RETIREMENT ASSOCIATION

A Primer on Blockchain:
The Infrastructure for Digital Assets

Brad Chandler
Director, Nicholas Center for Corporate Finance and Investment Banking
January 25, 2022

Our Agenda

1. Bitcoin: The First Public Blockchain
2. Ethereum: The First Multi-Purpose Public Blockchain
3. Private Blockchains
4. Key Takeaways

Section 1:

Bitcoin: The First Public Blockchain

Bitcoin: The First Public Blockchain

1. Design
2. Implementation
3. Evaluation

1. Bitcoin Design: The Origin Story

1970-2008

- Key advances in cryptography date back to the 1970s
  - Public key encryption (1970s / 1980s)
  - Blind signatures for untraceable payments (1980s)
- Key predecessors: DigiCash (1989, Chaum) and Hashcash (Back, 1997), among others
- Blockchain concept dates back to Bayer, Haber and Stornetta (1991)

2009

- Cryptocurrencies arrived on the scene in 2009
- Bitcoin = first public blockchain and first enduring cryptocurrency
- Developed by Satoshi Nakamoto (pseudonym) in response to the Wall Street bailout in 2008

1. Bitcoin Design:
Bitcoin Arrived in 2009 With A Distrust in Centralized Institutions

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Bitcoin Genesis Block
Raw Hex Version Dated 1/3/2009

```
00000000  01 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000010  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000020  00 00 00 00 3B A3 ED FD  7A 7B 12 B2 7A C7 2C 3E  ....;£íýz{.²zÇ,>
00000030  67 76 8F 61 7F C8 1B C3  88 8A 51 32 3A 9F B8 AA  gv.a.È.Ã^ŠQ2:Ÿ¸ª
00000040  4B 1E 5E 4A 29 AB 5F 49  FF FF 00 1D 1D AC 2B 7C  K.^J)«_Iÿÿ...¬+|
00000050  01 01 00 00 00 01 00 00  00 00 00 00 00 00 00 00  ................
00000060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00000070  00 00 00 00 00 00 FF FF  FF FF 4D 04 FF FF 00 1D  ......ÿÿÿÿM.ÿÿ..
00000080  01 04 45 54 68 65 20 54  69 6D 65 73 20 30 33 2F  ..EThe Times 03/
00000090  4A 61 6E 2F 32 30 30 39  20 43 68 61 6E 63 65 6C  Jan/2009 Chancel
000000A0  6C 6F 72 20 6F 6E 20 62  72 69 6E 6B 20 6F 66 20  lor on brink of
000000B0  73 65 63 6F 6E 64 20 62  61 69 6C 6F 75 74 20 66  second bailout f
000000C0  6F 72 20 62 61 6E 6B 73  FF FF FF FF 01 00 F2 05  or banksÿÿÿÿ..ò.
000000D0  2A 01 00 00 00 43 41 04  67 8A FD B0 FE 55 48 27  *....CA.gŠý°þUH'
000000E0  19 67 F1 A6 71 30 B7 10  5C D6 A8 28 E0 39 09 A6  .gñ¦q0·.\Ö¨(à9.¦
000000F0  79 62 E0 EA 1F 61 DE B6  49 F6 BC 3F 4C EF 38 C4  ybàê.aÞ¶I6¼?LI8Ä
00000100  F3 55 04 E5 1E C1 12 DE  5C 38 4D F7 BA 0B 8D 57  óU.å.Á.Þ\8M÷º..W
00000110  8A 4C 70 2B 6B F1 1D 5F  AC 00 00 00 00           ŠLp+kñ._¬....
```


THE TIMES
Chancellor on brink of second bailout for banks

1. Bitcoin Design:
Bitcoin White Paper's Purpose

What Problem Is It Solving?

The inability to send a final (i.e., non-reversible) transaction over the Internet without involving a financial institution (i.e., a trusted third party)

1. Bitcoin Design: Bitcoin White Paper's Purpose

What Problem Is It Solving?

The inability to send a final (i.e., non-reversible) transaction over the Internet without involving a financial institution (i.e., a trusted third party)

Isn't That Already Available?

- In the physical world yes (i.e., with cash)
- But using the Internet requires going through an intermediary (i.e., financial institution, credit card company, payments network)

1. Bitcoin Design: Bitcoin White Paper's Purpose

What Problem Is It Solving?	The inability to send a final (i.e., non-reversible) transaction over the Internet without involving a financial institution (i.e., a trusted third party)
Isn't That Already Available?	• In the physical world yes (i.e., with cash) • But using the Internet requires going through an intermediary (i.e., financial institution, credit card company, payments network)
The Proposed Solution	An **electronic payment** system **based on cryptographic proof of trust**, allowing any two willing parties to transact directly with each other **without the need for a trusted third party**

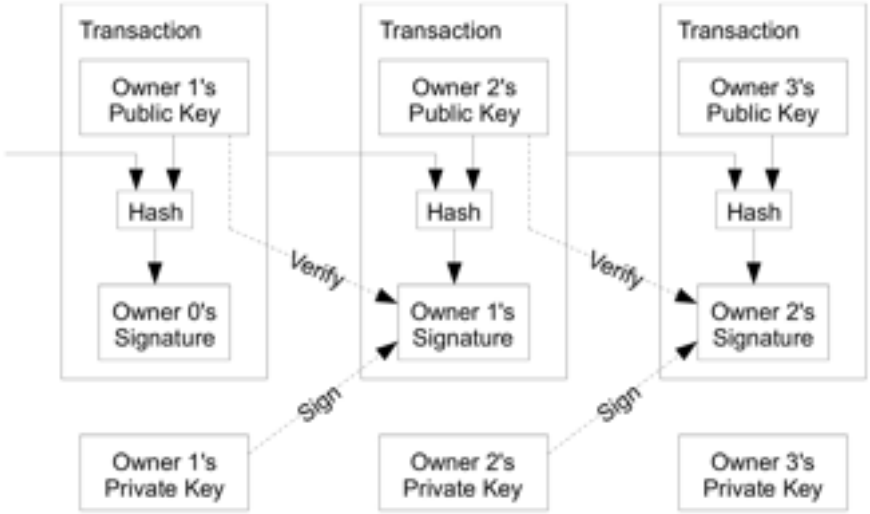
1. Bitcoin Design: The Proposed Solution

"Electronic Coin"

"Electronic coin" is a chain of digital signatures

Transferring Coins

Each owner **transfers the coin** to the next by **digitally signing** a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin



1. Bitcoin Design: The Core Problem

What is the core problem with this proposed solution?

