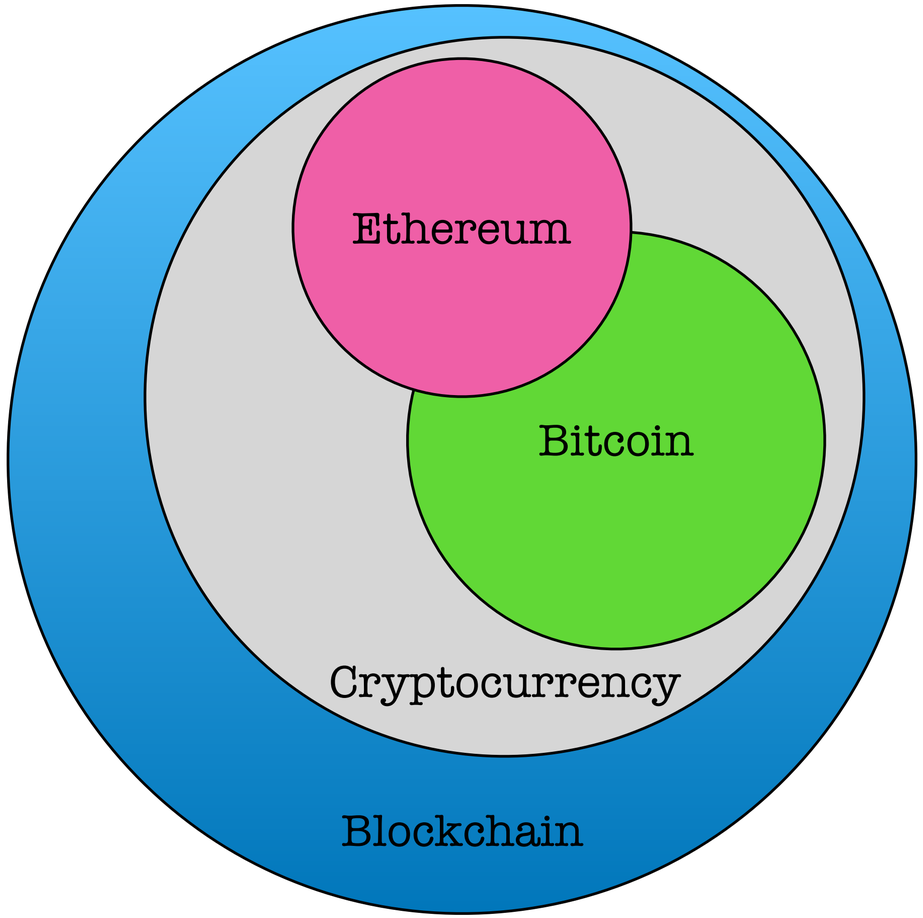
Bitcoin, Cryptocurrency & Blockchain

Russell Sutherland
ITS EIS Network Development

What is Blockchain?

What is Bitcoin?

