7CCSMDLC: Distributed Ledgers & Cryptocurrencies

Lecture 2: Cryptography & Mining

Peter McBurney
Professor of Computer Science
Department of Informatics
King’s College London

Email: peter.mcburney@kcl.ac.uk
Bush House Central Block North – Office 7.15

2020

Outline

- Cryptography & Hashing

Operation of the Bitcoin Blockchain

- Transactions
- Mining and Proof of Work
- Consensus

Licence terms

- Unless otherwise stated, the diagrams are taken from:
  - Andreas Antonopoulos [2017]: *Mastering Bitcoin.* 2nd Edition. O'Reilly.
  - Version on Github at:

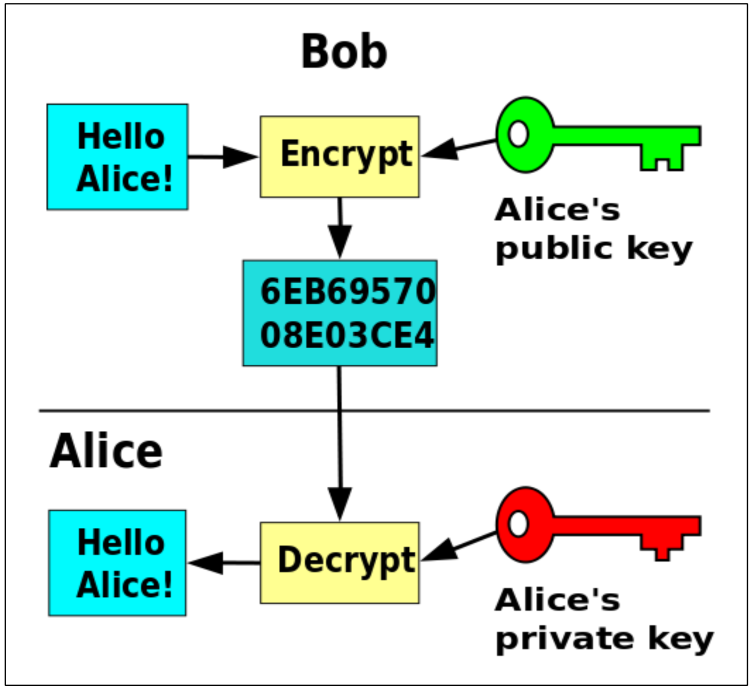
    https://github.com/bitcoinbook/bitcoinbook/

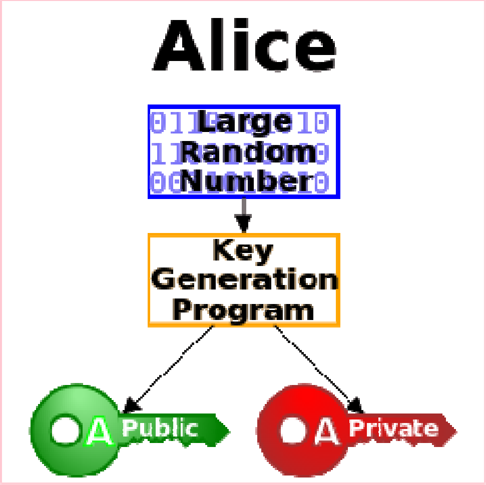
- The licence allowing this is the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.
  - A copy of this license is at: http://creativecommons.org/licenses/by-nc-nd/4.0/

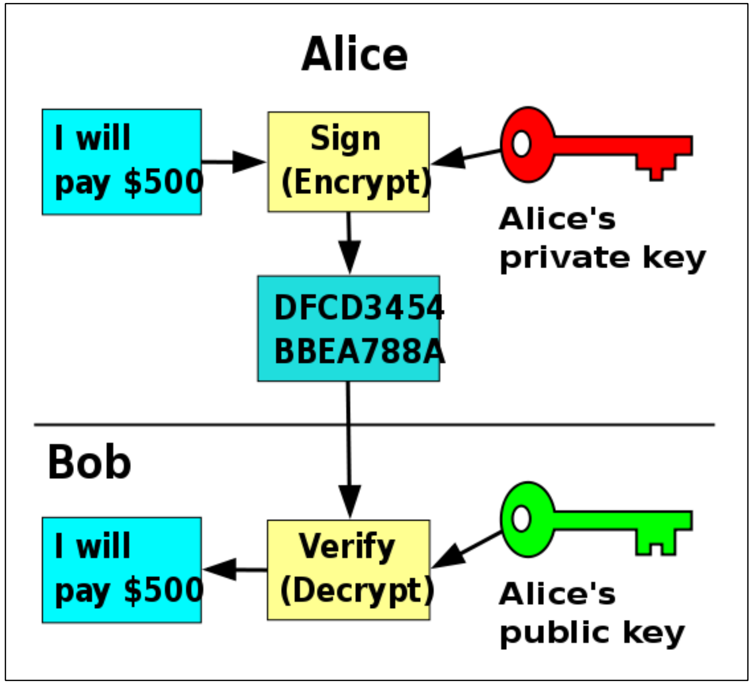
- Any subsequent use of this content is under this this licence.

Cryptography & Hashing

Public & Private Keys







Source: WikiBooks: Communications & Networking

Hashing

- Converts a digital object of arbitrary length (eg, a document, an image) into a single string of fixed length (a hash)
  - Not continuous
    - Two similar documents result in very different hashes.
  - Very hard to reverse engineer
  - Thus, a form of encryption.

See examples next slide.

Hashing in Bitcoin blockchain:

- Hashing of public keys for bitcoin address
- Encryption of private keys
- The work for proof-of-work (hashcash algorithm)
- Each block contains hash of the merkle root of the transactions in that block.
- Each block contains hash of the header of the previous block
- Payloads may be hashed.

Examples of hashing similar phrases

I am Satoshi Nakamoto0 => a80a81401765c8eddee25df36728d732...

I am Satoshi Nakamoto1 => f7bc9a6304a4647bb41241a677b5345f...

I am Satoshi Nakamoto2 => ea758a8134b115298a1583ffb80ae629...

I am Satoshi Nakamoto3 => bfa9779618ff072c903d773de30c99bd...

I am Satoshi Nakamoto4 => bce8564de9a83c18c31944a66bde992f...

I am Satoshi Nakamoto5 => eb362c3cf3479be0a97a20163589038e...

I am Satoshi Nakamoto6 => 4a2fd48e3be420d0d28e202360cfbaba...

I am Satoshi Nakamoto7 => 790b5a1349a5f2b909bf74d0d166b17a...

I am Satoshi Nakamoto8 => 702c45e5b15aa54b625d68dd947f1597...

I am Satoshi Nakamoto9 => 7007cf7dd40f5e933cd89fff5b791ff0...

I am Satoshi Nakamoto10 => c2f38c81992f4614206a21537bd634a...

I am Satoshi Nakamoto11 => 7045da6ed8a914690f087690e1e8d66...

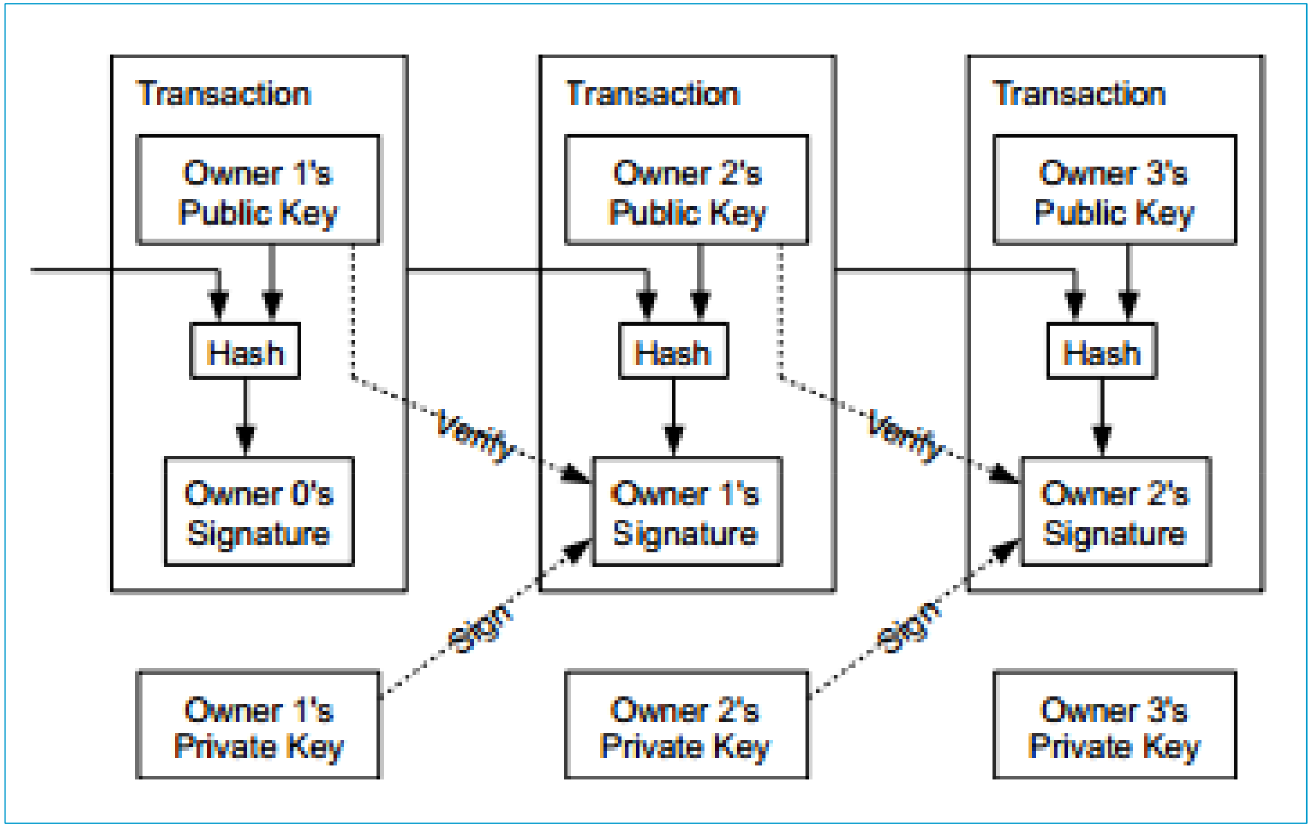
I am Satoshi Nakamoto12 => 60f01db30c1a0d4cbce2b4b22e88b9b...

I am Satoshi Nakamoto13 => 0ebc56d59a34f5082aaef3d66b37a66...

I am Satoshi Nakamoto14 => 27ead1ca85da66981fd9da01a8c6816...

I am Satoshi Nakamoto15 => 394809fb809c5f83ce97ab554a2812c...

Hashing used to chain blocks together



Source: Nakamoto 2008

Bitcoin "address"

A bitcoin address is a string of 26-35 alphanumeric characters in Base58Check encoding, beginning with the number 1 or 3:

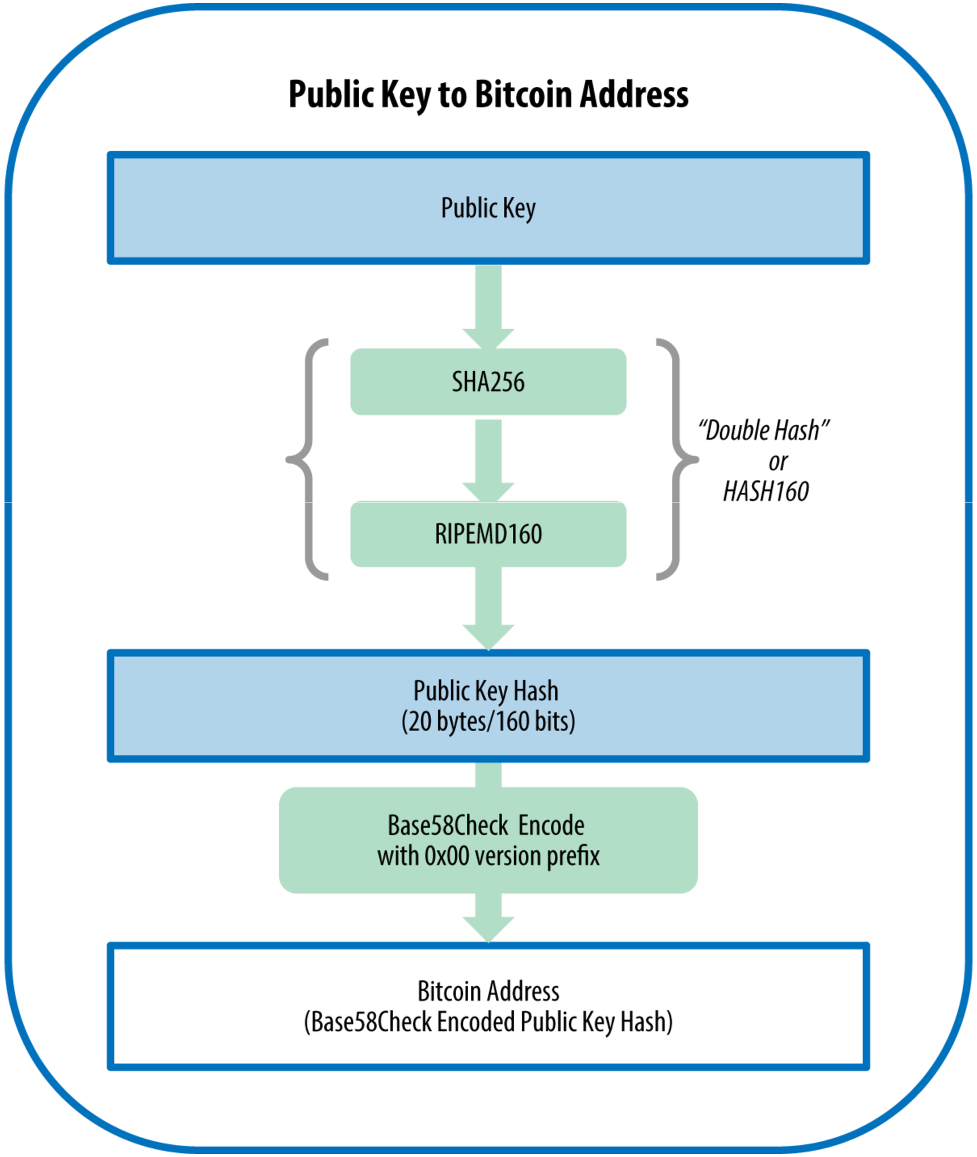
1DSrfJdB2AnWaFNgSbv3MZC2m74996JafV

or

3J98t1WpEZ73CNmQviecrnyiWrnqRhWNLy

- It is a hash of a public key or the hash of a script.
- Two common types of transaction pay to such addresses:
  - P2PKH – Pay-to-Public-Key-Hash
  - P2SH – Pay-to-Script-Hash
- It represents the destination of a payment, and acts to redeem the encumbrance of a payment.