1. Bitcoin Design: The Core Problem

What is the core problem with this proposed solution?

Double spend problem:

The entity receiving the payment can't verify that the current owner did not already spend the coin in another transaction

1. Bitcoin Design: Solution to the Double Spend Problem

Solution to Double Spend Problem (1)

"Peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions"

I can confirm those are words, but what do they mean?

(1) Without the use of a trusted third party

1. Bitcoin Design: Solution to the Double Spend Problem

Solution to Double Spend Problem (1)

"Peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions"

I can confirm those are words, but what do they mean?

Key elements

1. Transactions are publicly announced
2. Need a system for a group of participants to agree on a single history of the order in which transactions were received ("consensus")

(1) Without the use of a trusted third party

2. Bitcoin Implementation

A. Peer-to-Peer Network

B. Secured by Cryptography

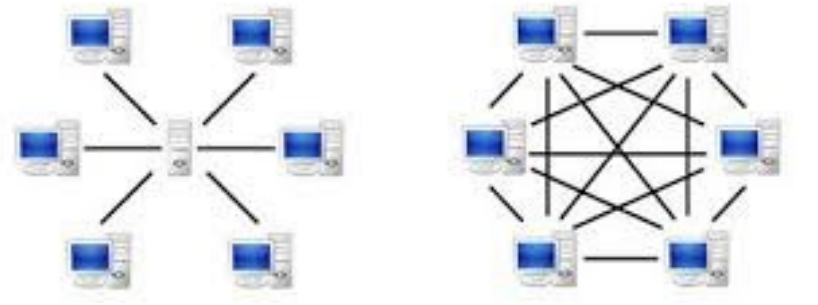
C. Transaction-Based Ledger

D. Ordered Chronologically

E. Consensus

F. Fixed Supply Schedule

2. Bitcoin Implementation:
A. Peer-to-Peer Network

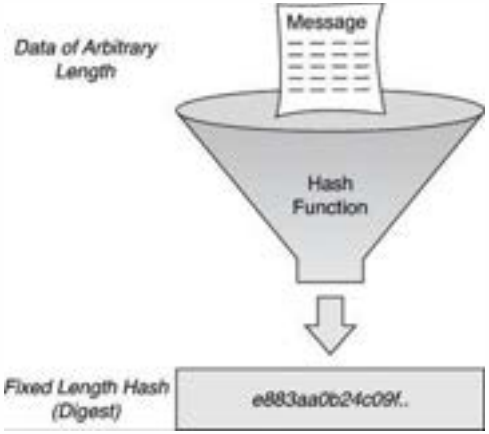


Centralized System **vs** **Peer-to-Peer Network**

- Bitcoin network is permissionless – anyone with computing setup can run a node
- Nodes are equal
  - No centralized connector, no hierarchy
  - Interconnected in a mesh network with a "flat" topology

Note: Some nodes perform different functions from the others, but that is a choice the user makes. The suite of functions include network routing, wallet, miner and full blockchain database. Some nodes choose to be full nodes that perform all functions, while others choose to be lightweight nodes with more limited functionality

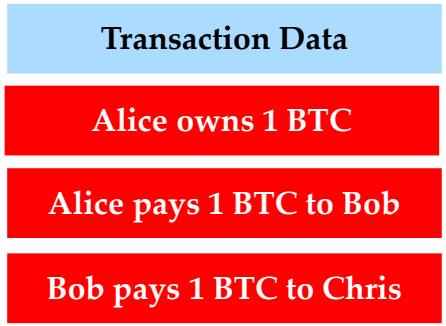
2. Bitcoin Implementation:
B. Secured by Cryptography



SHA-256 Hash Function Example	• Hash for the entire King James Bible: **47f63b8cd8470051acd3a3c0bd5c77c4aa9574d79cf5bfb3e576facabbc11491** • Hash for the King James Bible with 5 characters deleted: **961c112581bd04e67285f56a354c98ad56cd65244dc768545cfde5bd8ef639c1**