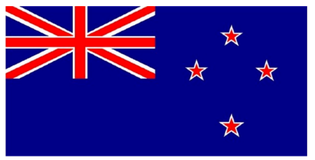

Ethereum
Bitcoin
Cryptocurrency
Blockchain

International Money Transfer

$500

3-4 days

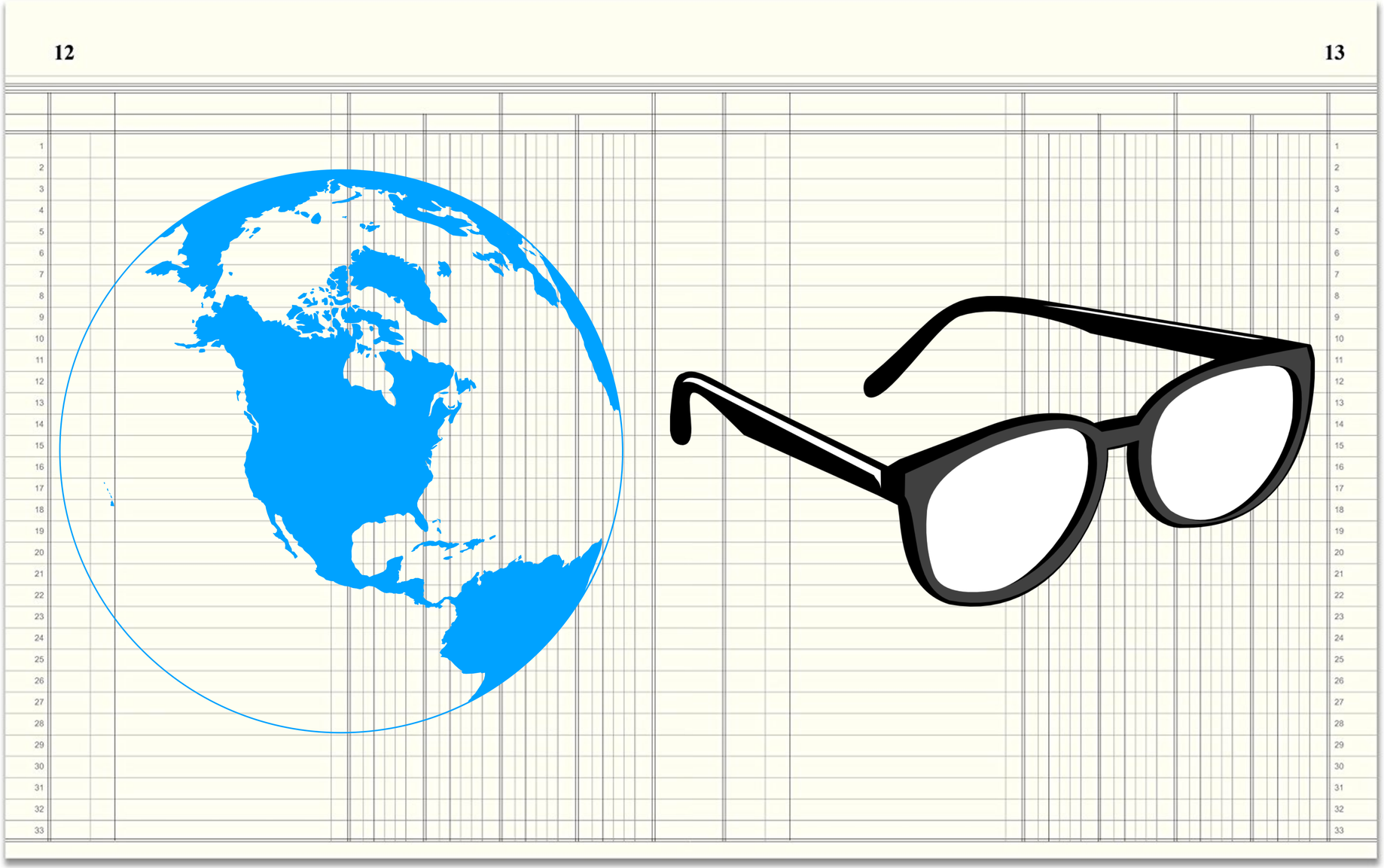
$489



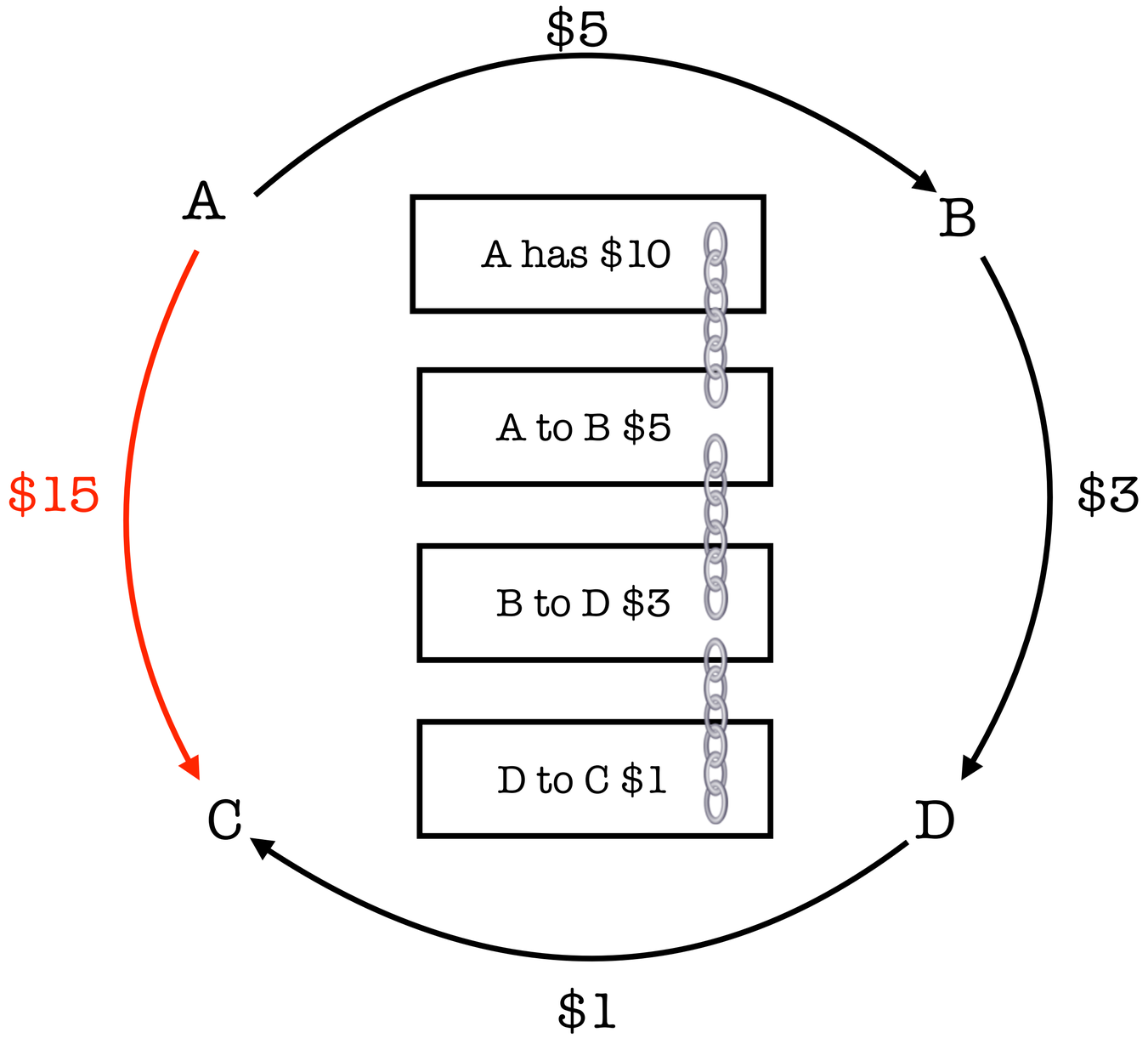

BIG SALE
Merchant
Networks
Daily Deposit
BANK
Merchant Ban
Card Issuing

Ledger

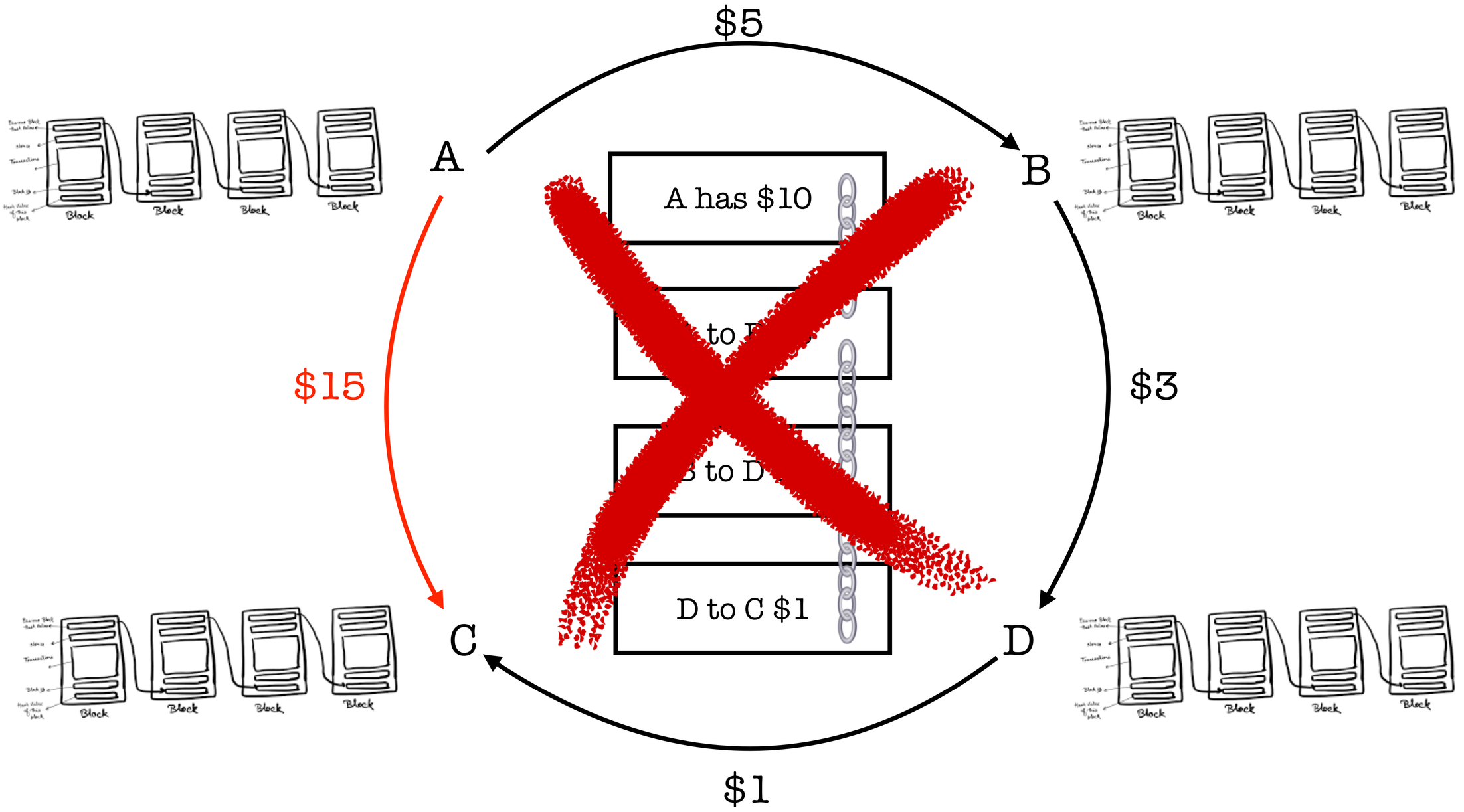
Open Ledger



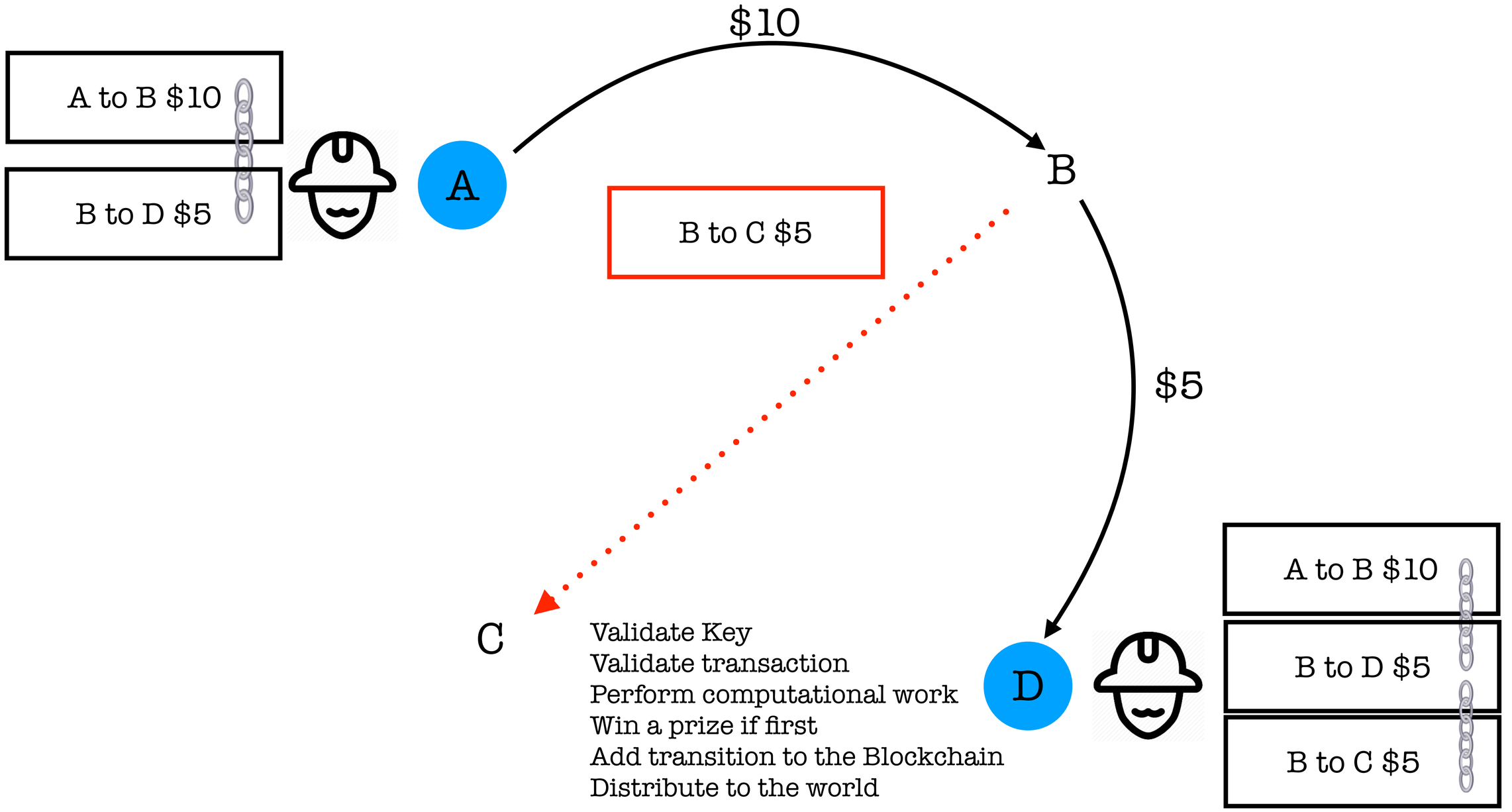
Open Ledger (Chain)



Distributed Open Ledger (Chain)



Synchronized Distributed Open Ledger (Chain)



How does Blockchain work?

Bitcoin as an example

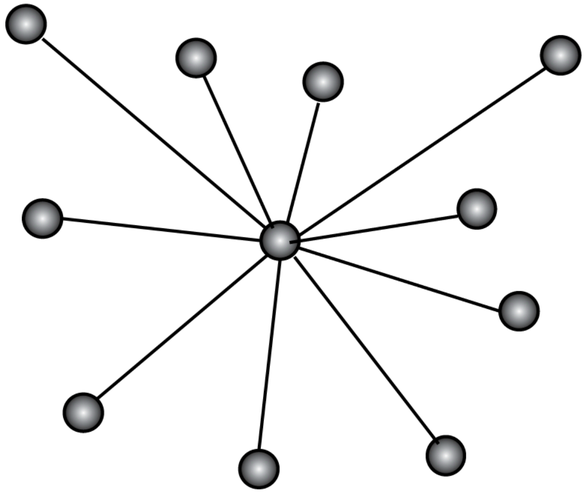
Bitcoin is a digital currency

Currency == Value

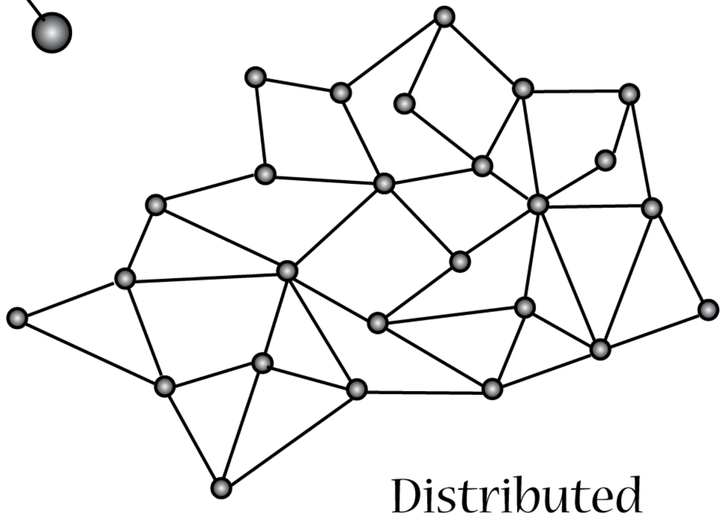
Bitcoin is a network protocol

Distributed Open Ledger (Chain)