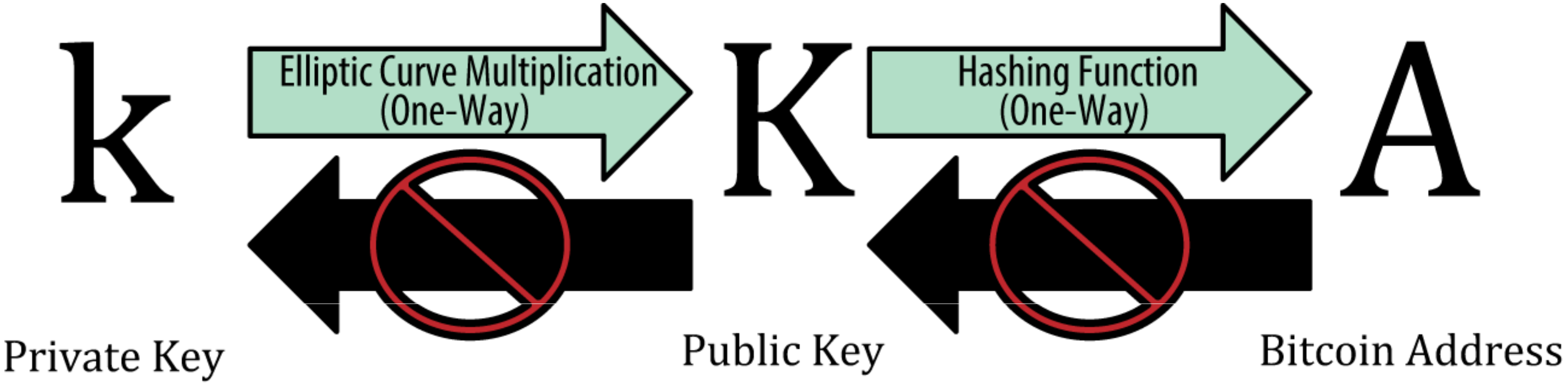
Public key conversion to Bitcoin address



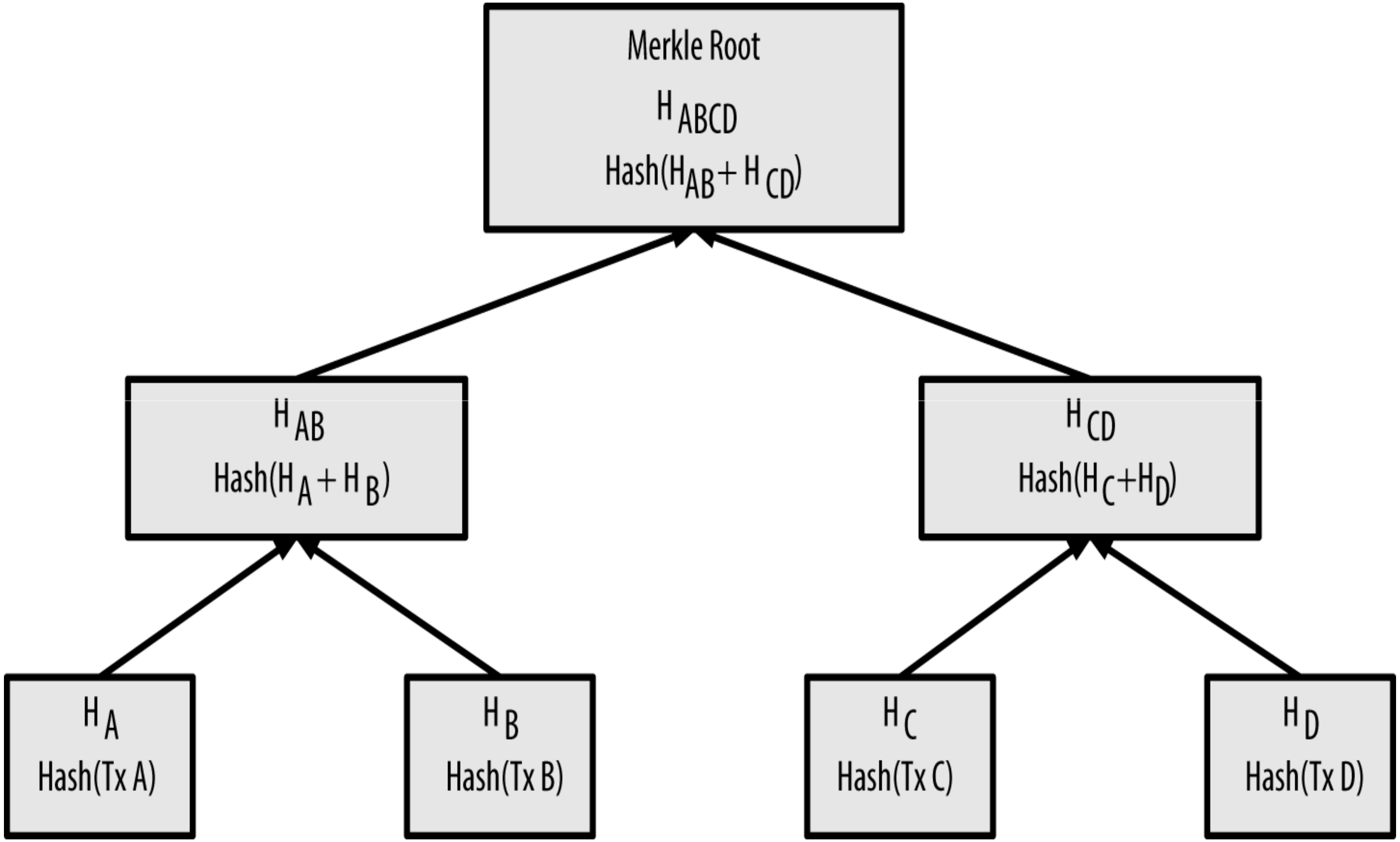
Public Key to Bitcoin Address

Public Key
SHA256
RIPEMD160
"Double Hash" or HASH160
Public Key Hash (20 bytes/160 bits)
Base58Check Encode with 0x00 version prefix
Bitcoin Address (Base58Check Encoded Public Key Hash)

Private and public keys and Bitcoin address

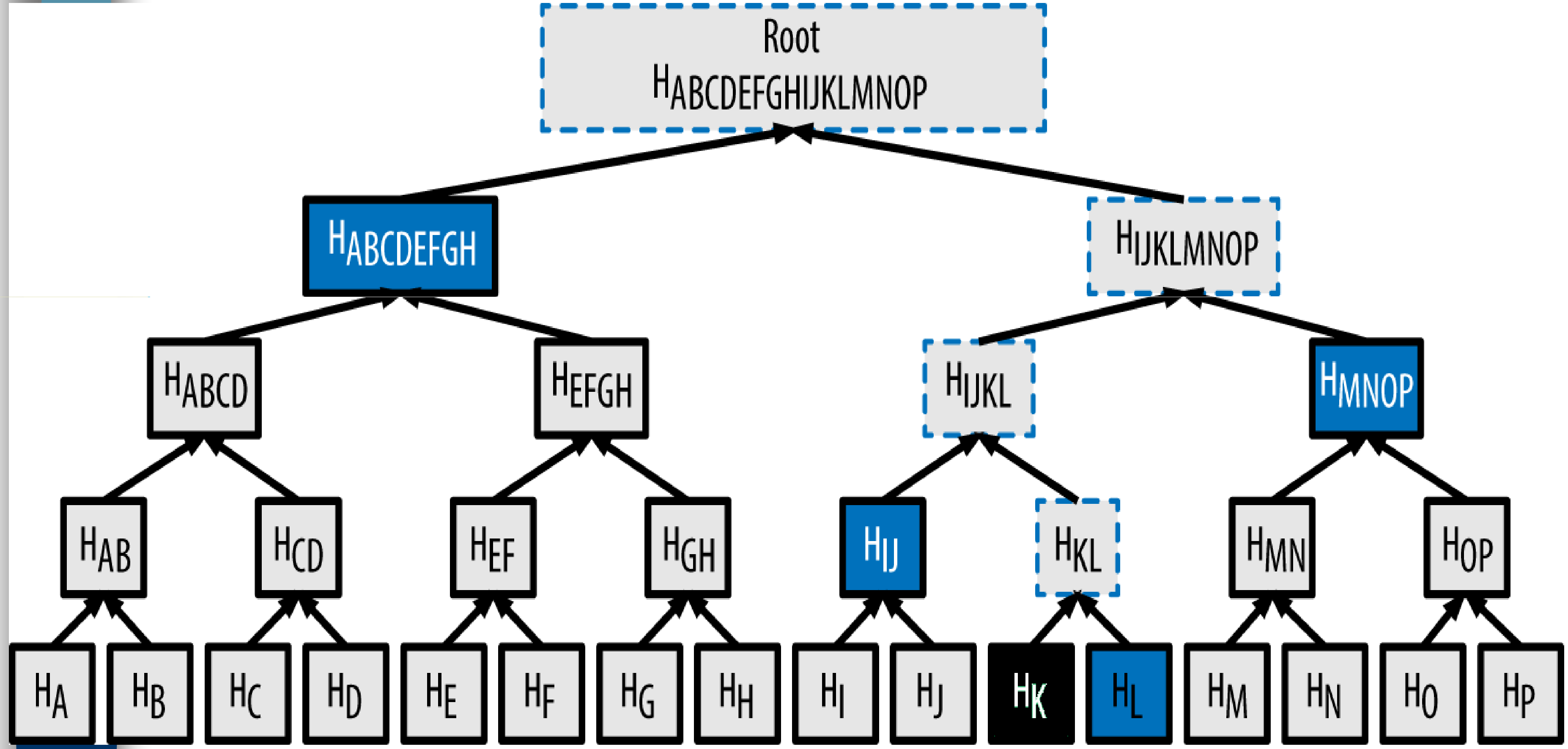

k
Elliptic Curve Multiplication (One-Way)
K
Hashing Function (One-Way)
A
Private Key
Public Key
Bitcoin Address

Merkle Tree


Merkle Root
H_{ABCD}
$\mathrm{Hash}(H_{AB} + H_{CD})$
H_{AB}
$\mathrm{Hash}(H_A + H_B)$
H_{CD}
$\mathrm{Hash}(H_C + H_D)$
H_A
Hash(Tx A)
H_B
Hash(Tx B)
H_C
Hash(Tx C)
H_D
Hash(Tx D)

Merkle path

To prove transaction K included in hash, need only provide 4 hashes (each 32 bytes long): hashes for L, IJ, MNOP & ABCDEFGH.



Operation of the Bitcoin Blockchain

Bitcoin blockchain - Components

- Bitcoin
  - 1 satoshi = 10^-8 Bitcoin = 0.00000001 Bitcoin = smallest possible unit
  - 1 Bitcoin = 100 million satoshis
  - 1 MilliBit = 0.001 Bitcoin =100,000 satoshis

- Total number of BTC to be issued: 2,099,999,997,690,000 satoshis
  - Almost 21 million BTC
  - Will be achieved in ca. 2140 (13.4 million blocks)

Components:

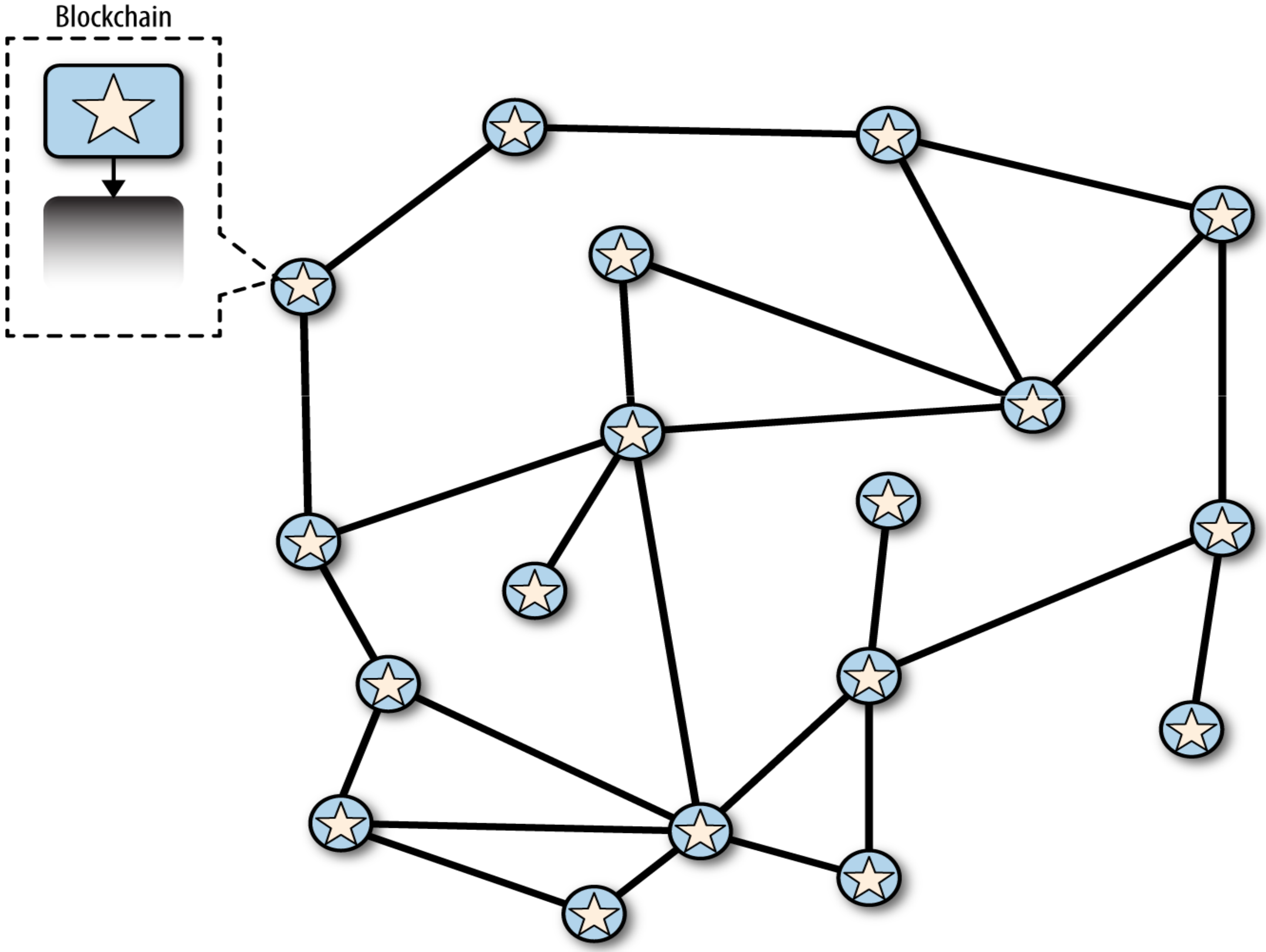
- Users with wallets

- Transactions

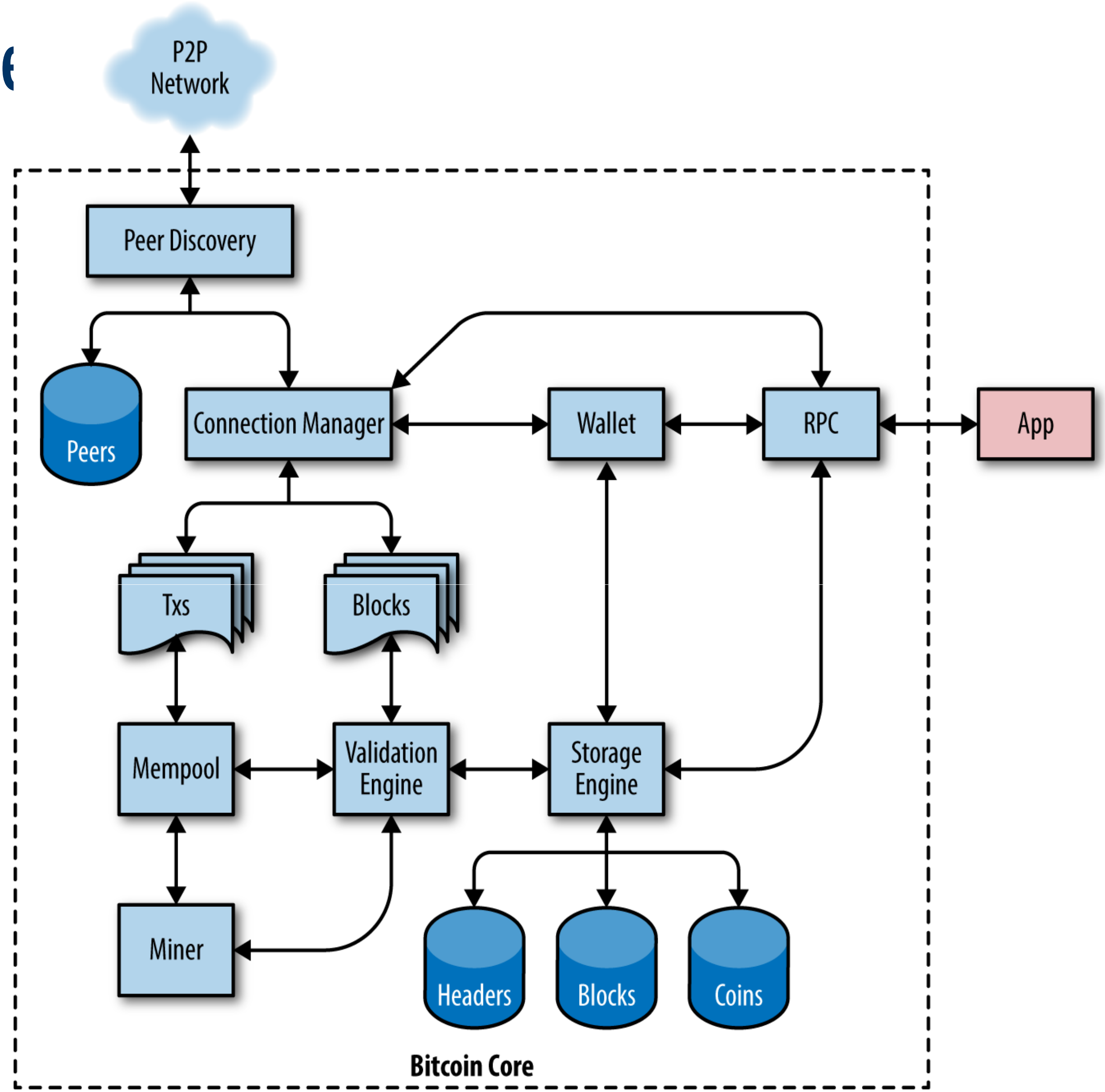
- Miners

- Light vs full clients.

Blockchain assumes a peer-to-peer (P2P) network
No node is in control.



Bitcoin core



Bitcoin Scripting Language I

- Called “**Script**”
  - Reverse-Polish notation stack-based execution language
  - Syntax is like Forth

- Two stack operations:
  - **Push** (adds an item to the top of the stack)
  - **Pop** (removes the item at the top of the stack)

- Items are processed left to right
  - Eg: OP_ADD
    - Pops two items from stack, adds them, and pushes sum to stack

Bitcoin Scripting Language 2

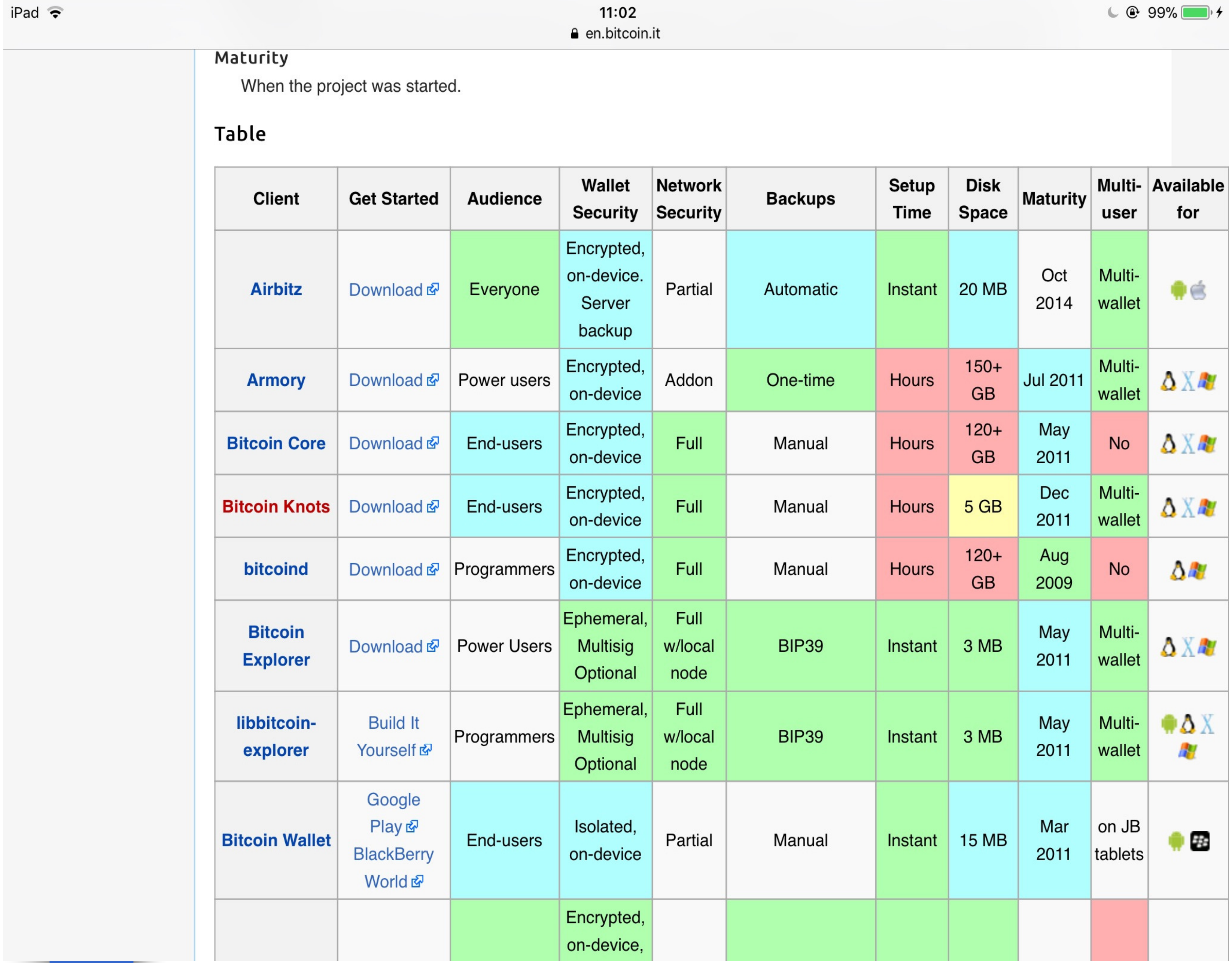
- Script is deliberately simple & widely applicable
  - Not hardware dependent
  - Enables execution on devices with limited memory (eg, embedded devices)
  - Stateless
    - No state prior to execution, no state saved after execution

- Does not permit loops or complex program control features
  - This means predictable execution times
  - Precludes attacks
  - No infinite loops
  - Not Turing-complete.

- Ethereum was developed to allow Turing-complete computation over a blockchain

Wallets

- Wallet is the primary user interface
  - Controls access to a user's bitcoin
  - Manages keys and addresses
  - Tracks current balance
  - Enables creation and signing of transactions.
- May be held on client machine or on an exchange
- Wallet can keep a copy of the transaction
  - Or can query the chain when needed
- Wallet also refers to the data structure used to store and manage a user's keys and address.



Maturity
When the project was started.

Table

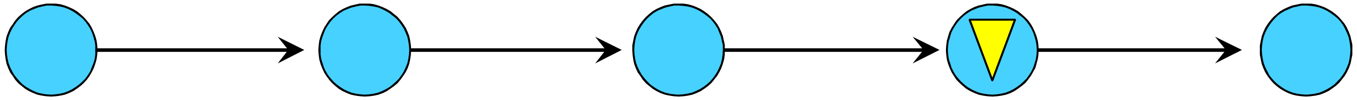
Client	Get Started	Audience	Wallet Security	Network Security	Backups	Setup Time	Disk Space	Maturity	Multi-user	Available for
Airbitz	Download	Everyone	Encrypted, on-device. Server backup	Partial	Automatic	Instant	20 MB	Oct 2014	Multi-wallet	🤖 🍎
Armory	Download	Power users	Encrypted, on-device	Addon	One-time	Hours	150+ GB	Jul 2011	Multi-wallet	🐧 X ⊞
Bitcoin Core	Download	End-users	Encrypted, on-device	Full	Manual	Hours	120+ GB	May 2011	No	🐧 X ⊞
Bitcoin Knots	Download	End-users	Encrypted, on-device	Full	Manual	Hours	5 GB	Dec 2011	Multi-wallet	🐧 X ⊞
bitcoind	Download	Programmers	Encrypted, on-device	Full	Manual	Hours	120+ GB	Aug 2009	No	🐧 ⊞
Bitcoin Explorer	Download	Power Users	Ephemeral, Multisig Optional	Full w/local node	BIP39	Instant	3 MB	May 2011	Multi-wallet	🐧 X ⊞
libbitcoin-explorer	Build It Yourself	Programmers	Ephemeral, Multisig Optional	Full w/local node	BIP39	Instant	3 MB	May 2011	Multi-wallet	🤖 🐧 X ⊞
Bitcoin Wallet	Google Play BlackBerry World	End-users	Isolated, on-device	Partial	Manual	Instant	15 MB	Mar 2011	on JB tablets	🤖 ⊞
			Encrypted, on-device,							