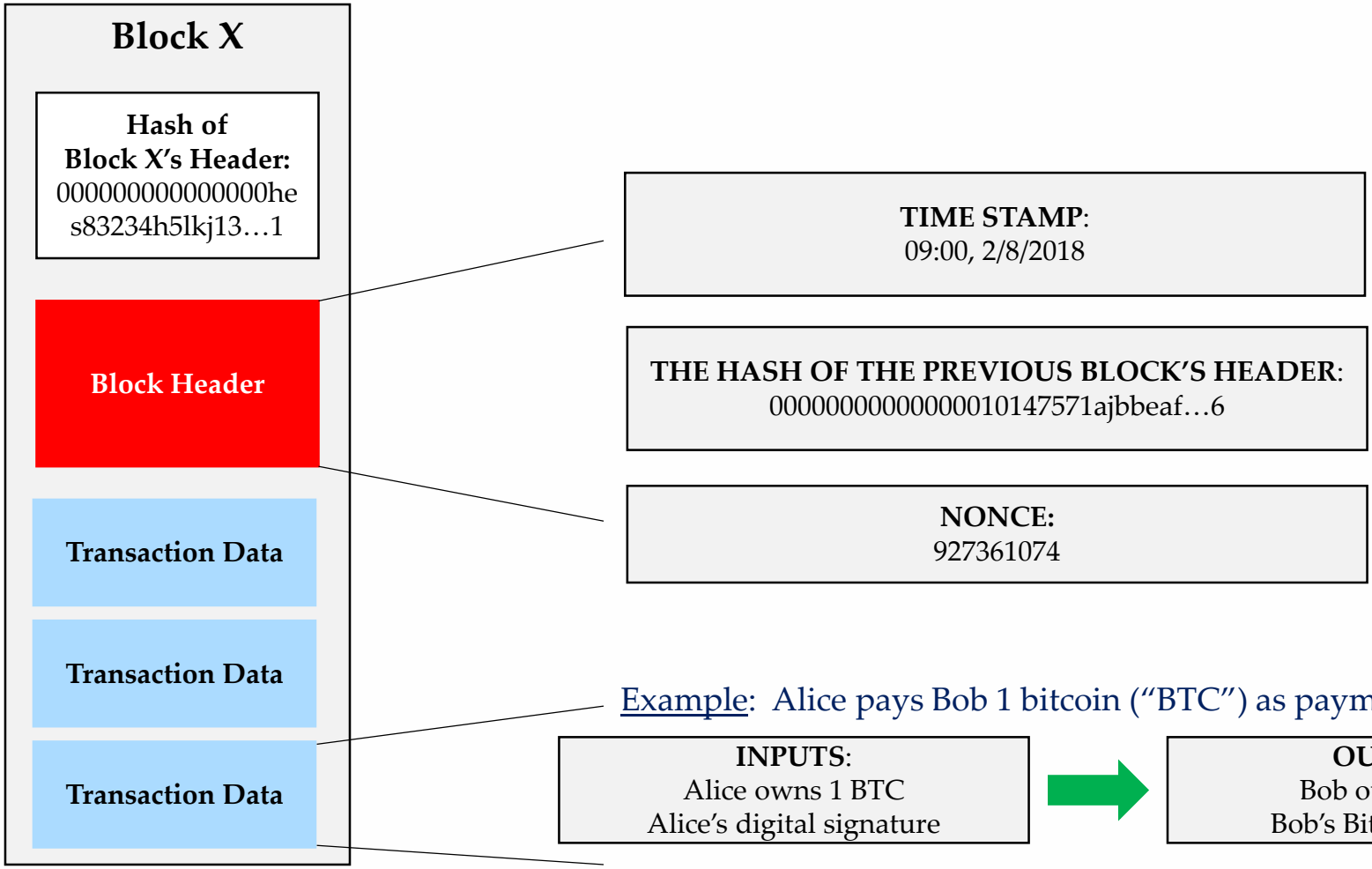
Private Key → **Elliptic Curve Multiplication**[1] → **Public Key** → **Hash Function** → **Bitcoin Address (On Chain)**