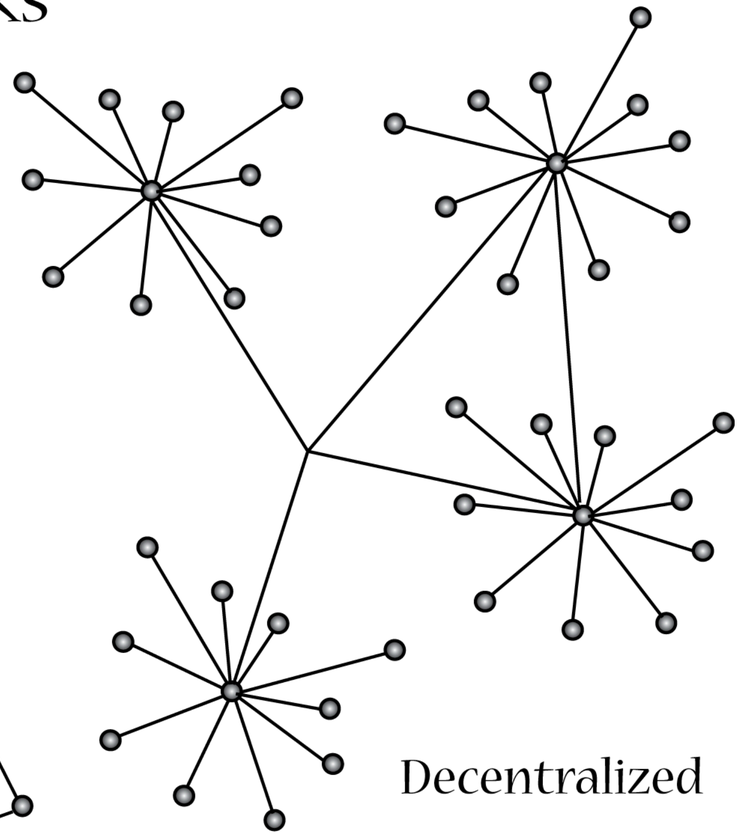
Types of Networks



Centralized



Distributed



Decentralized

Cryptography

Implementation

Hashing Function

Aka Message Digest

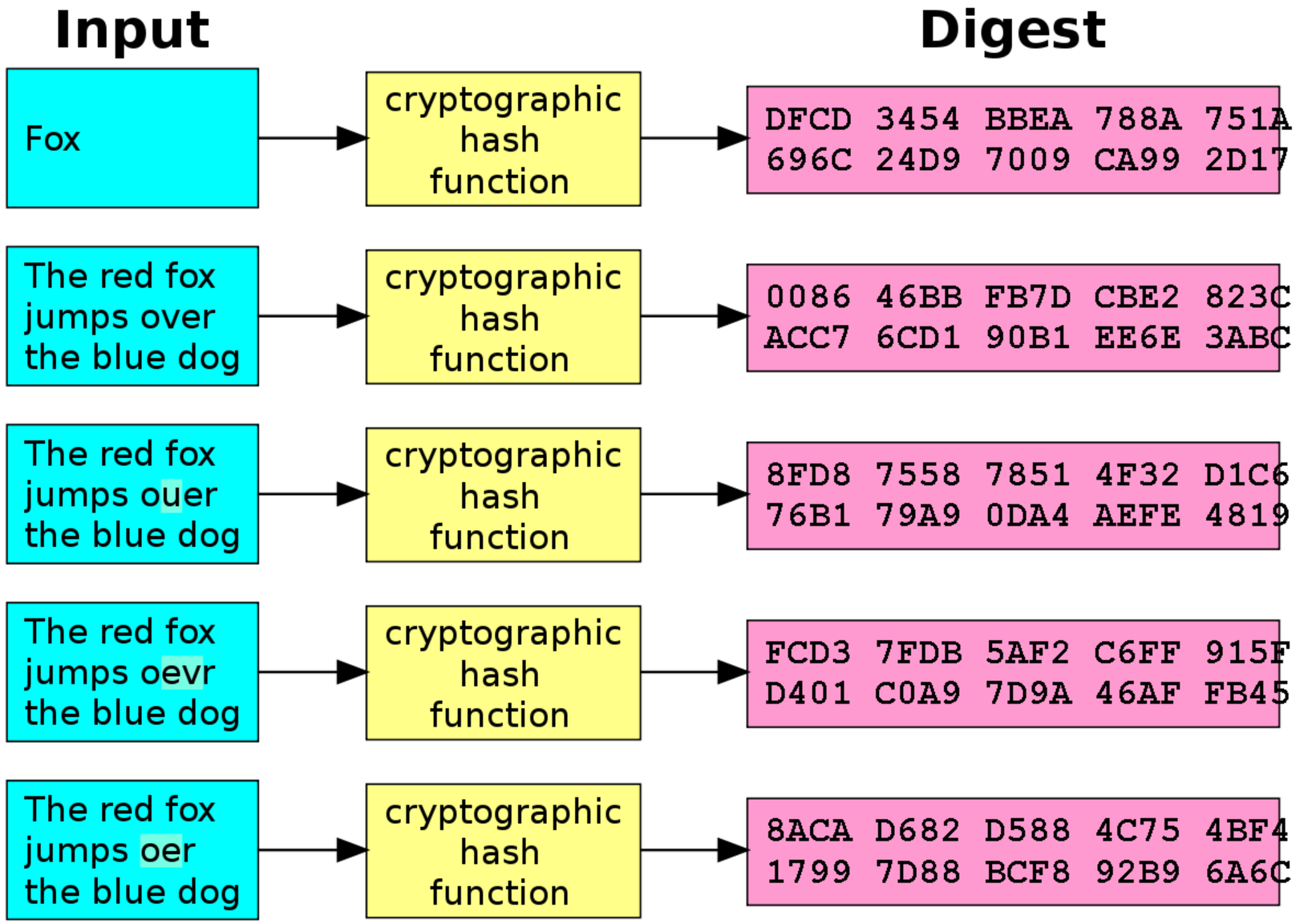
Hashing Function Properties

Takes as variable length data as input
and outputs data of a fixed length

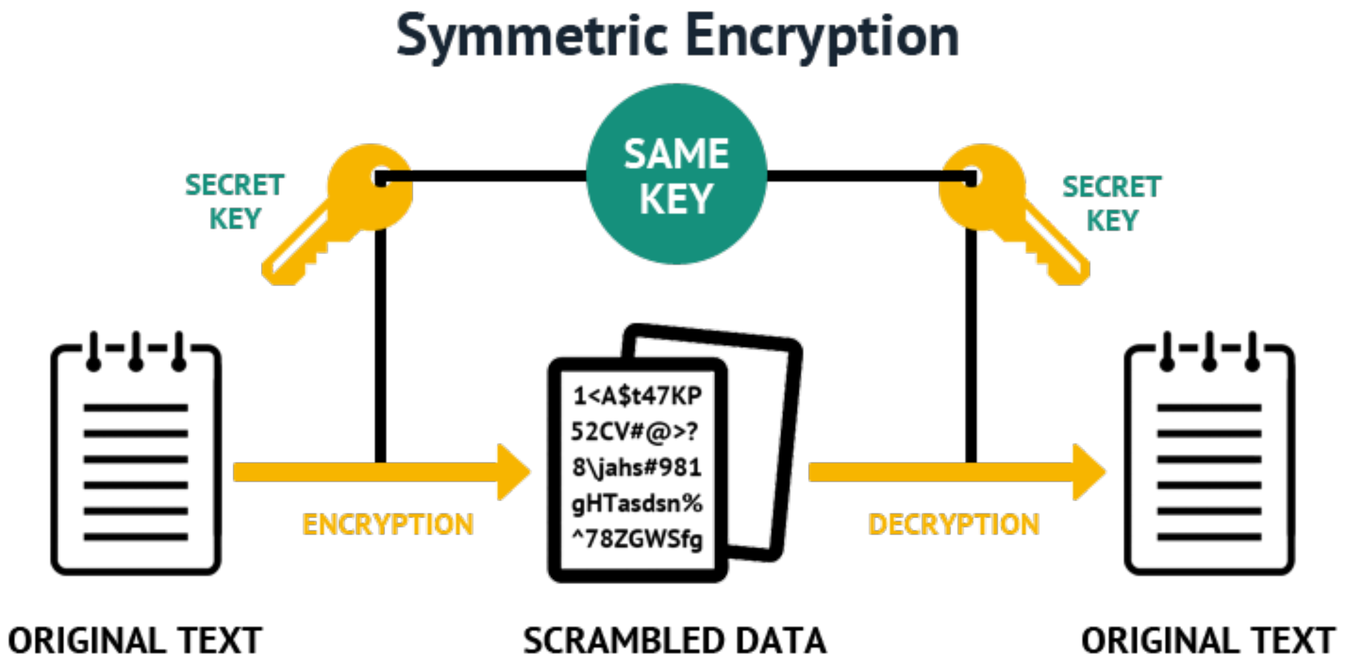
For any given input, there is only one unique output