Source: https://en.bitcoin.it/wiki/Clients

Transaction Process

Any node

Miners



Origination

Broadcast

Validation of transaction and propagation

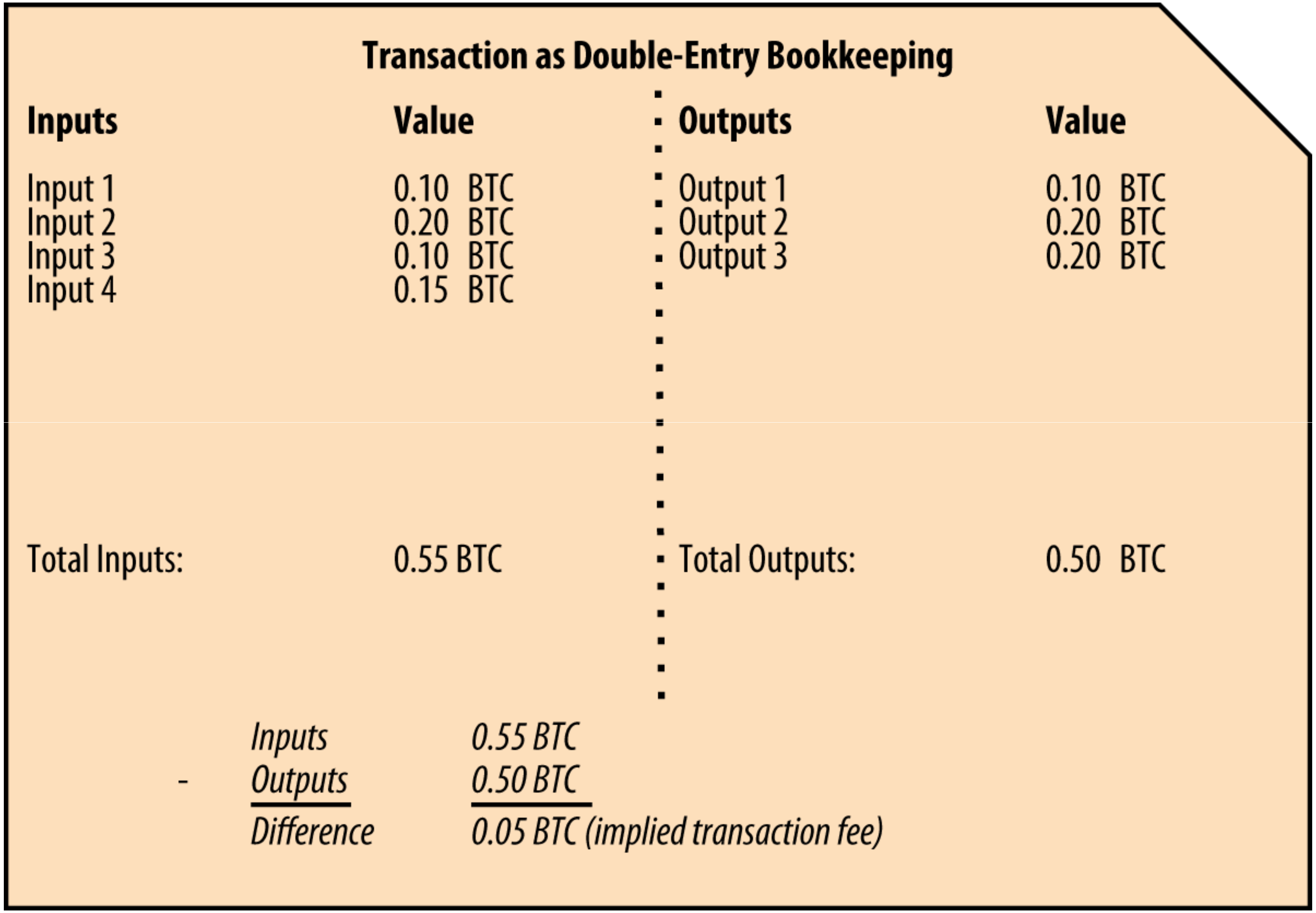
Transactions aggregated into Block and PoW

Validation of block and Confirmation

Transactions

- Transactions move value from inputs to outputs
- A transaction has at least 1 input and at least 1 output
- Outputs < Inputs
  - Implied difference between outputs and inputs is taken by the miner as a fee for processing the transaction

Transactions as inputs and outputs

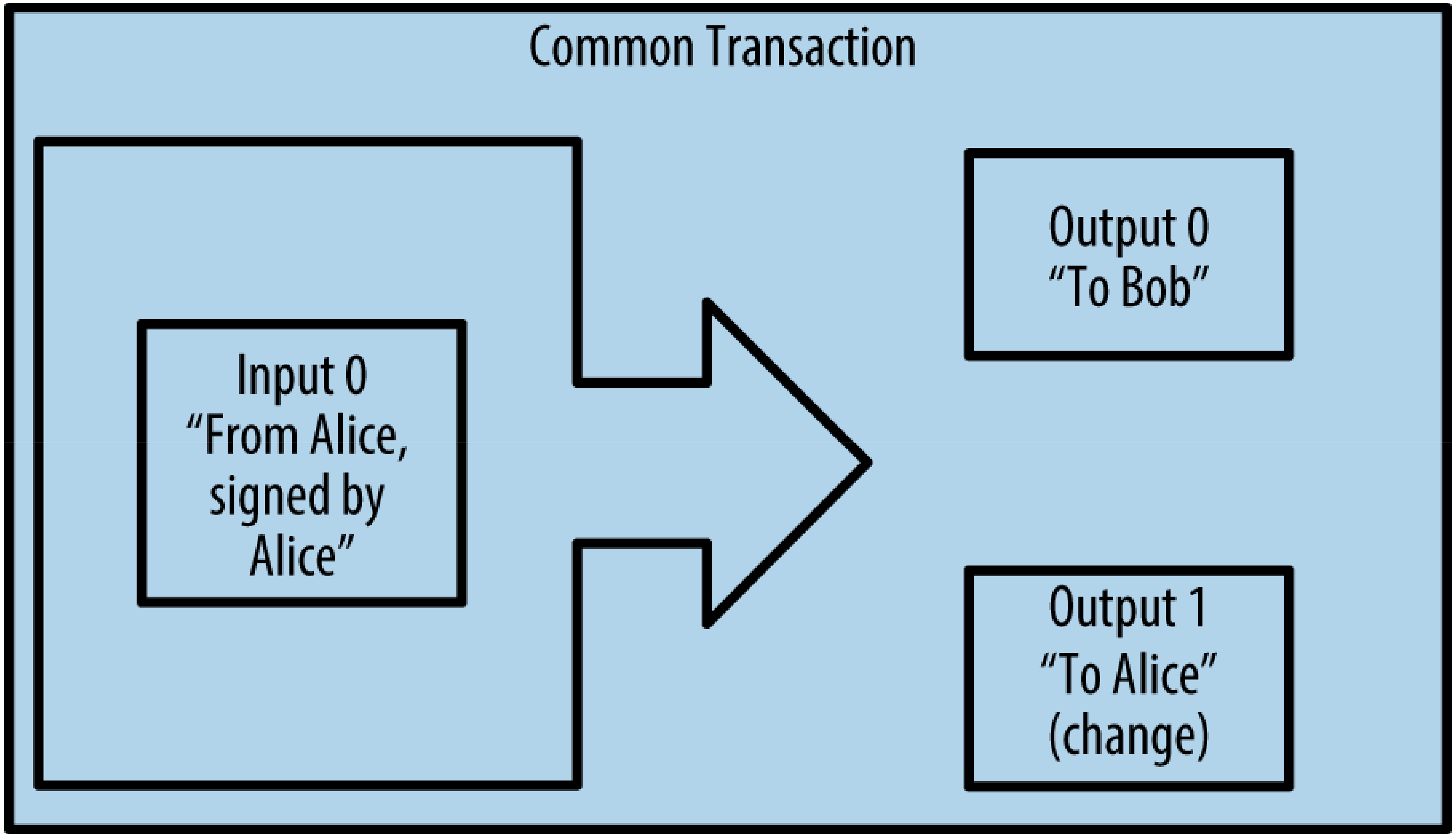


Transaction as Double-Entry Bookkeeping

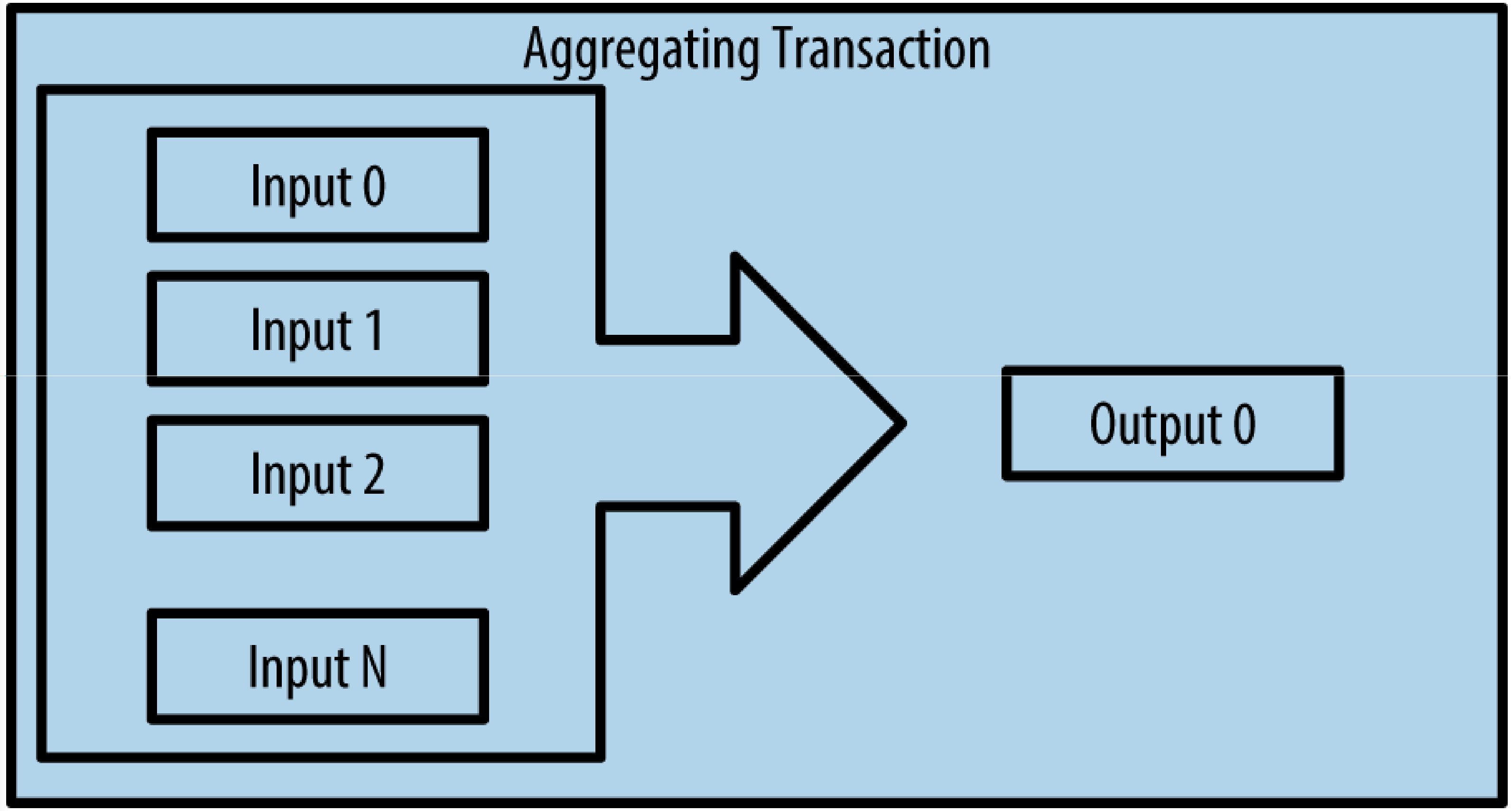
Inputs	**Value**	**Outputs**	**Value**
Input 1	0.10 BTC	Output 1	0.10 BTC
Input 2	0.20 BTC	Output 2	0.20 BTC
Input 3	0.10 BTC	Output 3	0.20 BTC
Input 4	0.15 BTC		
Total Inputs:	0.55 BTC	Total Outputs:	0.50 BTC

	Inputs	*0.55 BTC*
-	*Outputs*	*0.50 BTC*
	Difference	*0.05 BTC (implied transaction fee)*