2. Bitcoin Implementation:
C. Transaction-Based Ledger (Cont'd)

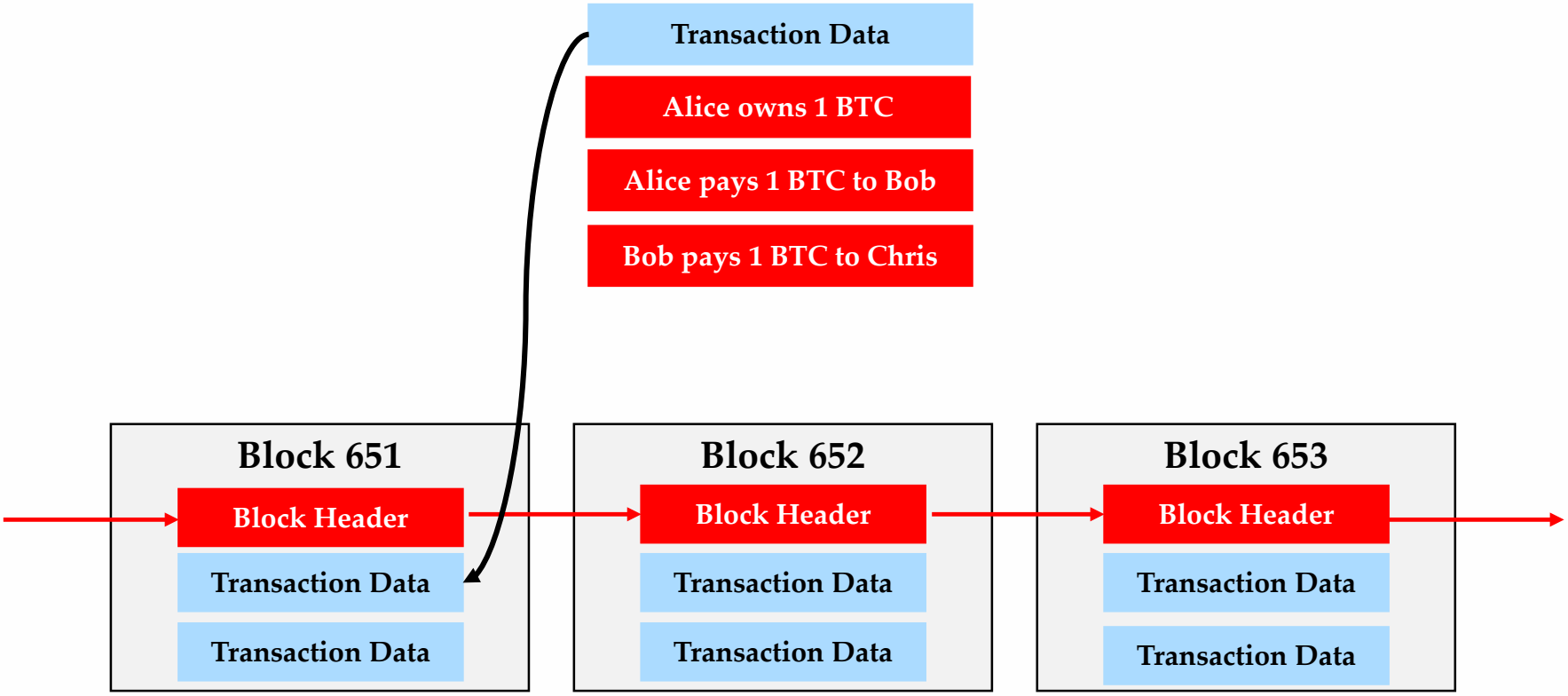


- Bitcoin provides a transaction-based ledger to track every bitcoin transaction in its history
- Bitcoin's ledger is an open, widely published and available for inspection
  - You can view Bitcoin transactions at https://blockchain.info/
- The ledger is "Append only" – you can add transactions, but can never delete them

2. Bitcoin Implementation:
C. Transaction-Based Ledger (Cont'd)



2. Bitcoin Implementation:
D. Ordered Chronologically (Cont'd)



2. Bitcoin Implementation: E. Consensus

How do we get the nodes in the peer-to-peer network to agree on which transactions are added to the blockchain?

2. Bitcoin Implementation: E. Consensus (Cont'd)

- Miners play a crucial role by building new blocks
- Any full node can be a miner
- Miners validate transactions, assemble them into blocks and propose new blocks to add to the blockchain

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes

} **Broadcasting**

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes
2. Each miner node collects new transactions into a new block

Broadcasting

Mining

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes
2. Each miner node collects new transactions into a new block
3. Each miner node works on solving the proof-of-work algorithm for a new block

Broadcasting

Mining

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes
2. Each miner node collects new transactions into a new block
3. Each miner node works on solving the proof-of-work algorithm for a new block
4. When a miner node solves the proof-of-work algorithm, it broadcasts the new block to all nodes

Broadcasting

Mining

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes
2. Each miner node collects new transactions into a new block
3. Each miner node works on solving the proof-of-work algorithm for a new block
4. When a miner node solves the proof-of-work algorithm, it broadcasts the new block to all nodes
5. Receiving nodes validate the transactions and accept only if all are valid; once accepted, the miner receives the block reward

Broadcasting

Mining

Distributed Consensus

2. Bitcoin Implementation: E. Consensus (Cont'd)

Steps in creating new blocks:

1. Transactions are broadcast to all Bitcoin nodes
2. Each miner node collects new transactions into a new block
3. Each miner node works on solving the proof-of-work algorithm for a new block
4. When a miner node solves the proof-of-work algorithm, it broadcasts the new block to all nodes
5. Receiving nodes validate the transactions and accept only if all are valid; once accepted, the miner receives the block reward
6. Nodes express their acceptance by moving to work on the next block, incorporating the hash of the accepted block

Broadcasting

Mining

Distributed Consensus

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus

Introduction

Consensus algorithm that requires **miners** to **expend computing resources** in order **to add new blocks** to the Bitcoin blockchain

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus

Introduction

Consensus algorithm that requires **miners** to **expend computing resources** in order **to add new blocks** to the Bitcoin blockchain

4 Factors That Ensure Accurate Transactions and Provide Security

1. Solving these puzzles are difficult
   (i.e., real computing resources are required)

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus

Introduction

Consensus algorithm that requires **miners** to **expend computing resources** in order **to add new blocks** to the Bitcoin blockchain

4 Factors That Ensure Accurate Transactions and Provide Security

1. Solving these puzzles are difficult
   (i.e., real computing resources are required)
2. Miners are selected at random
   (i.e., no one can predict who will solve the puzzle first)

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus

Introduction

Consensus algorithm that requires **miners** to **expend computing resources** in order **to add new blocks** to the Bitcoin blockchain

4 Factors That Ensure Accurate Transactions and Provide Security

1. Solving these puzzles are difficult
   (i.e., real computing resources are required)
2. Miners are selected at random
   (i.e., no one can predict who will solve the puzzle first)
3. Once the puzzle is solved, receiving nodes can quickly confirm the accuracy of the new block