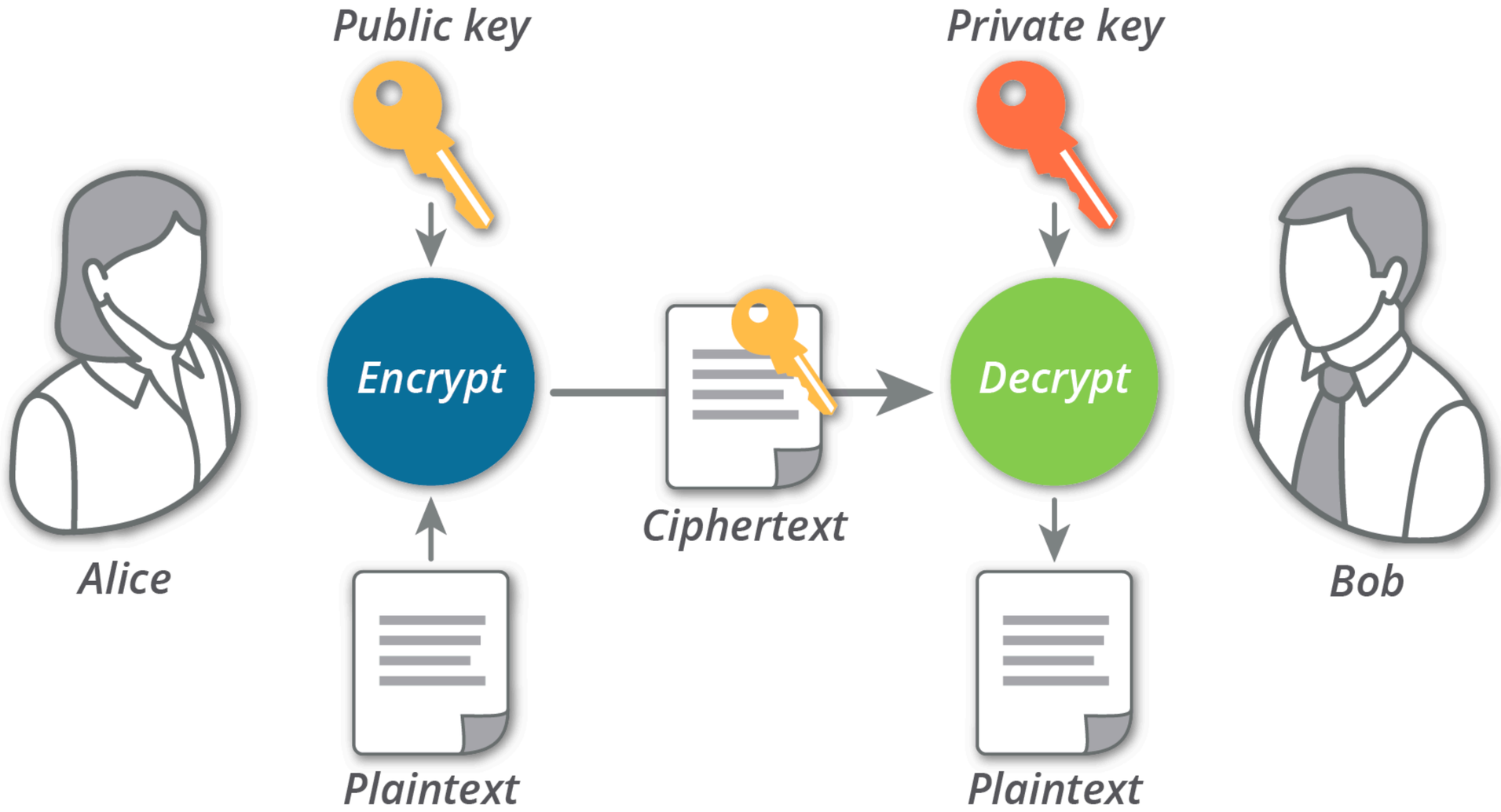
Given an output it is impossible* to find the corresponding input

* computationally impossible

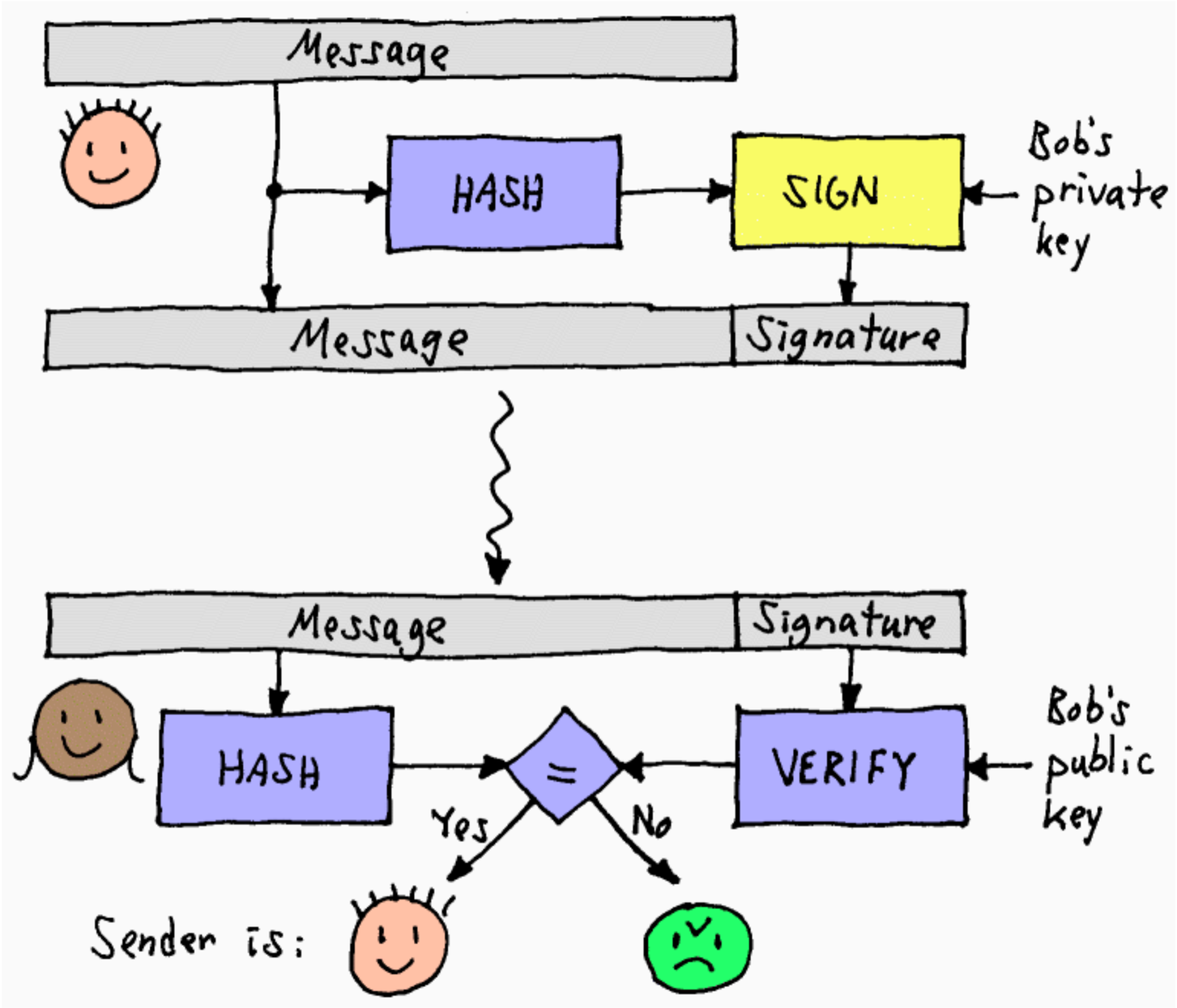

Input
Digest
Fox
cryptographic hash function
DFCD 3454 BBEA 788A 751A
696C 24D9 7009 CA99 2D17
The red fox jumps over the blue dog
cryptographic hash function
0086 46BB FB7D CBE2 823C
ACC7 6CD1 90B1 EE6E 3ABC
The red fox jumps ouer the blue dog
cryptographic hash function
8FD8 7558 7851 4F32 D1C6
76B1 79A9 0DA4 AEFE 4819
The red fox jumps oevr the blue dog
cryptographic hash function
FCD3 7FDB 5AF2 C6FF 915F
D401 C0A9 7D9A 46AF FB45
The red fox jumps oer the blue dog
cryptographic hash function
8ACA D682 D588 4C75 4BF4
1799 7D88 BCF8 92B9 6A6C

Encryption


Symmetric Encryption
SAME KEY
SECRET KEY
SECRET KEY
1<A$t47KP
52CV#@>?
8\jahs#981
gHTasdsn%
^78ZGWSfg
ENCRYPTION
DECRYPTION
ORIGINAL TEXT
SCRAMBLED DATA
ORIGINAL TEXT

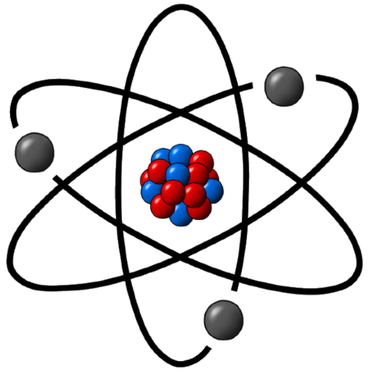

Public key
Private key
Encrypt
Decrypt
Ciphertext
Alice
Bob
Plaintext
Plaintext

Digital Signature

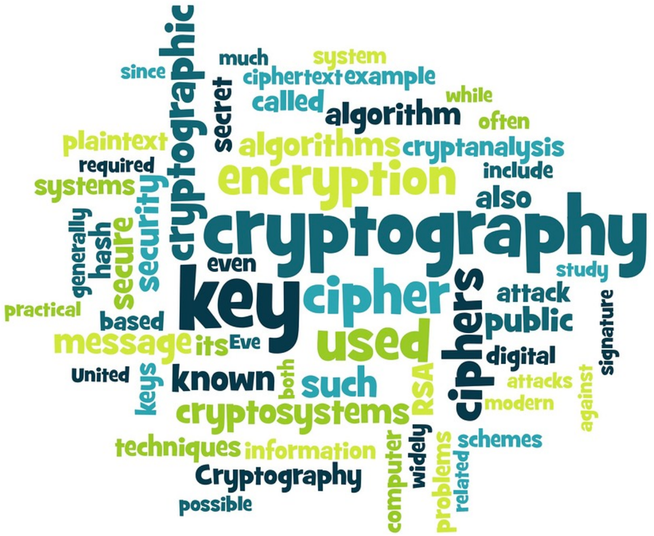

Message
HASH
SIGN
Bob's private key
Message
Signature
Message
Signature
HASH
=
VERIFY
Bob's public key
Yes
No
Sender is:

Let’s create a digital currency called: ByteCoin

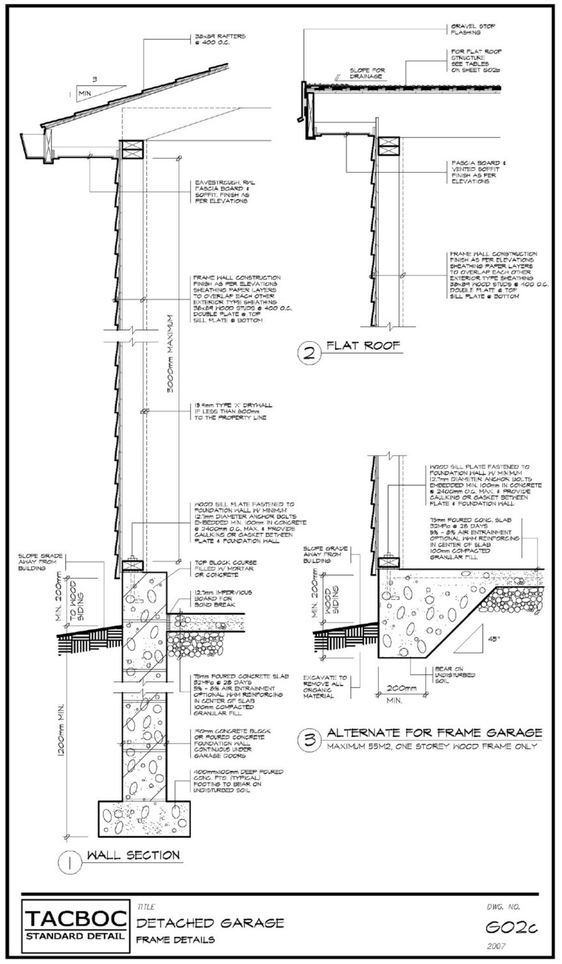
Security

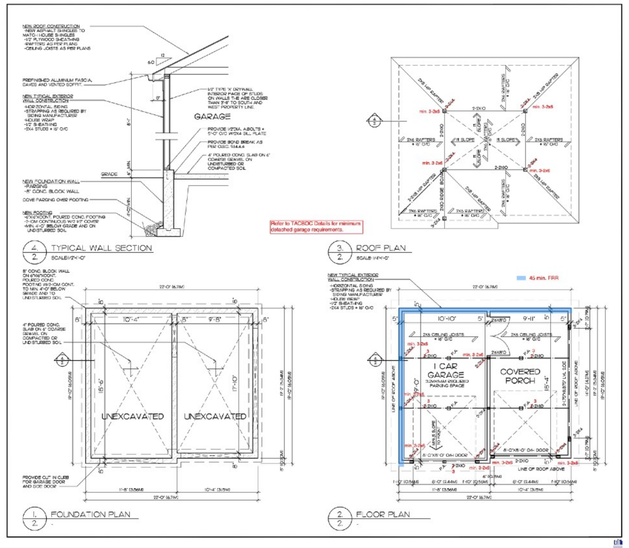


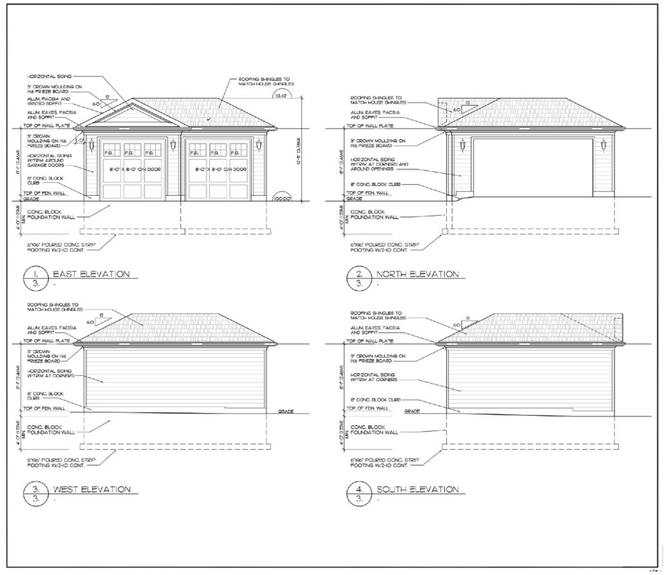
Security


cryptographic
much
system
since
ciphertext
example
secret
called
algorithm
while
often
plaintext
algorithms
cryptanalysis
required
include
systems
encryption
also
security
cryptography
generally
hash
secure
even
study
key
cipher
attack
practical
based
ciphers
public
message
its
Eve
used
digital
United
known
both
such
attacks
signature
keys
RSA
cryptosystems
modern
against
techniques
information
schemes
Cryptography
computer
widely
problems
related
possible

Build Up the Protocol in Stages



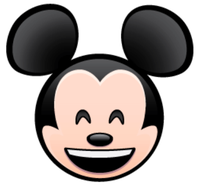




Digital Problems

10100101000101010101010101010101011001

10100101000101010101010101010101011001



Double Spending

Digital Problems

10100101000101010101010101010101011001

10100101000101010101010101010101011001

10100101000101010101010101010101011001

Forgery/Fraud

Let’s use Digital Signatures

“I Alice am giving Mickey one ByteCoin”

Hash + Sign

Private

“I Alice am giving Mickey one ByteCoin” + 101010101010110

Digital Signature

101010101010110
Digital Signature

Public

10101

“I Alice am giving Mickey one ByteCoin” + Hash

10101

If the two hashes match, Alice has shown her intent to give one ByteCoin to Mickey

Establishes the intent of Alice to give one ByteCoin to Mickey

Limited protection from forgery *

*Others can duplicate her message after the fact

Let’s use serial numbers to make each Bytecoin uniquely identifiable

Assume Mickey receives the following:

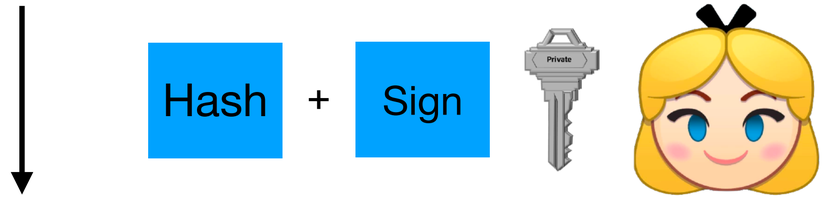
“I Alice am giving Mickey one ByteCoin” + 101010101010110

“I Alice am giving Mickey one ByteCoin” + 101010101010110

Does that mean Alice wants to send a total of 2 Bytecoins?

Or has her message been accidentally duplicated?

“I Alice am giving Mickey one ByteCoin with SN **7763112**”



Private

“I Alice am giving Mickey one ByteCoin with SN **7763112**” + 1010101010110

Digital Signature

101010101010110
Digital Signature

Public

10111

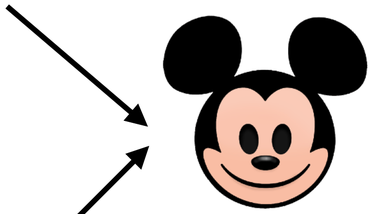
“I Alice am giving Mickey one ByteCoin with SN **7763112**” + Hash

10111

If hashes match Mickey knows Alice with intent wants to give him one Bytecoin with Serial Number **7763112**

“I Alice am giving Mickey one ByteCoin with SN **7763112**”

“I Alice am giving Mickey one ByteCoin with SN **9184331**”



Mickey now knows Alice wants to send him 2 different Bytecoins

I.e. this is NOT a case of double or multiple spending

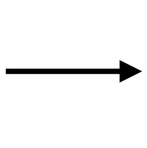
We now need a Bank (a trusted source) which:

* assigns unique labels to Bytecoins

* keeps track of who owns which Bytecoins

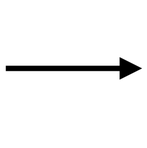
* verify that transactions are legitimate

I would like to use one of my Bytecoins

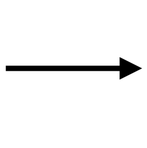


You may use Bytecoin **7651234**

I Alice am giving Mickey one ByteCoin with SN **7651234**



Does Bytecoin **7651234** belong to Alice?
Has Alice spent that coin already?
If legit please put that Bytecoin in my account



Your account has been debited by one Bytecoin
You are no longer the owner of Bytecoin **7651234**

Your account has been credited by one Bytecoin
You are now the owner of Bytecoin **7651234**

Let’s get rid of the Bank as a central authority

Let’s make everyone collectively the Bank

Every person has a copy of the complete Bank ledger

The shared common ledger is called the **Blockchain**


“I Alice am giving Mickey one ByteCoin with SN **7651234**”



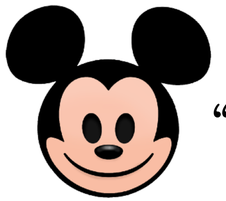
Does Bytecoin **7651234** belong to Alice?
Has Alice spent that coin already?
If OK, I except and will update my Blockchain

+

Bytecoin **7651234**
Alice to Mickey


+

Bytecoin **7651234**
Alice to Mickey



"I Alice am giving Mickey one ByteCoin with SN **7651234**"

"I Alice am giving Darth one ByteCoin with SN **7651234**"

And

Does Bytecoin **7651234** belong to Alice?
Has Alice spent that coin already?
If OK, I except and will update my Blockchain

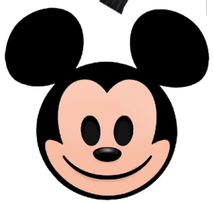
\+ Bytecoin **7651234**
Alice to Mickey

\+ Bytecoin **7651234**
Alice to Darth

The Double Spending Problem

"I Alice am giving Mickey one ByteCoin with SN **7651234**"

"I Alice am giving Darth one ByteCoin with SN **7651234**"



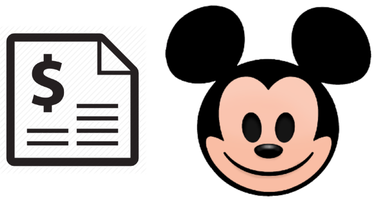
+ Bytecoin **7651234**
Alice to Darth

+ Bytecoin **7651234**
Alice to Mickey

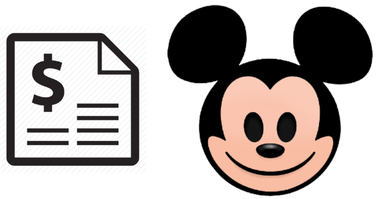
How are the rest of members to update their Blockchains?