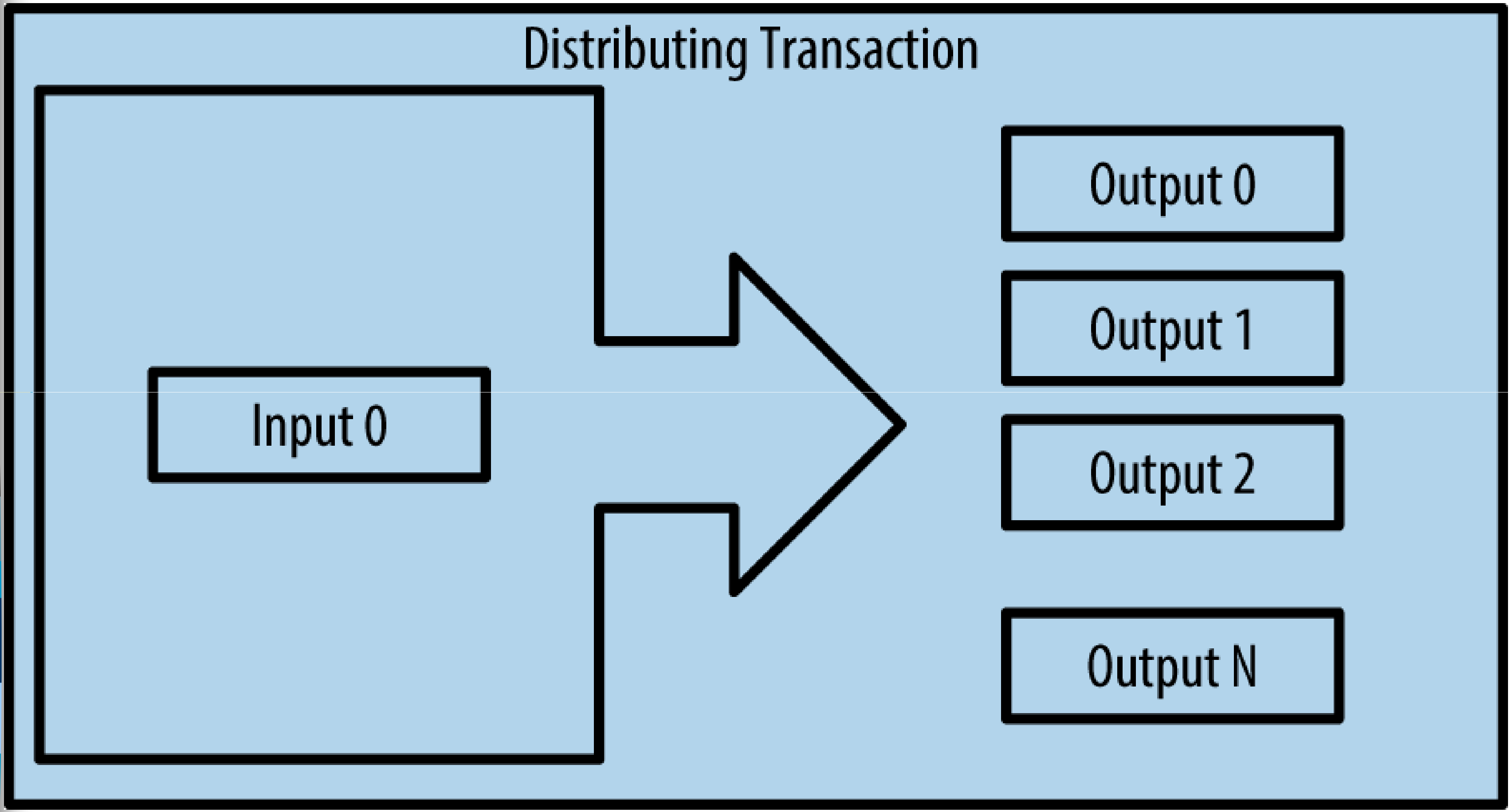
Common transaction: one to one plus change



Transaction aggregating funds: Many to one

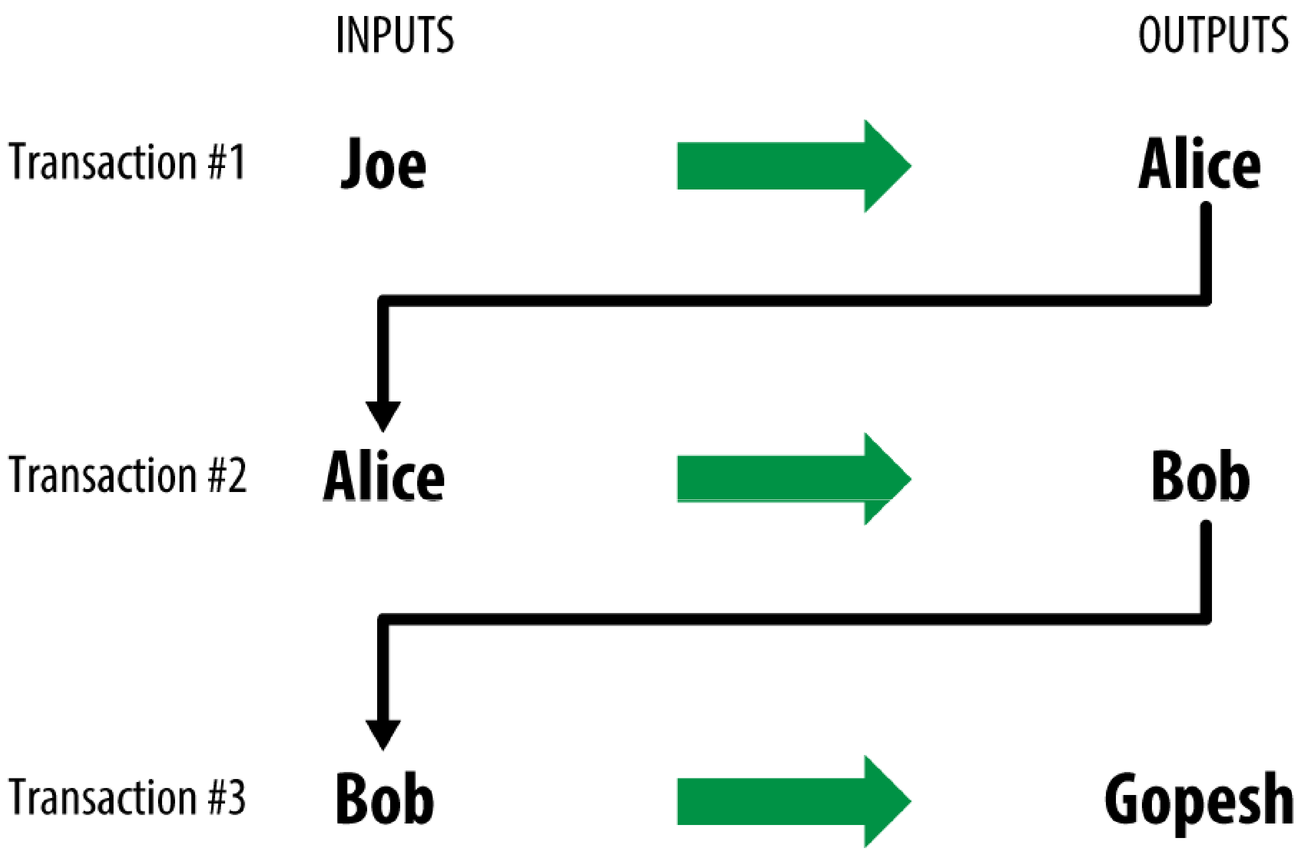


Transaction distributing funds: one to many



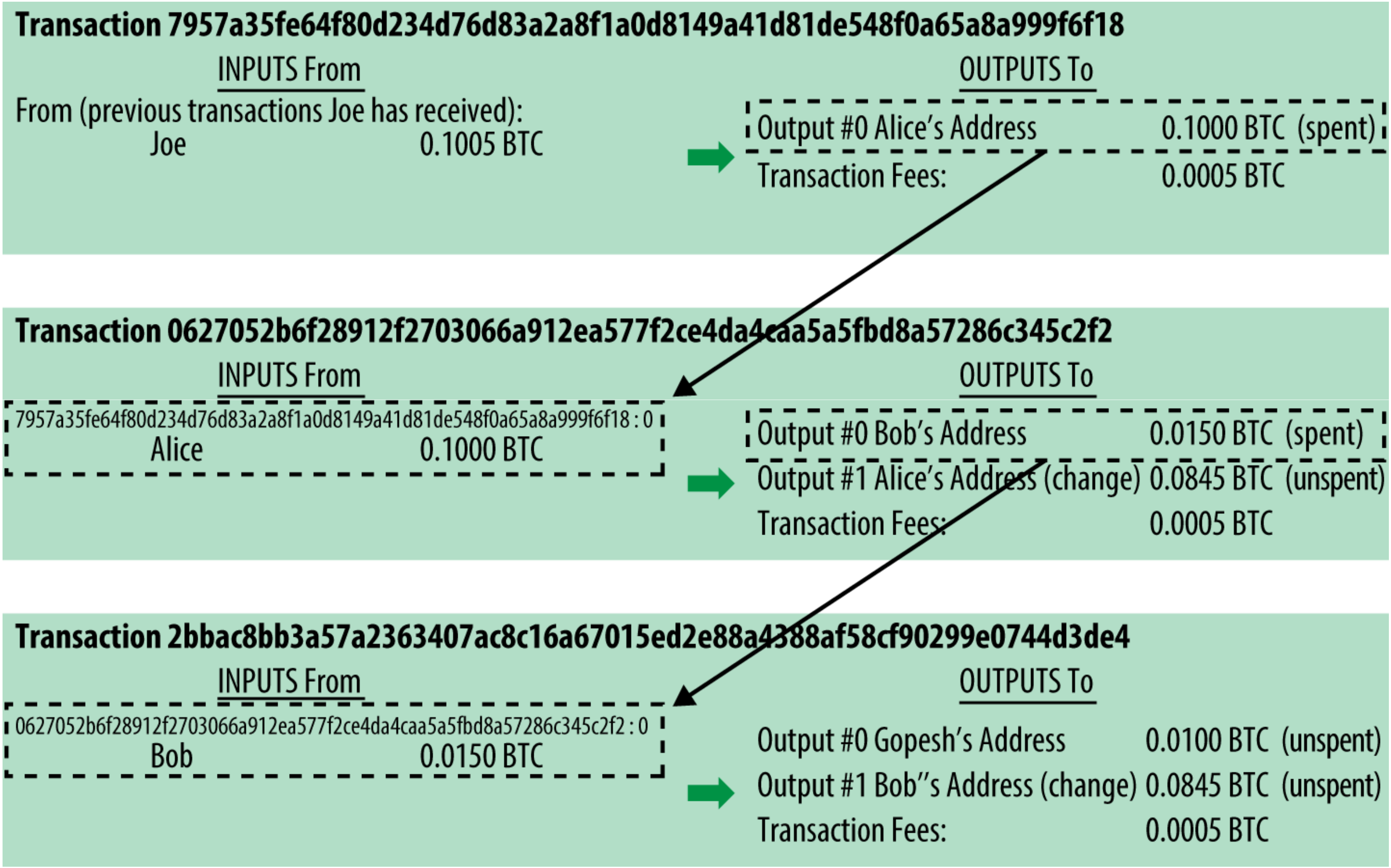
Metaphor – Mixing buckets of water

Images: West Roane County Fire Department



	INPUTS		OUTPUTS
Transaction #1	Joe	→	Alice
Transaction #2	Alice	→	Bob
Transaction #3	Bob	→	Gopesh

A chain of transactions: Joe to Alice to Bob



Transaction 7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18

INPUTS From

From (previous transactions Joe has received):
Joe 0.1005 BTC

OUTPUTS To

Output #0 Alice’s Address 0.1000 BTC (spent)
Transaction Fees: 0.0005 BTC

Transaction 0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2

INPUTS From

7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18 : 0
Alice 0.1000 BTC

OUTPUTS To

Output #0 Bob’s Address 0.0150 BTC (spent)
Output #1 Alice’s Address (change) 0.0845 BTC (unspent)
Transaction Fees: 0.0005 BTC

Transaction 2bbac8bb3a57a2363407ac8c16a67015ed2e88a4388af58cf90299e0744d3de4

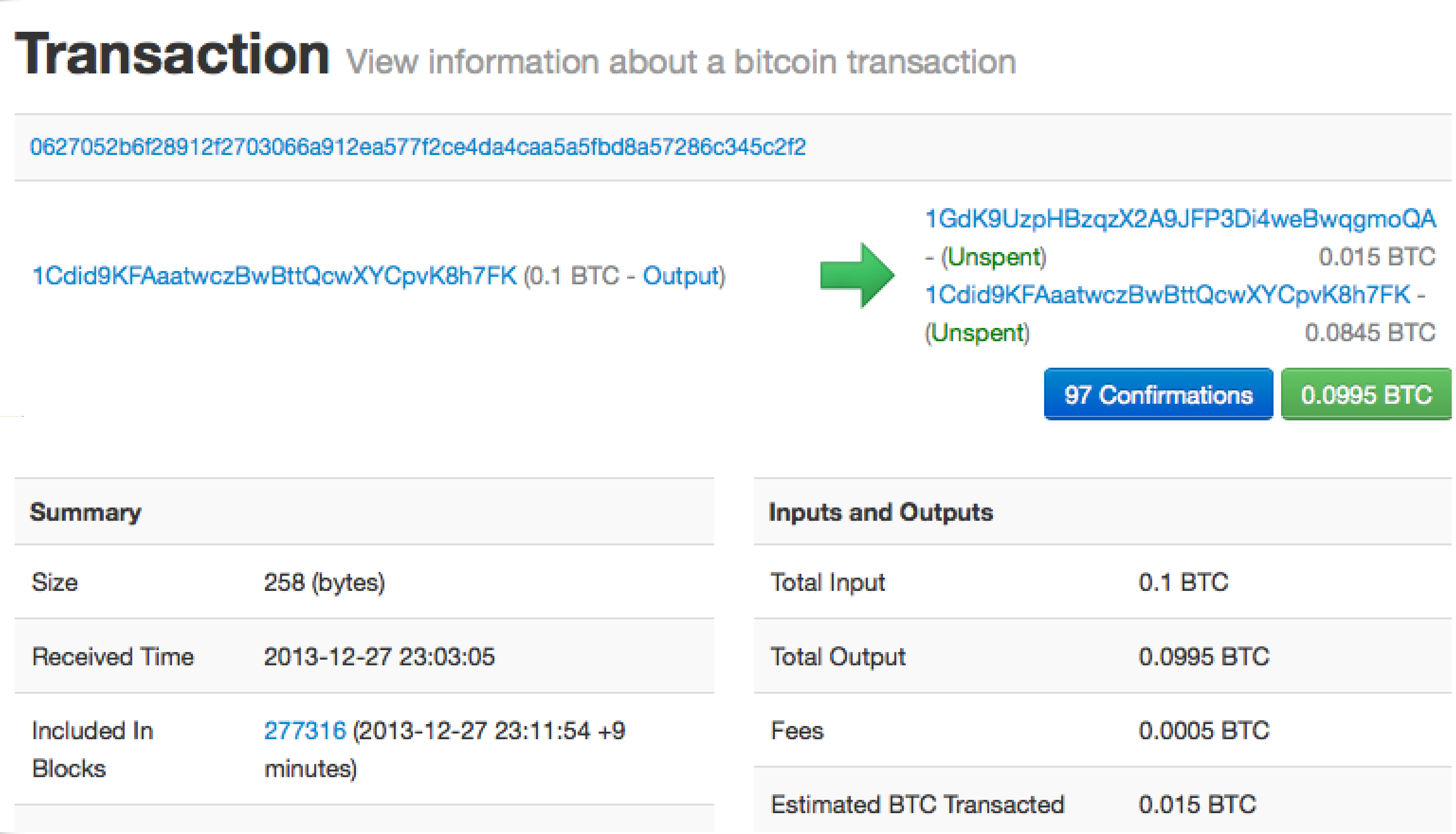
INPUTS From

0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2 : 0
Bob 0.0150 BTC

OUTPUTS To

Output #0 Gopesh’s Address 0.0100 BTC (unspent)
Output #1 Bob’’s Address (change) 0.0845 BTC (unspent)
Transaction Fees: 0.0005 BTC