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus

Introduction

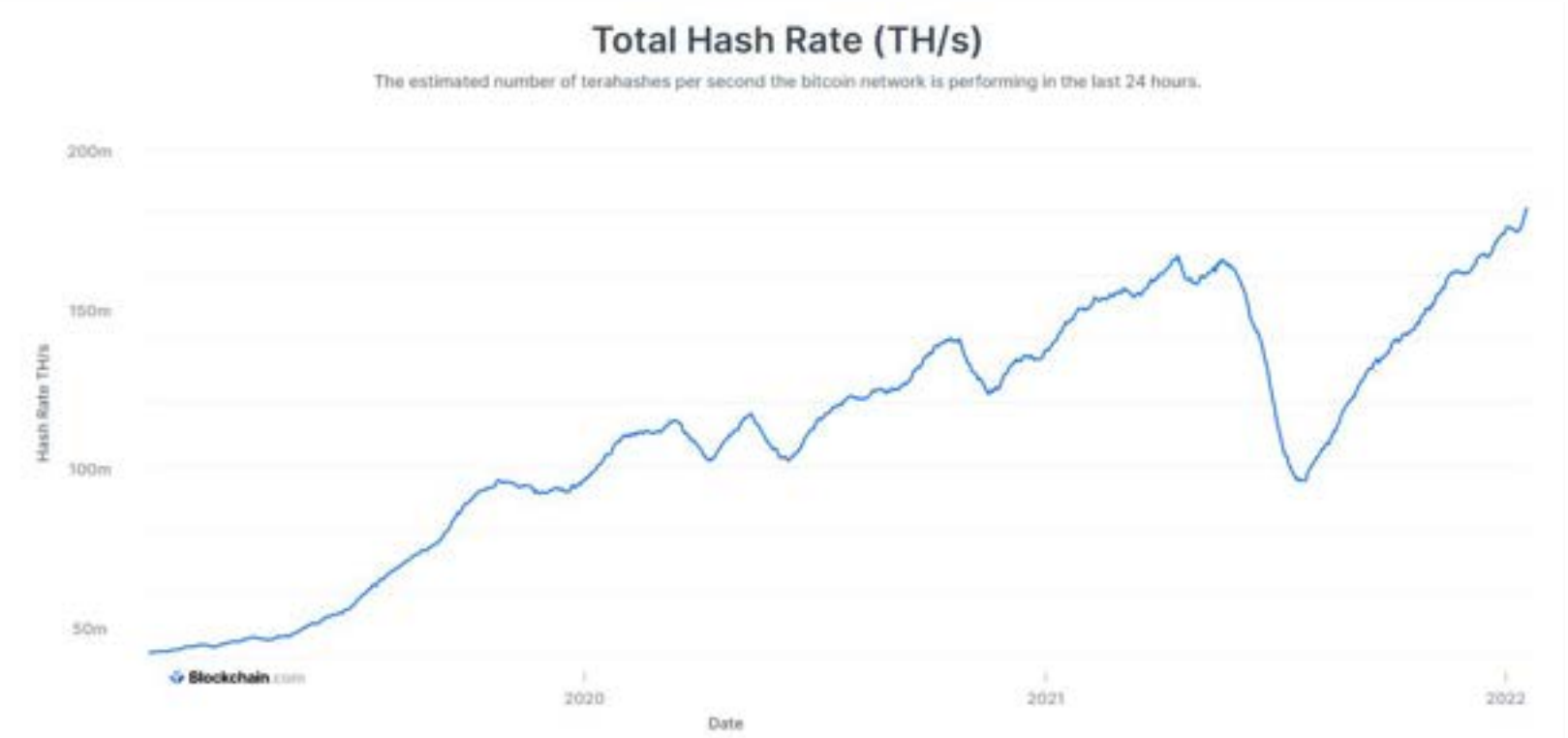
Consensus algorithm that requires **miners** to **expend computing resources** in order **to add new blocks** to the Bitcoin blockchain

4 Factors That Ensure Accurate Transactions and Provide Security

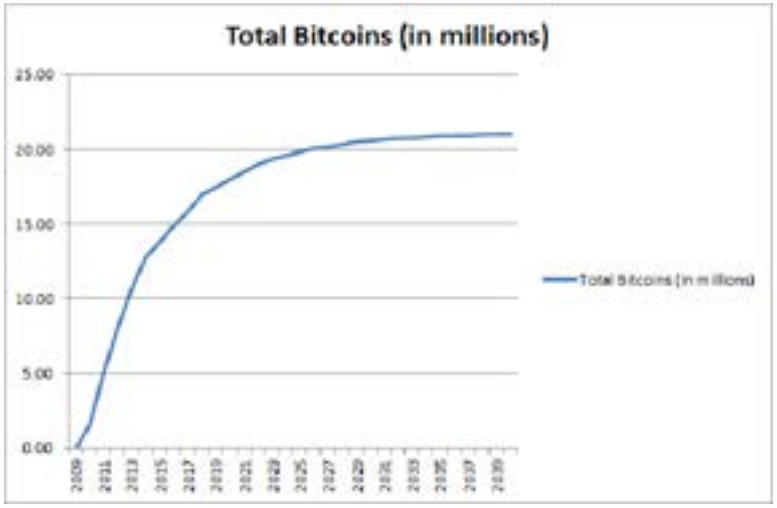
1. Solving these puzzles are difficult
   (i.e., real computing resources are required)
2. Miners are selected at random
   (i.e., no one can predict who will solve the puzzle first)
3. Once the puzzle is solved, receiving nodes can quickly confirm the accuracy of the new block
4. Once the hash output of a new block is accepted, it is immutable – nothing within the block can be modified without impacting the hash
   - Bitcoin links blocks together by their hash outputs, so that a single change anywhere would be easily detectible

2. Bitcoin Implementation: E. Consensus (Cont'd)

Proof-of-Work Consensus



2. Bitcoin Implementation: F. Fixed Supply Schedule



- Bitcoin is initially inflationary to incentivize mining, but has a fixed supply limit built into the protocol
- Limit of 21MM bitcoin is built into the Bitcoin protocol
- ~19MM bitcoin have been mined to date

2. Bitcoin Implementation

A. Peer-to-Peer Network

B. Secured by Cryptography

C. Transaction-Based Ledger

D. Ordered Chronologically

E. Consensus

F. Fixed Supply Schedule

2. Bitcoin Implementation: What is a Blockchain?

A **structure for storing data** in which groups of valid transactions, called **blocks**, form a **chronological chain**, with each block **cryptographically linked** to the previous one

Source: MIT Technology Review (2018)

3. Evaluation: Blockchain's Strengths

Coordination Tool Without A Middleman

- Allows interconnected parties to coordinate without relying on a trusted third party
- Potentially able to automate and improve existing processes

3. Evaluation: Blockchain's Strengths

Coordination Tool Without A Middleman

- Allows interconnected parties to coordinate without relying on a trusted third party
- Potentially able to automate and improve existing processes