How will they keep their Blockchains consistent and synchronized

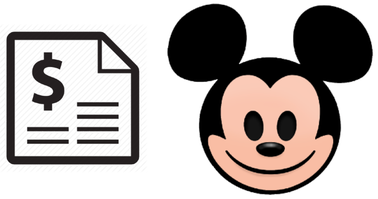
A Double Spending Problem Solution



"I Alice am giving Mickey one ByteCoin with SN **7651234**"



Does Bytecoin **7651234** belong to Alice?
Has Alice spent that coin already?
If OK, I will broadcast this to the world



"I Alice am giving Mickey one ByteCoin with SN **7651234**" →

enough ← Yes, Alice owns Bytecoin 7651234 it may be transferred to Mickey

+

Bytecoin **7651234**
Alice to Mickey

What does enough mean?

Everyone on the network?

A certain fraction of users on the network?

4

Proof of

What does enough mean in terms of concensus?

Greater than 50 percent ?

If Alice wants to double spend and fool the network users
all she has to do is create millions/billions of accounts
which will agree that both of her transactions are legitimate

Proof of

1. Make it computationally costly for users to validate transactions
2. Reward users for trying to help validate transactions

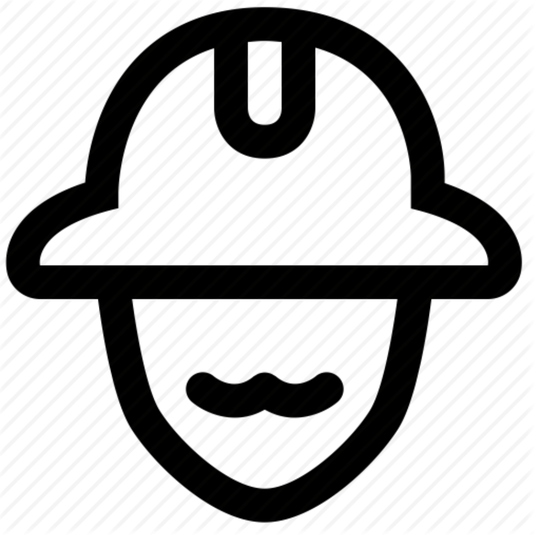
Both these ideas are counterintuitive

Proof of

1. Reward motivates people to help validate transactions
   even though they have to invest considerable resources to do so.

2. Making the validation process costly prevents
   bad/dishonest users attempting to overwhelm
   the network with sheer numbers. I.e. a majority attack

3. Computational power is what matters, not number of users.

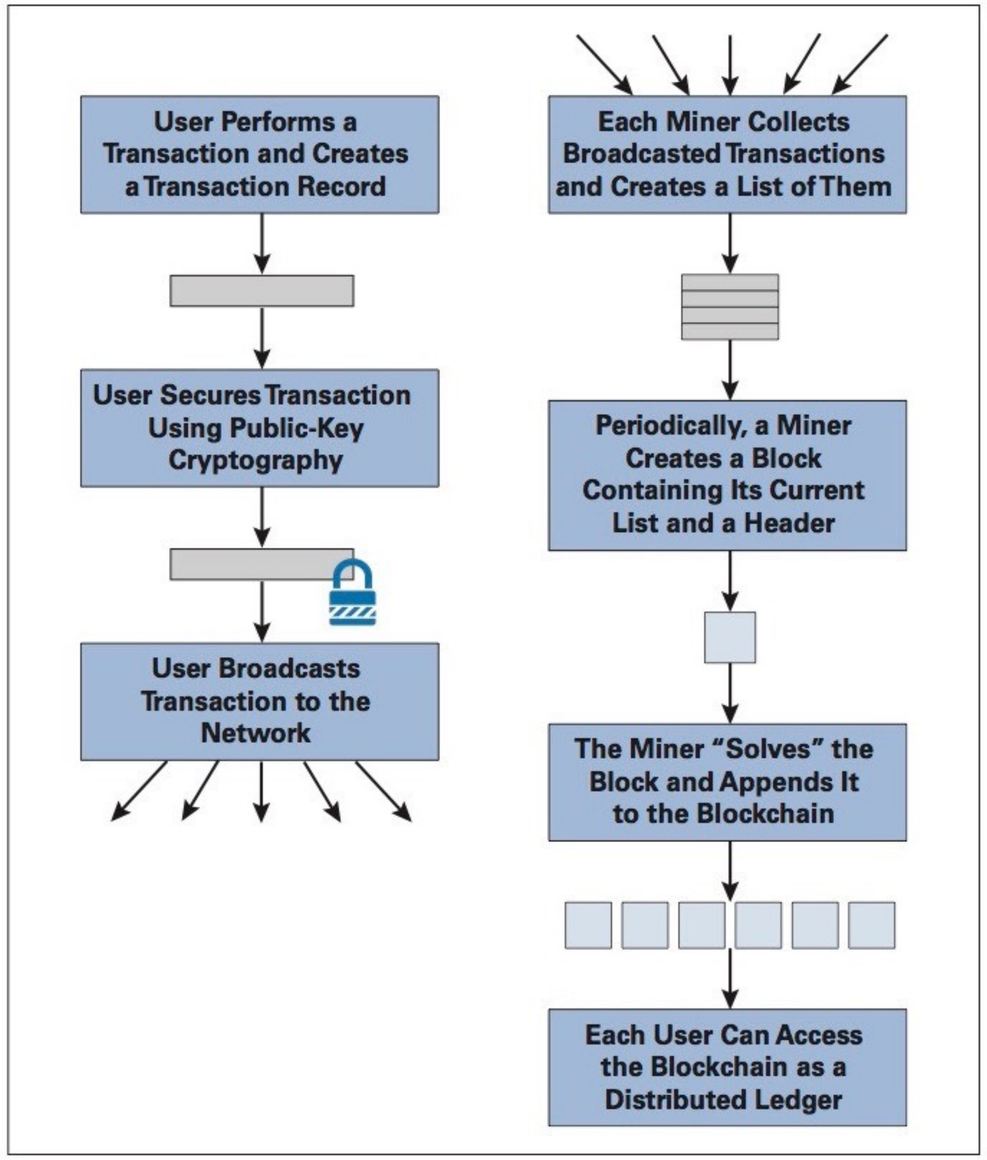


Those who expend computational energy to validate transactions are called Miners.

Successful Miners get a reward if they are the first to solve a mathematical problem.

The reward is in terms of the currency/coins being traded

Basic Blockchain Logic/Protocol



Mark the Miner receives the following 3 transactions:

I Tom am giving Sue one Bytecoin with SN **1201133**

I Sydney am giving Sasha one Bytecoin with SN **4431212**

I Vanya am giving Xu one Bytecoin with SN **5539955**

Mark the Miner combines the transactions with a numerical guess and feeds the data to a hash function (SHA256)

I Tom am giving Sue one Bytecoin with SN **1201133**

+

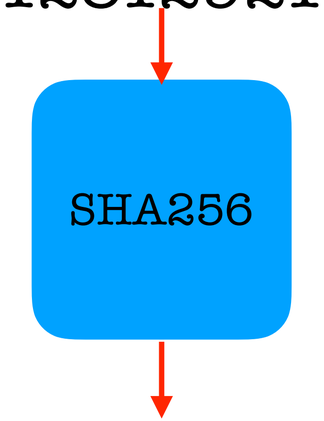
I Sydney am giving Sasha one Bytecoin with SN **4431212**

+

I Vanya am giving Xu one Bytecoin with SN **5539955**

+

12312921



2f7aed9cc58e75f3173b22d4ff952c59ba879744429ba1cf8cf9ad9e6e37e59d

Mark the Miner combines the transactions with a numerical guess and feeds the data to a hash function (SHA256)

I Tom am giving Sue one Bytecoin with SN **1201133**

+

I Sydney am giving Sasha one Bytecoin with SN **4431212**

+

I Vanya am giving Xu one Bytecoin with SN **5539955**

+

453

cdfba543ee8ef7fdb3d8b587648cc22dd792bbd6272cc5447307c7c106c2374c

Mark the Miner combines the transactions with a numerical guess and feeds the data to a hash function (SHA256)

I Tom am giving Sue one Bytecoin with SN **1201133**

+

I Sydney am giving Sasha one Bytecoin with SN **4431212**

+

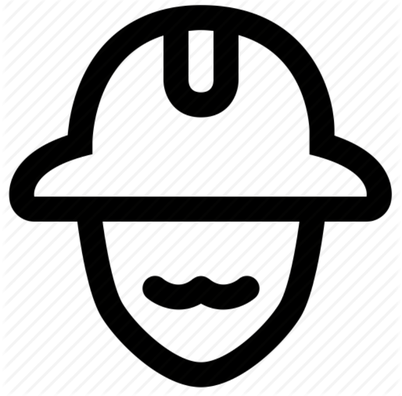
I Vanya am giving Xu one Bytecoin with SN **5539955**