Transactions – block explorer view



Transaction View information about a bitcoin transaction

0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2

1Cdid9KFAaatwczBwBttQcwXYCpvK8h7FK (0.1 BTC - Output)

1GdK9UzpHBzqzX2A9JFP3Di4weBwqgmoQA - (Unspent) 0.015 BTC

1Cdid9KFAaatwczBwBttQcwXYCpvK8h7FK - (Unspent) 0.0845 BTC

97 Confirmations 0.0995 BTC

Summary	
Size	258 (bytes)
Received Time	2013-12-27 23:03:05
Included In Blocks	277316 (2013-12-27 23:11:54 +9 minutes)

Inputs and Outputs	
Total Input	0.1 BTC
Total Output	0.0995 BTC
Fees	0.0005 BTC
Estimated BTC Transacted	0.015 BTC

Transactions

Transaction Outputs

For most transaction, there are two parts:

- An amount of Bitcoin (denominated in satoshis)

- A locking script (an “encumbrance”)
  - The amount is locked unless specific conditions are met

The intended recipient has to provide something redeem the payment

- Typically they provide their signature (which encodes their private key) and a hash of their public key (their Bitcoin address)
- They may also provide their signature (which encodes their private key) and a hash of a script.
- Some transactions require multiple parties to provide something before the locking script is unlocked.

UTXO

- **Unspent Transaction Output (UTXO)** is the output of a transaction which may be spent as an input in a subsequent transaction.

- “Sending” a recipient some bitcoin is done by creating some UTXO registered to their address
  - Encumbered to their public key hash or to a script

- All the UTXO of the system is known by every node
  - Held in a database called **UTXO set** or **UTXO pool**.

- It is locked to a specific address and may be scattered.

- A wallet will aggregate the UTXO belonging to a single address.

5 Standard Transactions

These are based on what is needed to redeem the payment
(ie, to satisfy the encumbrance)

- Pay-to-Public-Key-Hash (P2PKH)
  - A hash of a specific public key (a Bitcoin address) is needed to redeem
- Pay-to-Public-Key
  - Mostly used in coinbase transactions
- Multi-sig (multiple-signature)
  - limited to 15 keys
  - M of N schemes
- Pay-to-Script-Hash (P2SH)
- Data Output
  - 40 bytes of non-payment data to a Transaction output.

Mining & Consensus

Four parts of decentralized consensus

A Independent verification of each transaction, by every full node

B Independent aggregation of those transactions into new blocks by mining nodes

together with demonstrated computation through a Proof-of-Work algorithm

C Independent verification of the new blocks by every node and assembly into a chain

D Independent selection, by every node, of the chain with the most cumulative computation demonstrated through Proof-of-Work.

A: Independent verification of transactions

Each node checks against the following list of criteria:

- The transaction’s syntax and data structure is correct.
- Neither lists of inputs or outputs are empty.
- The transaction size in bytes is less than MAX_BLOCK_SIZE.
- Each output value, as well as the total, is within the allowed range of values
- None of the inputs have hash=0, N=–1 (coinbase transactions should not be relayed)
- nLocktime is equal to INT_MAX, or nLocktime and nSequence values are satisfied according to MedianTimePast.
- The transaction size in bytes is greater than or equal to 100.
- The number of signature operations (SIGOPS) contained in the transaction is less than the signature operation limit.
- The unlocking script can only push numbers on the stack, and the locking script must match isStandard forms.
- 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 is rejected.
- For each input, look in the main branch and the transaction pool to find the referenced output transaction. If the output transaction is missing for any input, this will be an orphan transaction. Add to the orphan transactions pool, if a matching transaction is not already in the pool.
- For each input, if the referenced output transaction is a coinbase output, it must have at least COINBASE_MATURITY confirmations.
- For each input, the referenced output must exist and cannot already be spent.
- Using the referenced output transactions to get input values, check that each input value, as well as the sum, are in the allowed range of values (less than 21m coins, more than 0).
- Reject if the sum of input values is less than sum of output values.
- Reject if transaction fee would be too low (minRelayTxFee) to get into an empty block.
- The unlocking scripts for each input must validate against the corresponding output locking scripts.

B: Aggregation TXs into Blocks & Mining

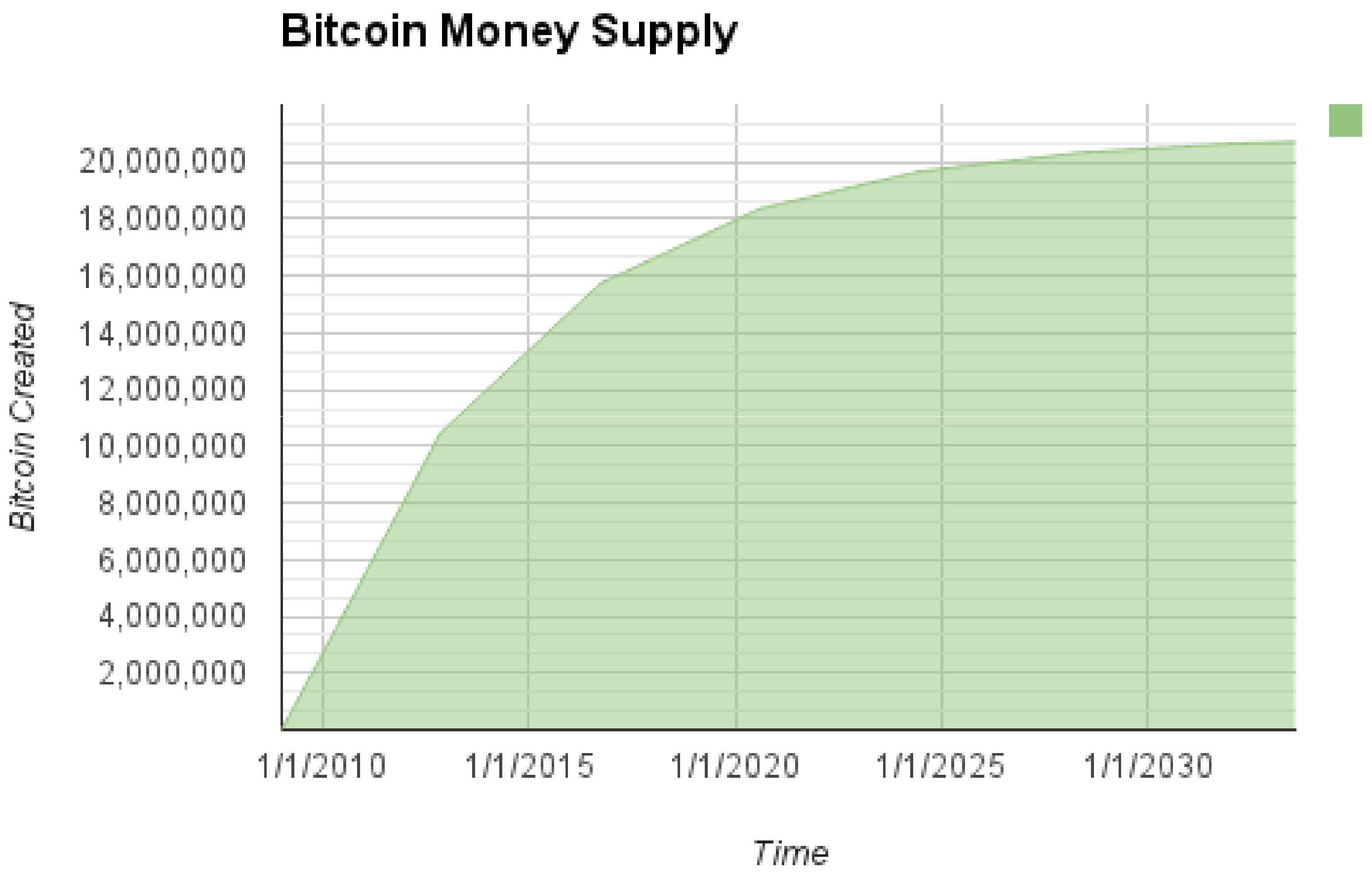
Mining new bitcoin

- New bitcoin are created during the creation of each block at a fixed and diminishing rate, approx. every 10 minutes.

- Every 210,000 blocks (ca. four years), the currency issuance rate is decreased by 50%
  - 2009-2012: 50 new bitcoin earnt per block
  - November 2012: 25 new bitcoin per block
  - July 2016: 12.5 bitcoin per block
  - ca. 2020: 6.25 bitcoin at block 630,000
  - ca. 2137: 1 satoshi per block (block 6,720,000) (99% of all BTC)
  - ca. 2140: After 6.93 million blocks a total of almost 2,099,999,997,690,000 satoshis (almost 21 million bitcoin).

- After that, payment to miners will only be via transaction fees.

Reward for mining is new Bitcoin



The Generation Transaction (Coinbase reward)

- The bitcoin earnt by mining are awarded via the first transaction of each new block
  - The Generation (or Coinbase) transaction

- There are no UTXO inputs for these transactions

- Generation transactions do not have an unlocking script (since there is no UTXO). So the field can have arbitrary content:
  - Eg, Satoshi Nakamoto on 03-01-2009 added to the genesis block:

“The Times 03/Jan/2009 Chancellor on brink of second bailout for banks”.

Format for the Block Header

Size	Field	Description
4 bytes	Version	Software/protocol version
32 bytes	Previous Block Hash	Reference to previous (parent) block
32 bytes	Merkle Root	Hash of root of merkle tree of these transactions
4 bytes	Timestamp	Creation time of block (seconds from Unix Epoch)
4 bytes	Target	PoW algorithm target for this block
4 bytes	Nonce	Counter used for Proof-of-Work algorithm

Mining problem

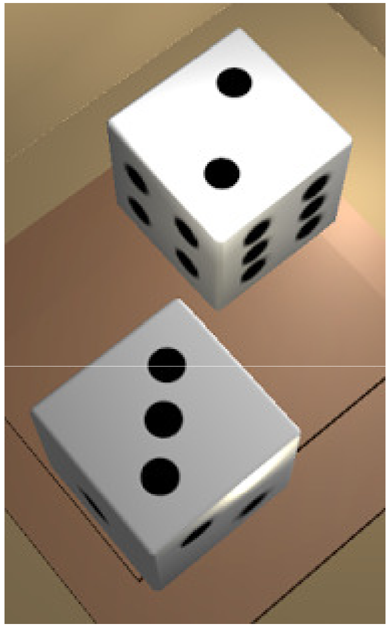
- Proof-of-Work is designed to create a hurdle to mining
  - Otherwise, nodes would spin-up multiple sock-puppet nodes to win the reward
  - A form of Sybil attack

- The problems get harder over time
  - To ensure that a new block is created about every 10 minutes.

- Problem: Find the hash a specified object with a nonce parameter which is less than sum pre-specified total.
  - Problem designed to be hard to do and easy to check.
  - Can only be solved by trial and error.

Two die example

When throwing two die (dices), how many possible outcomes are there when the total is less than a specified number?



- How many outcomes less than 12 in total
- How many outcomes less than 11 in total
- How many outcomes less than 10 in total

. . . .

- How many outcomes less than 3 in total?
- How many outcomes less than 2 in total?
- How many outcomes less than 1 in total?

Sum of two dice throws

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

How many outcomes less than 12 in total: 35 out of 36

How many outcomes less than 11 in total: 33 out of 36

How many outcomes less than 10 in total: 30 out of 36

How many outcomes less than 9 in total: 26 out of 36

. . . . .