Consensus as a Source of Trust

- Allows multiple parties to agree on transactions and the historical record of transactions
- The process is designed to work even if parties do not trust each other

3. Evaluation: Blockchain's Strengths

Coordination Tool Without A Middleman

- Allows interconnected parties to coordinate without relying on a trusted third party
- Potentially able to automate and improve existing processes

Consensus as a Source of Trust

- Allows multiple parties to agree on transactions and the historical record of transactions
- The process is designed to work even if parties do not trust each other

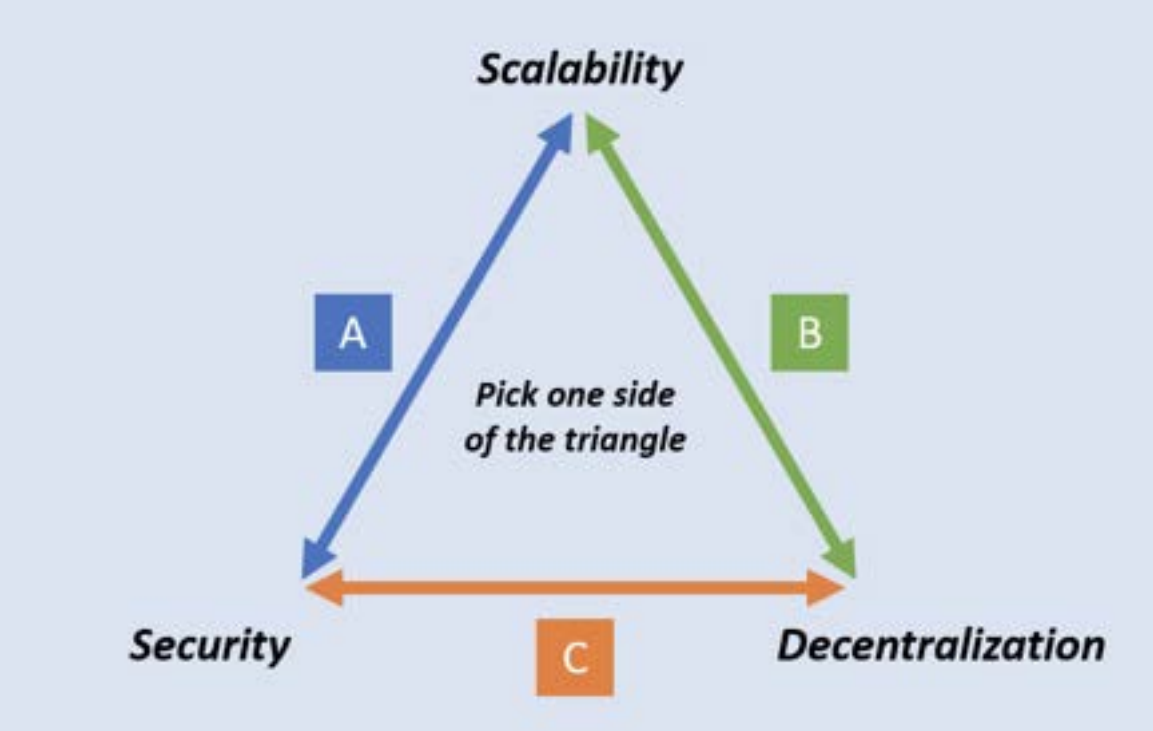
Transparency & Auditability

- A recording of valid transactions that is visible to participants
- Full history of transactions (and ownership) that participants can trace to their source

3. Evaluation: Blockchain's Strengths

Coordination Tool Without A Middleman	• Allows interconnected parties to coordinate without relying on a trusted third party • Potentially able to automate and improve existing processes
Consensus as a Source of Trust	• Allows multiple parties to agree on transactions and the historical record of transactions • The process is designed to work even if parties do not trust each other
Transparency & Auditability	• A recording of valid transactions that is visible to participants • Full history of transactions (and ownership) that participants can trace to their source
Immutability	• "Tamper resistant" • Once agreed, records cannot be changed by participants

3. Evaluation: Blockchain's Weaknesses



3. Evaluation: Blockchain's Weaknesses (Cont'd)

Reduced Performance

- Typically slower than centralized databases
- Typically more limited functionality (i.e., append-only)

3. Evaluation: Blockchain's Weaknesses (Cont'd)

Reduced Performance

- Typically slower than centralized databases
- Typically more limited functionality (i.e., append-only)

Governance

- More difficult to make changes to the database as future events occur
- Major changes require agreement from participating nodes

3. Evaluation: Blockchain's Weaknesses (Cont'd)

Reduced Performance

- Typically slower than centralized databases
- Typically more limited functionality (i.e., append-only)

Governance

- More difficult to make changes to the database as future events occur
- Major changes require agreement from participating nodes

Reduced Flexibility

- Downside of immutability
- Once transactions have been recorded, no ability to change even for good reasons (though there may be workarounds)

Section 2:

Ethereum: the First Multi-Purpose Public Blockchain

Vitalik Buterin

- 27 years old today
- One of the top developers in the blockchain space
- Wrote the Ethereum White Paper in 2013 (at age 19)

Ethereum vs. Bitcoin

	Bitcoin	Ethereum
Solution	Public blockchain	Public blockchain
Use Case	Electronic payments / currency	General purpose (no specific use case)
Buterin's View	• Like a Swiss Army Knife • Developers trying to explicitly support each new use case (e.g., 5 different tools for 5 different types of applications)	• "The ultimate abstract foundational layer" • "a blockchain with a built-in Turing-complete programming language" • Allows anyone to write smart contracts or decentralize applications with their own arbitrary rules

Definitions: Smart Contract

A computer program stored in a blockchain that **automatically moves digital assets** between accounts **if conditions** encoded in the program **are met**

Serves as a way to create a **mathematically guaranteed promise** between two parties

