them can still be reduced by using ASICs.

Proof of Useful Work

- Proof-of-Works play a critical role in bitcoin to guard against double-spending. Moreover, they have other important application in the protection of email spam and denial of service attacks.
- However, the work and energy expended is generally not useful for anything except proving that work had in fact been done.
- Concerning Bitcoin alone, the process of mining in PoW requires a significant amount of energy, which contributes to a large carbon footprint and environmental impact.

Total Bitcoin Electricity Consumption (2010 – 2022)



Proof of Useful Work (Cont.)

- Proof of useful work is challenging due to the following two properties:
  - ❖ Hardness: refers to the difficulty of performing the work required for validation. That is, it is important to make sure that the work is difficult enough to prevent malicious actors from easily taking over the network. The hardness of the work performed in such schemes need to be adjusted to balance security and efficiency.
  - ❖ Usefulness: refers to the tangible benefit or real-world value of the work performed by miners. The usefulness of the work is intended to make the process of validation more productive and beneficial.
- The real challenge relies on achieving both properties - the work must be difficult enough to secure the network, while also providing real-world value.

Proof of Useful Work (Cont.)

- Two of well-known proof of useful work systems are the Primecoin and Permacoin.
- In ***Primecoin***, miners compete to find a sequence of prime numbers, known as Cunningham chains, with specific properties.
- A Cunningham chain is a sequence of k prime numbers $p_1, p_2, \ldots, p_k$ such that $p_i = 2p_{i-1} + 1$ for each number in the chain.
- For a given challenge x (the hash of the previous block), the miners take the first m bits of x and consider any chain of length k or greater in which the first prime in the chain is an n-bit prime and has the same m leading bits as x to be a valid solution.
- This process provides a useful contribution to the scientific study of prime numbers.

Proof of Useful Work (Cont.)

- In ***Permacoin***, proof-of-storage is used by miners to demonstrate that they have stored a certain amount of data for a certain amount of time.
- Miners prove their storage by periodically providing proof of possession of the data, which is verified by the network. In return for their storage and verification work, miners receive rewards in the form of newly minted Permacoin tokens.
- The concept of Permacoin and proof-of-storage can be used to create decentralized storage solutions and ensure the durability of important data in a cost-effective manner.

Proof of Useful Work (Cont.)

- It is also possible to replace computational puzzles with virtual mining, which refers to a set of approaches the require only a small consumption of computational resources.
- Some of these alternative schemes are proof-of-stake, proof-of-activity, and proof-of-publication.

Proof-of-Stake

- Coin age (currency amount times the holding period) can be a viable alternative to proof-of-work.
- Proof-of-stake is hence a mechanism of defining reward based on coin age.
- More precisely, unlike proof-of-work, which requires miners to perform computationally intensive work to validate transactions and create new blocks, proof-of-stake allows validators to be chosen randomly to validate transactions and create blocks based on the amount of cryptocurrency they hold.
- In a coinstake transaction (as in coinbase transaction in Bitcoin), owners send coins in their possession to themselves and add a predefined percentage as their reward.

Proof-of-Stake (cont.)

- Validators using this scheme still need to get a hash value below a target value, yet the difficulty level, which is inversely proportional to the coin age, is individually determined.
- If a validator acts maliciously, their staked cryptocurrency can be taken as punishment.
- In contrast to proof-of-work, proof-of-stake declares the block chain with the highest total sum of destroyed coin age as the main chain.
- This, attacks like 51%-attack require an attacker to hold a huge amount of coins to succeed.

Proof-of-Stake (cont.)

- Proof-of-stake may be exposed to the nothing-at-stake problem.
- The problem occurs when validators can validate transactions on multiple chains simultaneously without incurring any cost or penalty. This may potentially lead to the creation of multiple conflicting versions of the blockchain.
- A validator validating transactions on multiple chains splits the network and creates multiple conflicting versions of the blockchain. This can erode trust in the system.
- To address the problem, some PoS systems include penalties for validators that validate on multiple chains.

Proof-of-Activity

- It is a hybrid consensus mechanism that combines aspects of both proof-of-work and proof-of-stake.
- It combines the security benefits of proof-of-work and energy-efficient property of proof-of-stake.
- The proof-of-stake component is usually performed by a larger set of node, while the proof-of-work component is used by a small set of nodes.
- The mining process begins the same way as in a proof-of-work, with various miners competing to find a new block. When a new block is found, the system switches to proof-of-stake, with the newly found block containing only a header and the miner's reward address.