How many outcomes less than 3 in total: 1 out of 36

Example of iterating nonce parameter

```
I am Satoshi Nakamoto0 => a80a81401765c8eddee25df36728d732...
I am Satoshi Nakamoto1 => f7bc9a6304a4647bb41241a677b5345f...
I am Satoshi Nakamoto2 => ea758a8134b115298a1583ffb80ae629...
I am Satoshi Nakamoto3 => bfa9779618ff072c903d773de30c99bd...
I am Satoshi Nakamoto4 => bce8564de9a83c18c31944a66bde992f...
I am Satoshi Nakamoto5 => eb362c3cf3479be0a97a20163589038e...
I am Satoshi Nakamoto6 => 4a2fd48e3be420d0d28e202360cfbaba...
I am Satoshi Nakamoto7 => 790b5a1349a5f2b909bf74d0d166b17a...
I am Satoshi Nakamoto8 => 702c45e5b15aa54b625d68dd947f1597...
I am Satoshi Nakamoto9 => 7007cf7dd40f5e933cd89fff5b791ff0...
I am Satoshi Nakamoto10 => c2f38c81992f4614206a21537bd634a...
I am Satoshi Nakamoto11 => 7045da6ed8a914690f087690e1e8d66...
I am Satoshi Nakamoto12 => 60f01db30c1a0d4cbce2b4b22e88b9b...
I am Satoshi Nakamoto13 => 0ebc56d59a34f5082aaef3d66b37a66...
I am Satoshi Nakamoto14 => 27ead1ca85da66981fd9da01a8c6816...
I am Satoshi Nakamoto15 => 394809fb809c5f83ce97ab554a2812c...
```

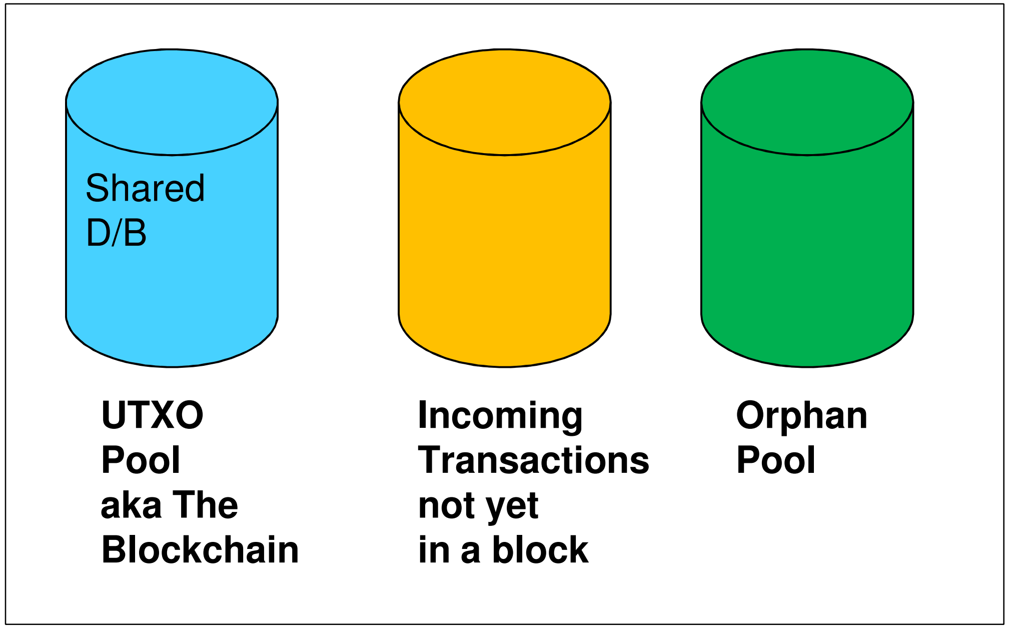
Intending miners

- When new block arrives, they tackle the next PoW problem
- Meanwhile, they assemble transactions that are not in a block into a candidate block
  - Prioritized by age (how many blocks since the UTXO was recorded) &
  - Size of transaction

- High priority:
  - 1 Bitcoin, aged 1 day

- As new blocks added, unused TXs increase in age

- When miner is restarted, its TX pool is wiped.



Shared D/B

UTXO Pool aka The Blockchain

Incoming Transactions not yet in a block

Orphan Pool

Four parts of decentalized consensus: C & D

C Independent verification of the new blocks by every node and assembly into a chain

D Independent selection, by every node, of the chain with the most cumulative computation demonstrated through Proof-of-Work.

- We can reference blocks by their height (about 614,000), or the hash of their header.
  - Block height may not be unique (if there is a fork).
- Block hash is not stored within the block
  - It is calculated by each node as the block is received.

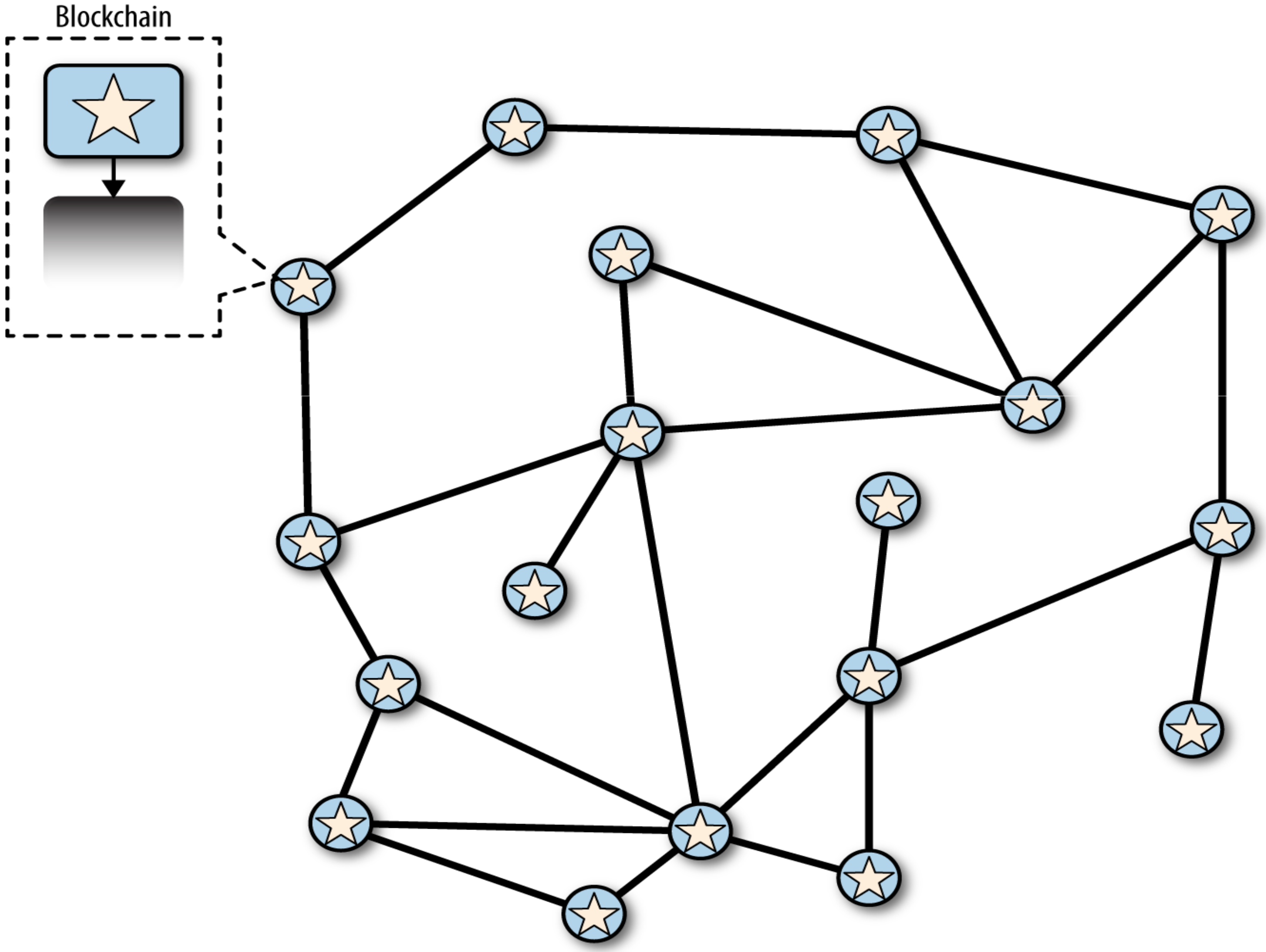
Validating a new block

Criteria for validation include:

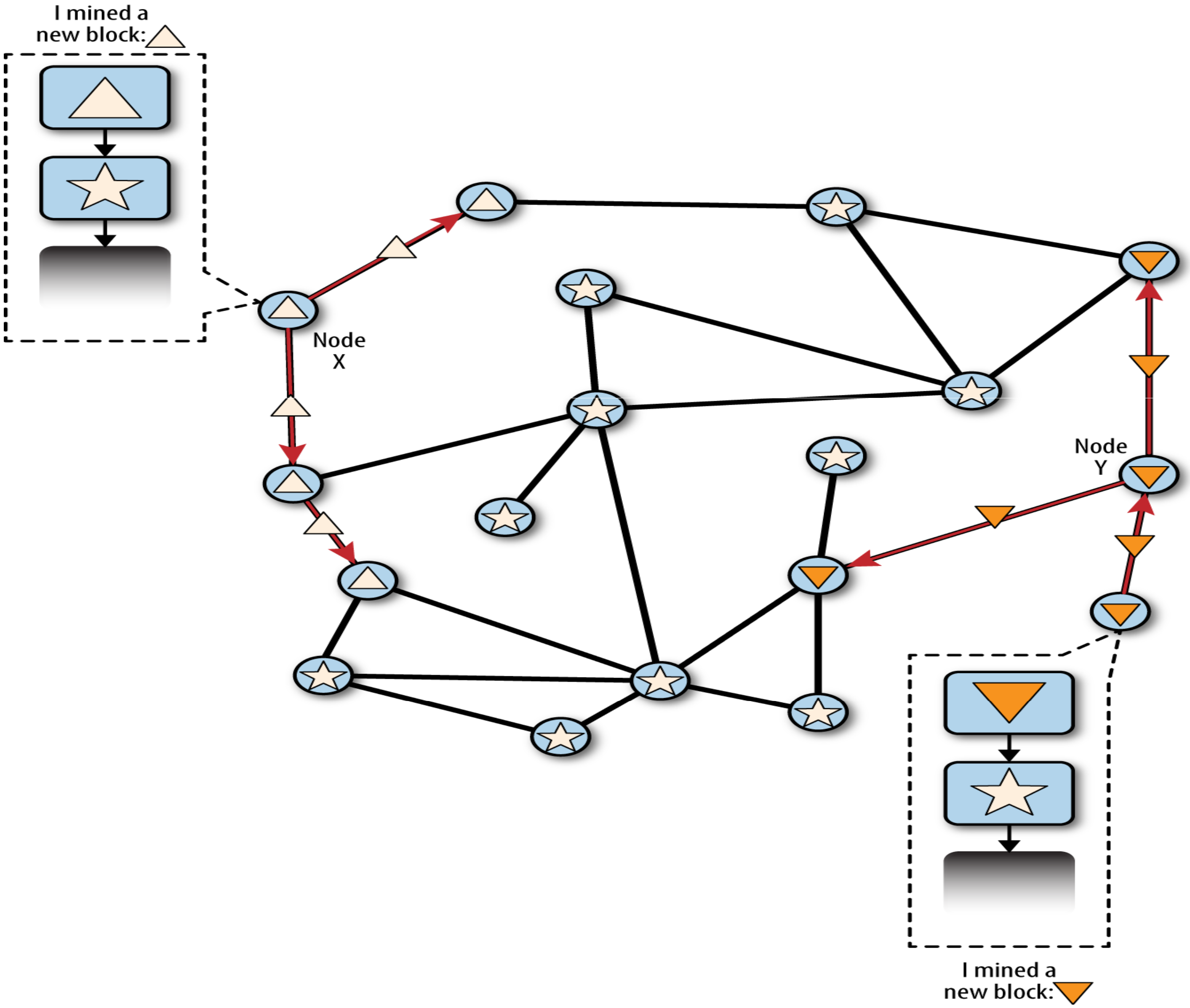
- The block data structure is syntactically valid
- The block header hash is less than the target (enforces Proof-of-Work)
- The block timestamp is less than two hours in the future (allowing for time errors)
- The block size is within acceptable limits
- The first transaction (and only the first) is a coinbase transaction
- All transactions within the block are valid using the transaction checklist for Independent Verification of Transactions.

Blockchain assumes a peer-to-peer (P2P) network

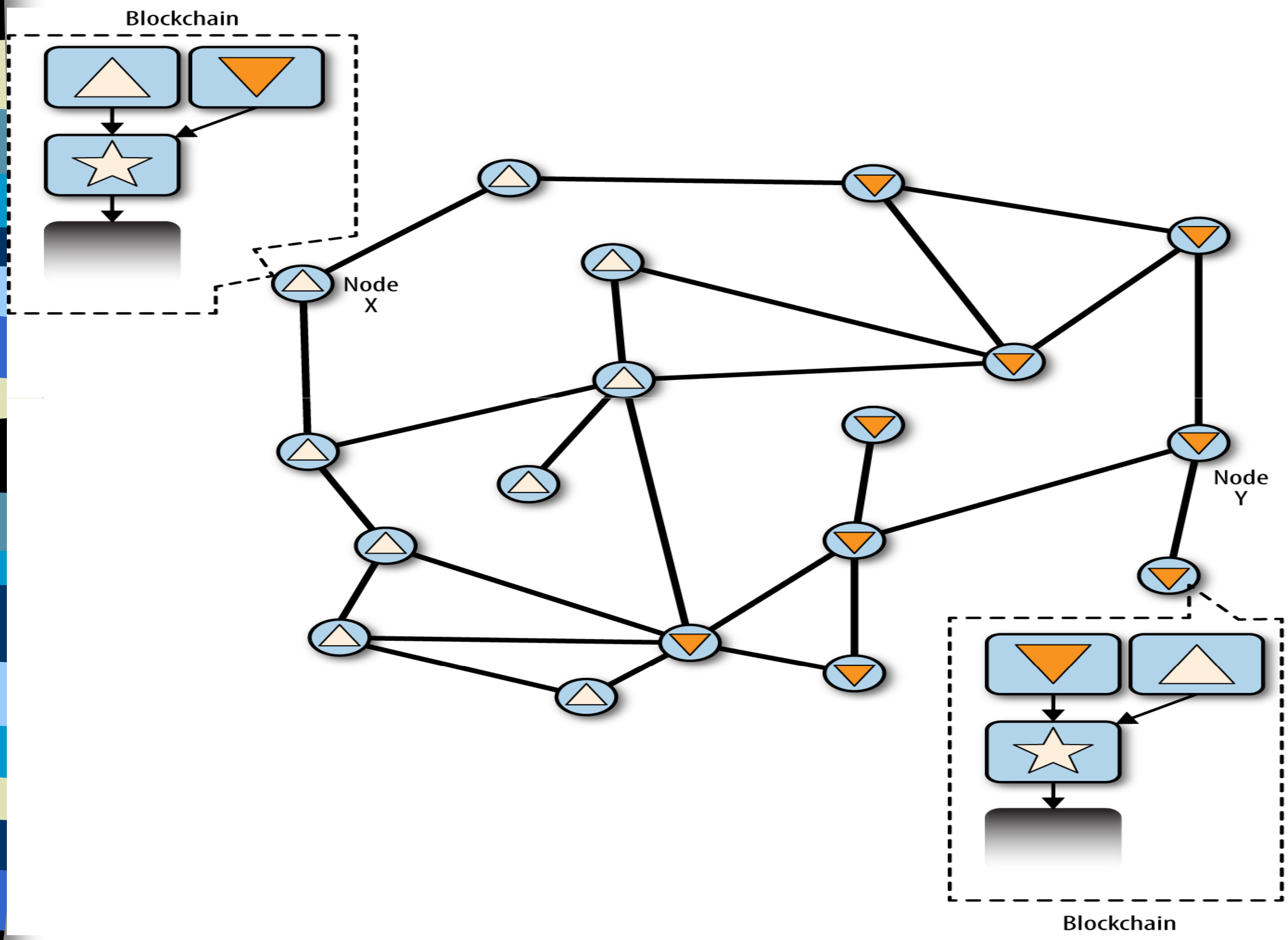
No one is in control.