Source: MIT Technology Review (2018)

Szabo – The Idea of Smart Contracts (1997)

Vending Machine as a "Primitive Ancestor" to Smart Contracts

Szabo – The Idea of Smart Contracts (1997)

1. **Programmable exchange**
   - If a user inserts sufficient money, the machine dispenses the desired product
   - May dispense change based on the listed prices

Szabo – The Idea of Smart Contracts (1997)

1. **Programmable exchange**
   - If a user inserts sufficient money, the machine dispenses the desired product
   - May dispense change based on the listed prices
2. **Open system**
   - Anyone with coins can use it

Szabo – The Idea of Smart Contracts (1997)

1. **Programmable exchange**
   - If a user inserts sufficient money, the machine dispenses the desired product
   - May dispense change based on the listed prices
2. **Open system**
   - Anyone with coins can use it
3. **Built-in security**
   - Lockbox security for coins
   - The cost of breaking in < value of coins inside
   - Allows the machines to be deployed widely and without constant monitoring

Szabo – An Illustrative Example

Digital Security System for Automobiles

Can we embed contract terms into physical and / or digital assets?

Szabo – An Illustrative Example (Cont'd)

Digital Security System for Automobiles

- **How would it work?**
  1. Develop cryptographic keys for each vehicle
  2. Program the software inside the vehicle to require the cryptographic keys to operate the vehicle
  3. Give control over the cryptographic keys to the person that is allowed to operate the vehicle
- No more theft?

Szabo – An Illustrative Example (Cont'd)

Digital Security System for Automobiles

- **What about special situations?**
  1. If you take out a loan when purchasing a vehicle
     - Cryptographic keys automatically revert to the bank if you fail to make a payment within a certain amount of time
     - When the loan is paid off, you get full ownership of the cryptographic keys
  2. Common sense exceptions – do not revoke keys while the car is driving on a highway

Section 3:

Private Blockchains

What is a "Private" Blockchain?

Public Blockchain

- Permissionless
- Anyone is allowed to participate and has the ability to transact on the blockchain
- Users are free to enter or exit at will

Private Blockchain

- Restricted access to certain users or operators
- Building of the blockchain is limited to a known set of entities
- End users can participate, but must rely on interfaces offered by permissioned operators

Perceived Problems with Public Blockchains by Enterprises

1. **Reversibility** – inability to edit transactions if a mistake is made or circumstances change
2. **Privacy** – all transaction information is available publicly
3. **Speed** – lack of system responsiveness
4. **Scalability** – inability to scale transaction volume significantly
5. **Governance** – difficulty in updating protocols as circumstances change

Source: Blockchain for Enterprise by Hamida et al

Private Blockchains: Key Adaptations for Enterprises

1. Restrict users
2. Add data privacy
3. Increase scalability
4. No forks
5. Easier governance

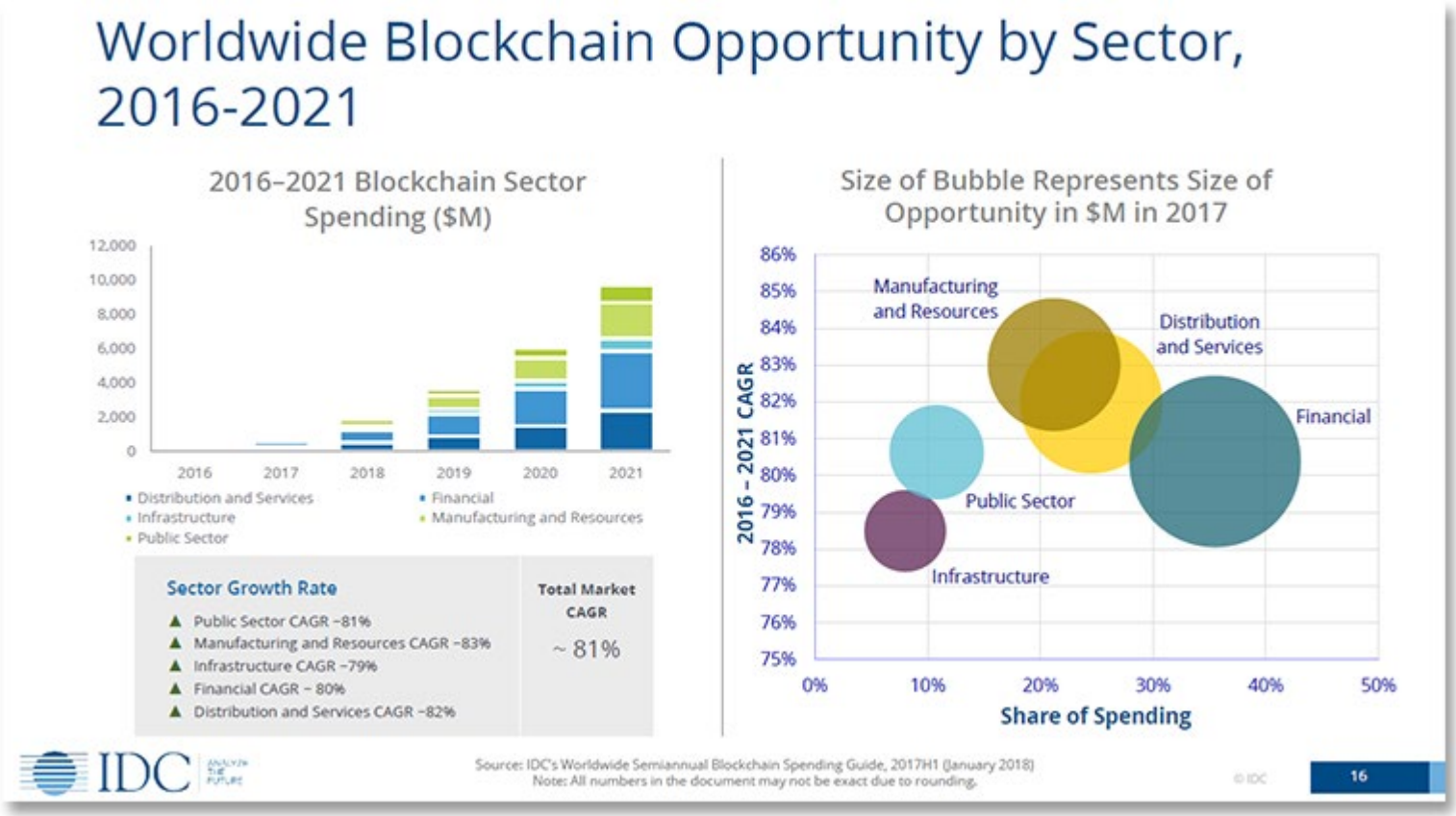
Source: Blockchain for Enterprise by Hamida et al