Proof-of-Activity (cont.)

- Based on the header details, a new, random group of validators from the blockchain network is selected to validate or sign the new block. The more coins a validator owns, the more chances they have for being selected as a signer.
- Once all the validators sign the newly-found block, it gains the status of a complete block, it gets identified and added to the blockchain.
- In the event that some of the selected signers are unavailable to sign the block, the process moves to the next winning block with a new set of validators being chosen at random (depending on their coin stake).
- The mining fees/rewards are split among the miner and the various validators who contributed in their respective roles to sign off on the block.

PROOF-OF-X CON...

Proof-of-Publication

- A method that uses Bitcoin or Bitcoin-like technologies to authenticate that a certain information was published at a certain date.
- It ensures the validity and authenticity of data on the network. It is designed to provide an efficient and secure way to publish and distribute data on a decentralized network.
- It encompasses other related proofs like proof-of-receipt (proves that every member m in of audience A has received message m), proof-of-nonpublication (proves that message m has not been published) , and proof-of-membership (proves the inclusion of a receiver in the audience).

Proof-of-Publication (cont.)

- Simplified steps to carryout proof-of-publication are:
  - *Creation of the digital signature*
  - *Broadcasting the data and signature*
  - *Validation by network nodes*
  - *Incorporation into the blockchain*

- The digital signature acts as a proof of publication, allowing other nodes in the network to verify the authenticity and validity of the data.

PROOF-OF-X CON...

Consensus	Blockchain	Block Time	TPS	Adversary Tolerance
PoW	Bitcoin	10 min	7	<51%
	Litecoin	2.5 min	56	
	Monero	2 min	Variable	
	Ethereum	12–14 s	15	
	Horizen	2.5 min	N/A	
	CBPoW	Variable	500+	
	PoEWAL	Variable	25	
PoS	Ethereum (PoS)	12 s	TBD	<51%
	Algorand	4.5 s	1000	<33%
	Dfinity	Variable	Variable	<33%
	Cosmos	6 s	1000+	<33%
	PIVX	60 s	173	<51%
	Microchain	9 s	230+	<33%
	PoSCS	Variable	Variable	<51%
PoW + PoS	Decred	5 min	14	<51%

SUMMARY

- In Bitcoin Merkle tree is used to improve the efficiency and security of verifying transactions in the blockchain.
- A Merkle tree is a data structure that organizes data elements in a binary tree where each leaf node represents a transaction and each non-leaf node is the hash of its child nodes.
- Such structure allows for efficient verification of the contents of a block without having to download the entire block. The root of the tree, called the Merkle root, is included in the block header and serves as a fingerprint for the entire block's transactions.

SUMMARY CONT...

- The Bitcoin blockchain is a decentralized, public ledger that records all Bitcoin transactions.
- It is structured as a chain of blocks, where each block contains a list of transactions and a reference to the previous block.
- The blocks are linked using cryptography and are verified by nodes on the network to maintain consensus.
- Each block also includes a nonce, which is a random number used to generate a unique block hash, and a timestamp.
- The block hash serves as a unique identifier for the block and helps ensure the integrity of the blockchain. The combination of these elements creates a secure and transparent record of all Bitcoin transactions.

SUMMARY CONT...

- Bitcoin mining is the process by which new bitcoins are added to the Bitcoin network and existing transactions are verified and added to the blockchain.
- Miners compete to be the first to solve a cryptographic puzzle associated with the creation of a new block. The winner adds the new block to the blockchain and is rewarded with new bitcoins and transaction fees.
- This process helps maintain the integrity and security of the blockchain by ensuring that all transactions are verified and that new blocks are added in a secure and decentralized manner.

SUMMARY CONT...

- Proof-of-Useful-Work is a consensus algorithm that is designed to overcome the inefficiencies of proof-of-work algorithms by incentivizing users to perform useful computations that benefit the network.
- Proof-of-Useful-Work aims to incentivize users to perform computations that are useful to the network, such as data analysis, machine learning, and scientific simulations.
- Some other alternative consensus mechanisms we discussed are proof-of-stake, proof-of-activity, and proof-of-publication.

THANKS!