+

1211

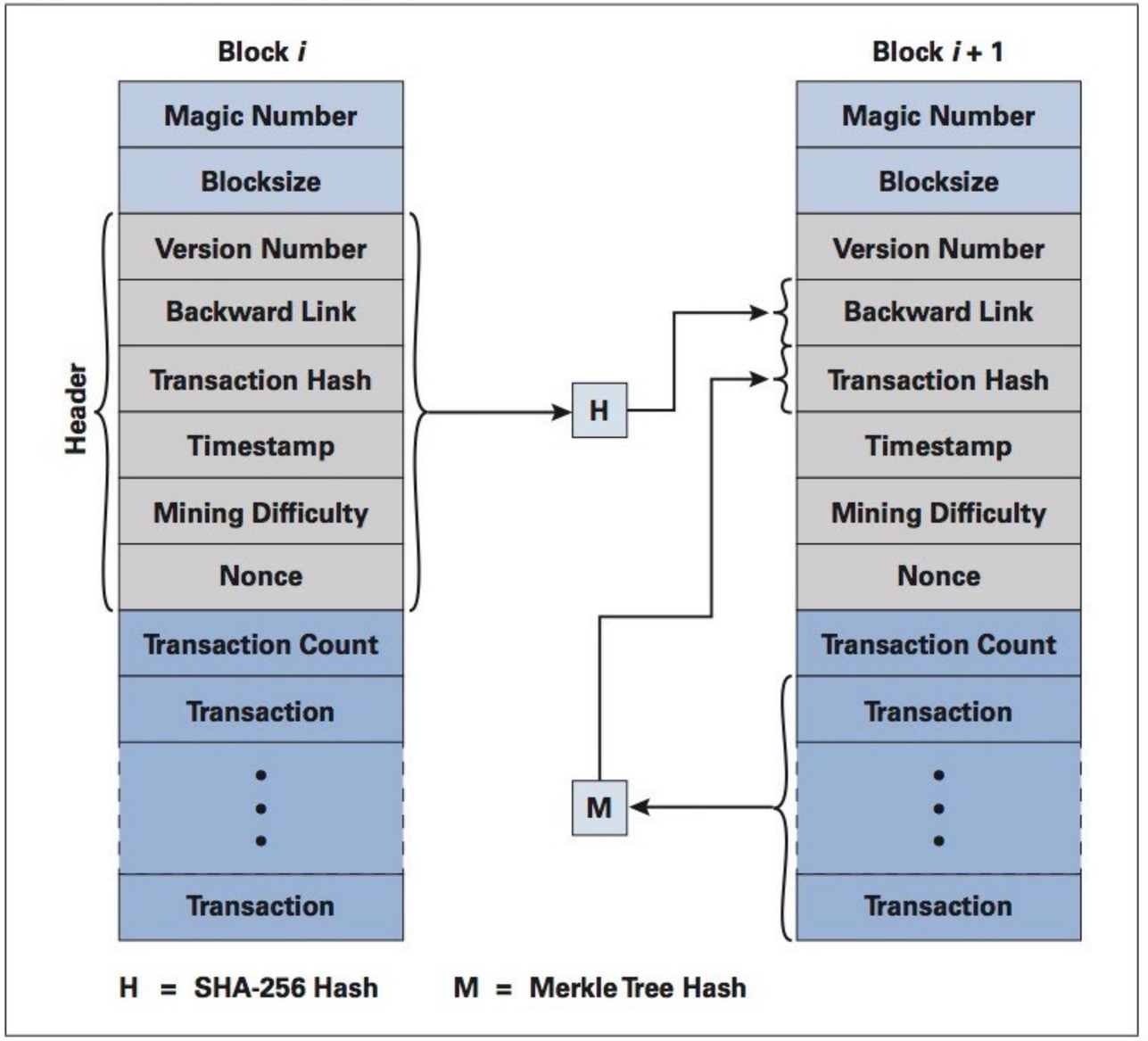
SHA256

2ecefe59ab6ae4e734bb1adec3bbb706be8c6fae0b39500d05f7a6665ffa7390



Mark the Miner wins when the output of his hash is less than a given target

Bitcoin Blockchain Structure



TRANSACTIONS PER DAY

The number of bitcoin transactions in the last 24 hours.

2 1 7 6 0 4

Transactions since Tue May 01 2018 9:00:19 PM.

1 BTC = $9,195.83

Interactive Chart →

MARKET CAP: $154,368,868,774.00

HASH RATE: 31,590,148.24 TH/s

Hash Rate == 3.2 x 10^21 = 3 200 000 000 000 000 000 000

GLOBAL BITCOIN NODES DISTRIBUTION

Reachable nodes as of Wed May 02 2018 20:55:39 GMT-0400 (EDT).

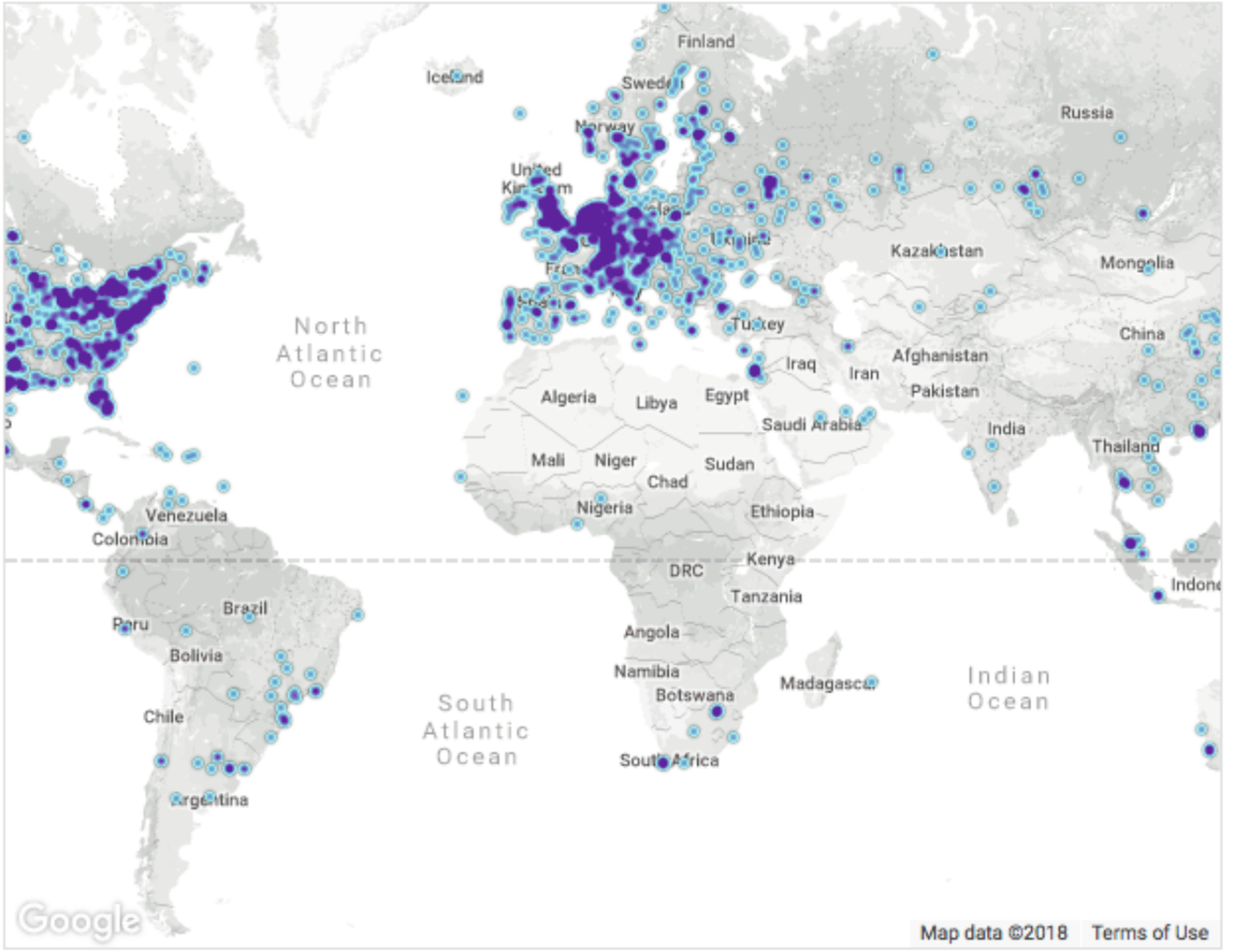
10431 NODES

24-hour charts »

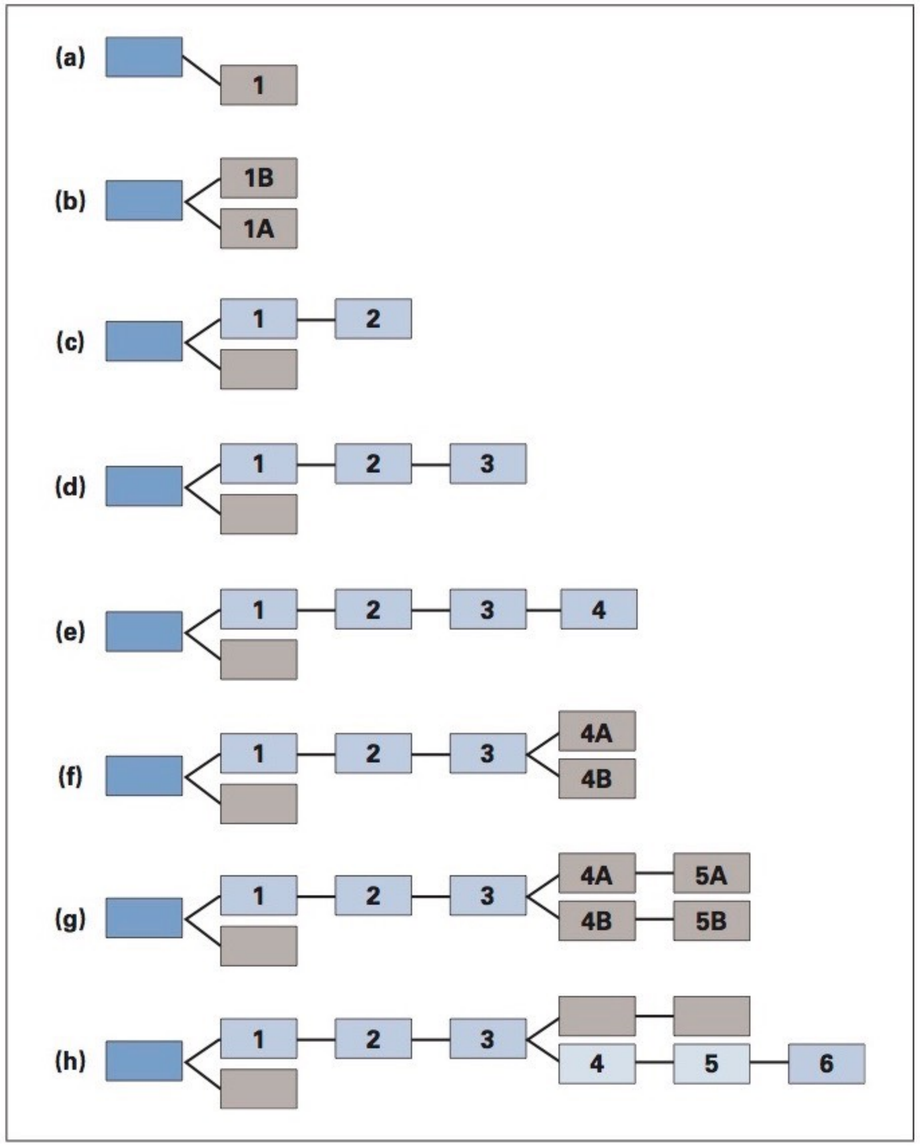
Top 10 countries with their respective number of reachable nodes are as follow.