For Example, Many Consensus Algorithms Scale Better Than Proof-of-Work

1. Proof of stake
2. Proof-of-elapsed time
3. Leader based consensus
4. Practical byzantine fault tolerance
5. Federal byzantine agreement
6. Tendermint
7. Diversity mining consensus

Source: Blockchain for Enterprise by Hamida et al

Companies are Investing in Blockchain Technology



Walmart's Food Traceability Initiative Using Blockchain Technology

The Problem (Example)

- **E coli outbreak** in romaine lettuce from the Yuma region of Arizona
- No method to detect which specific farms were impacted
- Therefore, **all romaine lettuce** from the region **was destroyed**

The Goal

- **Trace all fresh leafy greens** sold in Walmart **to the farm** where it originated **in seconds**
- Fully implemented in 2019

Benefits

- **Minimize negative impacts** on consumer health
- **Assist health officials** in investigating and preventing outbreaks
- **Help farmers** avoid losses if they are not affected

How Can Organizations Use Blockchain?

Financial Transactions

- Tracking the ownership of tangible, intangible or digital assets
- New digital currencies
- Cross-border payments

Financial Institutions

- Clearing and settling securities transactions
- Compliance and regulatory reporting
- Trade order generation
- Voting of any kind
- Dividend distributions

Supply Chain

- Track an asset
- Verify attributes

Mobility

- Safely store car data
- Decentralized ridesharing platforms
- Decentralized transportation ecosystem where people can use the same token to ride on a bus, rent a bike or carpool, without any central authority

Energy

- Certify green energy production
- Trade energy at the local grid level

Various Industries

- Customer loyalty programs
- Customer payments
- Inventory controls
- Medical recordkeeping
- Tax collection / payment
- Government benefit distribution

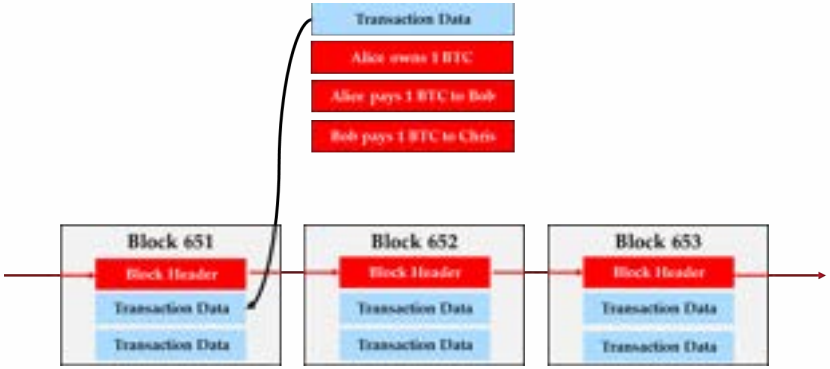
Source: An Introduction to Blockchain by Allayannis, Blockchain for Enterprise by Hamida et al

Section 4:

Key Takeaways

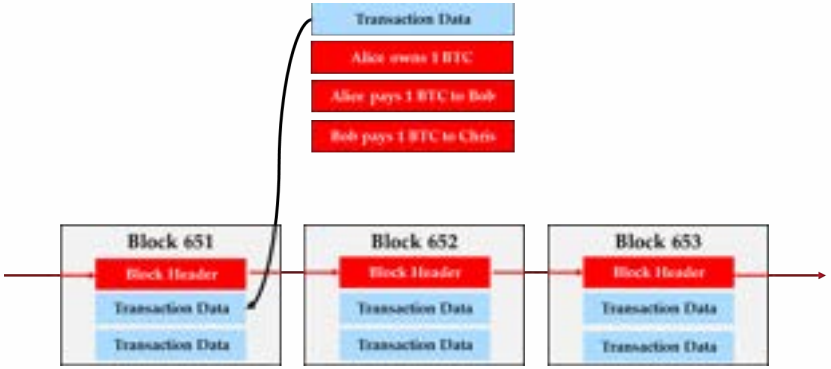
Key-Takeaways on Blockchain

1. **A basic blockchain is a simple recordkeeping device (e.g., a ledger)**



Key-Takeaways on Blockchain (Cont'd)

1. A basic blockchain is a simple recordkeeping device (e.g., a ledger)
2. A public blockchain is a decentralized recordkeeping device maintained by its users



Key-Takeaways on Blockchain (Cont'd)

1. **A basic blockchain is a simple recordkeeping device (e.g., a ledger)**
2. **A public blockchain is a <u>decentralized</u> recordkeeping device maintained by its users**
3. **Some public blockchains allow developers to:**
   - Create new digital assets
   - Program smart contacts to automatically control the ownership of digital assets
   - Build applications on top of these digital assets