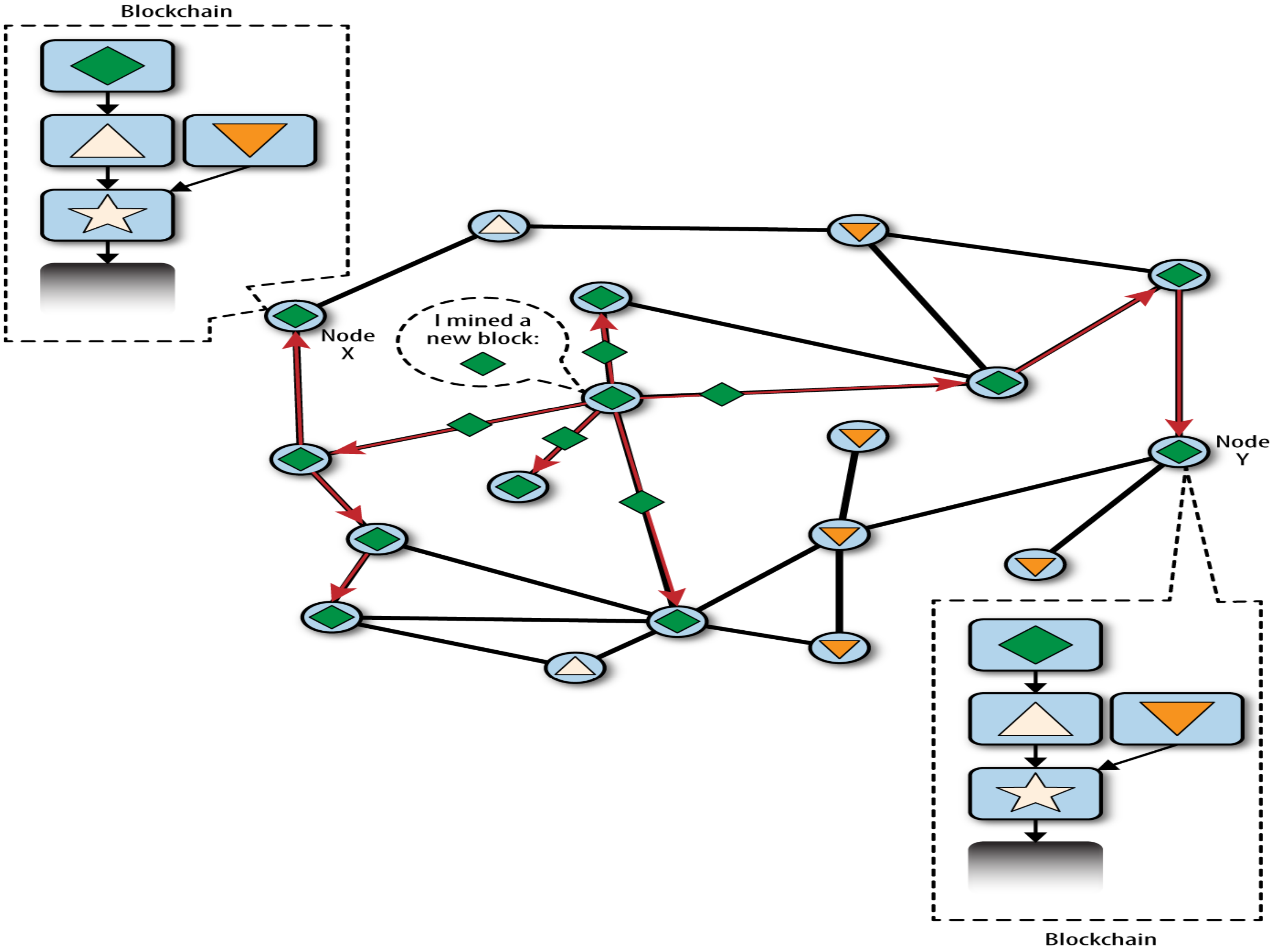
Nodes mine blocks and propagate them locally



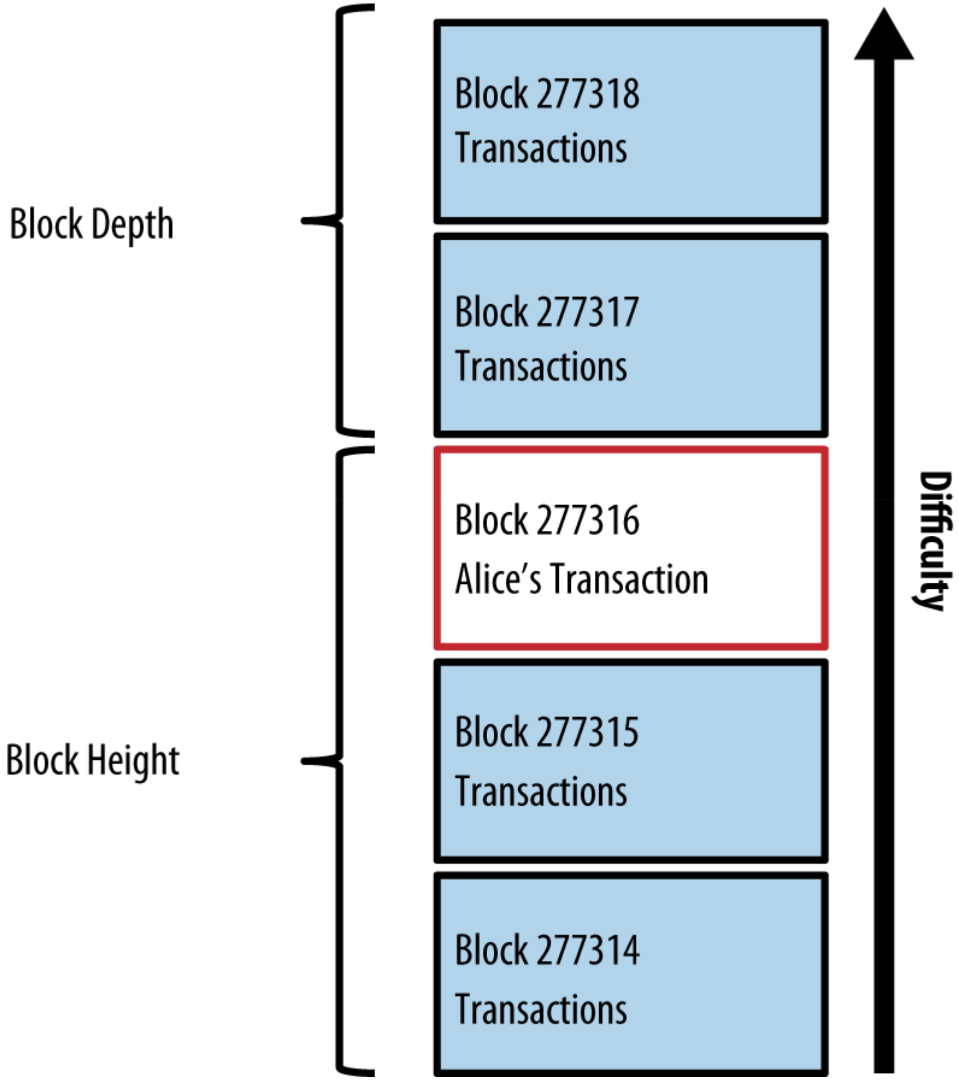
Competing new blocks from different miners



Which chain is "longer" (contains more work)?



Block height currently is about 614,000



https://blockchain.info/q/getblockcount

How do nodes decide between competing blocks?

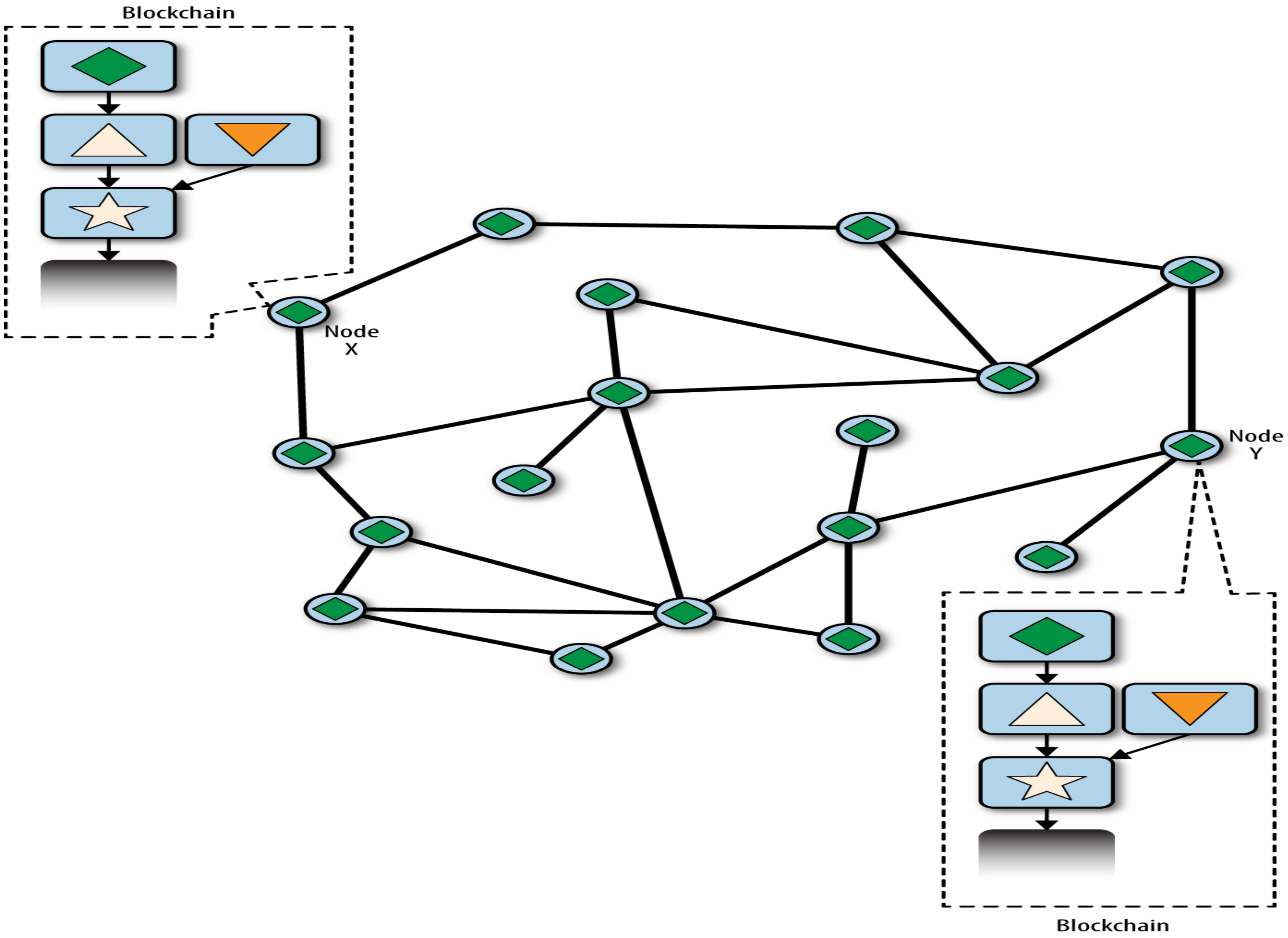
- Nodes keep three collections of blocks
  - Those on the main blockchain
  - Those that form branches off the main blockchain
  - Orphan blocks – those without a parent block

- The main chain is the chain with the most cumulative difficulty associated with it
  - Usually the chain with the most blocks
  - If two chains are equal length, then the main chain is the one with most PoW

- Forks usually resolved within 1 block

- 10 minutes for each block time is a compromise between
  - Fast confirmation times & the probability of a fork.

Eventually consensus is achieved



Thank you!

peter.mcburney@kcl.ac.uk

Exercises

1. List the sequence of events involved in acceptance of new blocks by nodes.
2. Describe the mathematical problems used in Bitcoin for PoW.
3. What is the total maximum number of Bitcoin to be issued? How many have been issued so far? What will miners be paid after the maximum is reached?
4. What is a wallet? What is the difference between wallets held on personal machines versus wallets held on an exchange?
5. List the major Bitcoin exchanges and their country of location. Is there a major exchange which has not been hacked at least once?
6. What are the hash functions SHA256 and RIPEMD160?