RANK	COUNTRY	NODES
1	United States	2603 (24.95%)
2	Germany	1976 (18.94%)
3	China	744 (7.13%)
4	France	679 (6.51%)
5	Netherlands	484 (4.64%)
6	Canada	376 (3.60%)
7	United Kingdom	358 (3.43%)
8	Russian Federation	343 (3.29%)
9	n/a	342 (3.28%)
10	Japan	225 (2.16%)

More (104) »



Adding Blocks to the Chain



Miners will always select the longest chain

If two Miners solve a block at the same time the block will fork into two

In such cases some Miners work on A and some on B. Which ever team solves their block first, becomes the new block as it will be the longest

Double Spending Fraud vs. Proof of Work

Case 1:

Alice is a fraudster and runs 1% of the global hashing rate.
She tries to solve a block with her double spending transactions:

I Alice am giving **Mickey** one ByteCoin, SN **2255991**
I Alice am giving **Darth** one ByteCoin, SN **2255991**

Assuming she solves a block first (1% chance of doing so), when she broadcasts her validated block the rest of the Miners agree the hash value is correct by reject the block as it contains conflicting transactions. #FAIL

Double Spending Fraud vs. Proof of Work

Case 2:

Alice is a fraudster and runs 1% of the global hashing rate.
She sends one transaction to 1/2 of the Miners and 1/2 to the others
She hopes to get both confirmed and added to the blockchain.

"I Alice am giving **Mickey** one ByteCoin, SN **2255991**" to **A** Miners
"I Alice am giving **Darth** one ByteCoin, SN **2255991"** to **B** Miners

The network will solve either one transaction or the other,
not both depending upon which group of Miners solve a block first.
Darth or Mickey will see their transaction not confirmed and ignore it.
#FAIL

Double Spending Fraud vs. Proof of Work

Case 3:

Alice is a fraudster and runs 1% of the global hashing rate.
Darth is one of her aliases. She wants to effect send money to herself.
She sends one transaction to Mickey and waits until 6 blocks are added, and Mickey then accepts her coin. She then creates a block sending a coin to Darth (herself) hoping to outrun the longest chain.

"I Alice am giving **Mickey** one ByteCoin, SN **2255991**" to **A** Miners

"I Alice am giving **Darth** one ByteCoin, SN **2255991"** to **B** Miners

With the entire rest of the network working on the longer chain, her chance of catching up is 1 in a trillion
#FAIL

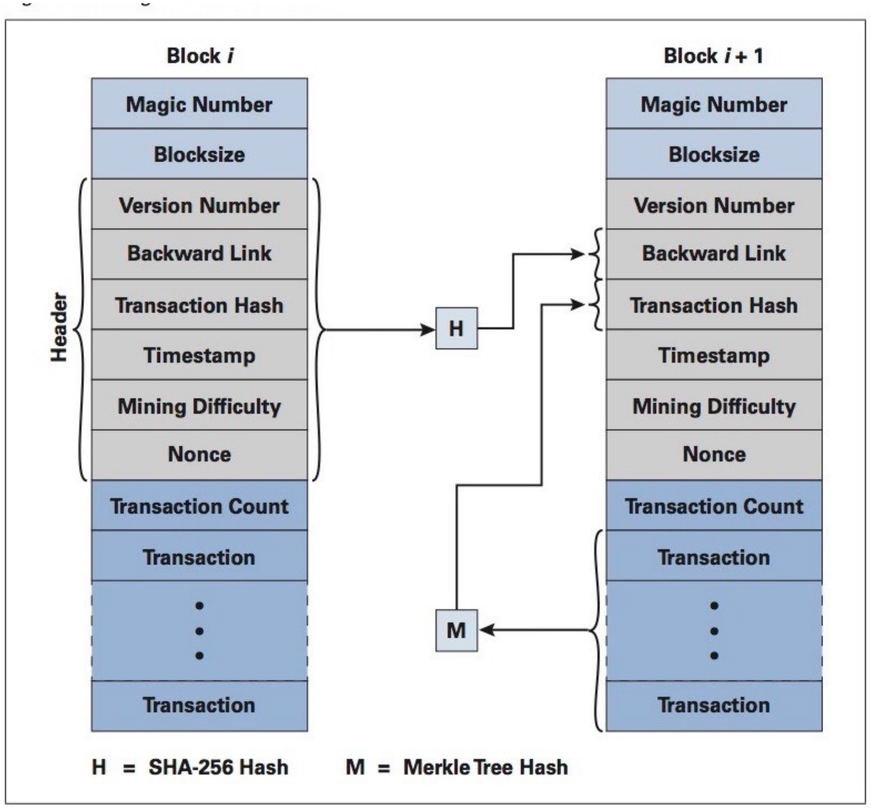
Changing an existing Block vs. Proof of Work

Case 1a:

Alice is a fraudster and runs 1% of the global hashing rate. She tries to modify the transaction list in an existing block.

It is computationally infeasible to modify the transaction list and leave both the Transaction Hash and Header Hash (Backward Link) unchanged, given SHA256 as a hashing function.

#FAIL



2^256 ~ 10^77

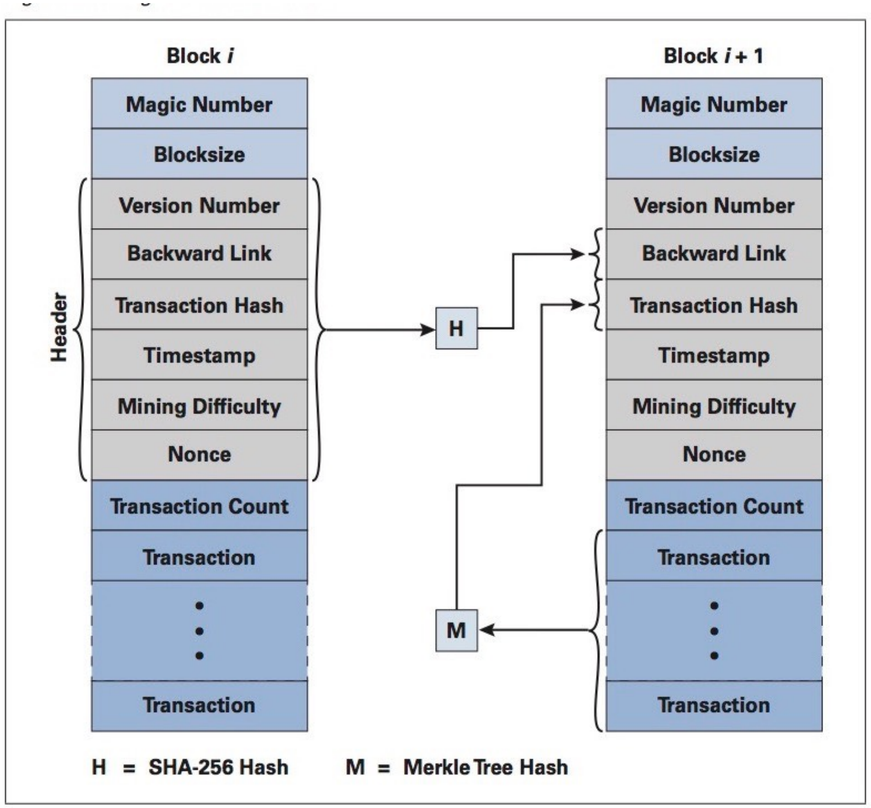
Changing an existing Block vs. Proof of Work

Case 1b:

Alice is a fraudster and runs 1% of the global hashing rate. She tries to modify the transaction list in an existing block.

It is computationally infeasible to modify the transaction list (allowing the Transaction Hash to change) and chose a new Nonce to leave the Header Hash (Backward Link) unchanged, given SHA256 as a hashing function.

#FAIL



2^256 ~ 10^77

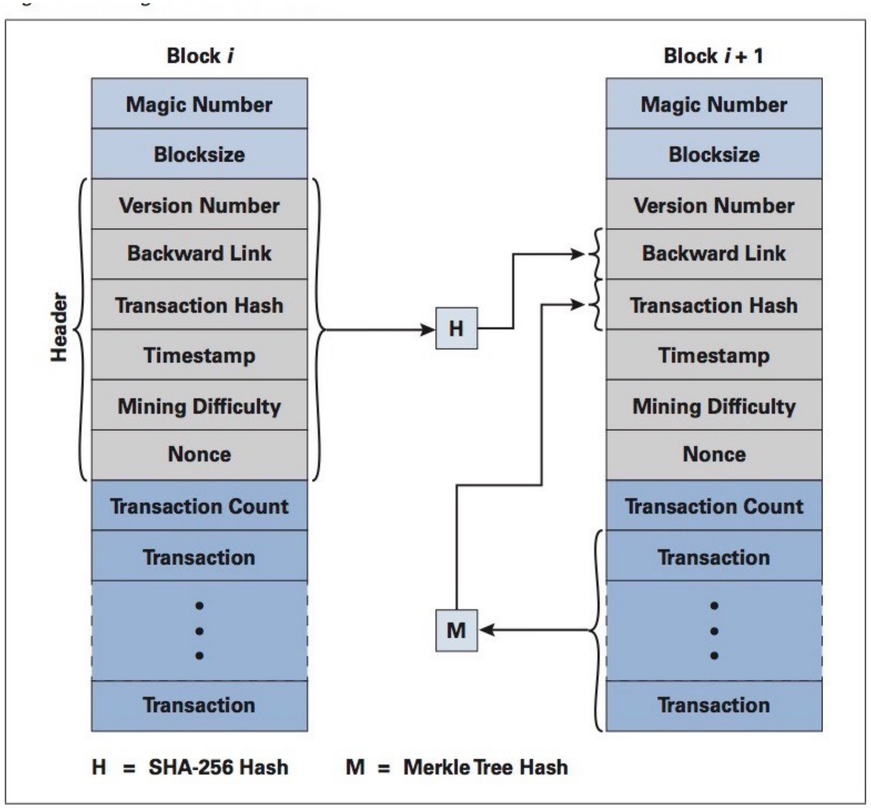
Inserting a new Block vs. Proof of Work

Case 2:

Alice is a fraudster and runs 1% of the global hashing rate. She tries to insert a new Block into the Blockchain.

It is computationally infeasible to find a Nonce such that the Header Hash equals that of the following Block, given SHA256 as a hashing function.

#FAIL



2^256 ~ 10^77