**LFS170x - Linux FoundationX**

Blockchain: Understanding Its Use Cases and Implications



Welcome to LFS170x: Blockchain - Understanding Its Use Cases and Implications

Linux FoundationX - LFS170x (2021 version)

Once you complete this course, you will have a good understanding of what blockchain is and its impact and potential for change around the world. You will also be familiar with some of the immediate blockchain use cases in technology, business, and enterprise products and institutions.

LFS170x is part of the [Blockchain for Business Professional Certificate](https://www.edx.org/professional-certificate/linuxfoundationx-blockchain-for-business). Skills in blockchain are being reported as the top job-skill in demand. This program is designed for the business professional who needs to understand the potential (or threat) of blockchain to their company and industry. Armed with better information of the blockchain landscape, this program will help you rise to new challenges in your current role by giving you a new dimension on which you can add value to your employer.

[Welcome!](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@f8765c952674412ab9fe9e185d56cfbb)

# Chapter 1. Introduction to Blockchain

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@d63d14d2b81a45e2886828c2b89b8ca1)
  2. [Introduction to Blockchain and The Early Internet](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@4ee4f8bc0dad4e09ad99b376b061b0c9)
  3. [Blockchain Basics](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@7c07cfafff4943d7af3f883c18082724)
  4. [Career Opportunities in Blockchain](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@31aac67795644f13bd993a27072e4902)
  5. [Blockchain Use Cases](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@b5d8fe9812784047954a2e6fcf3eba66)
  6. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@a401d966ab5a4a04a9fdb4a8290d563d)

# Chapter 2. Blockchain Mechanics

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@1088154e18e0450f9bb3a418121008c1)
  2. [Understanding Ledgers](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@47866c822c894049af4a2213c7a9d469)
  3. [Cryptography](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@5f4fa5704bd0432aad3727d0fb33c66a)
  4. [Transparency and Immutability](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@b2327e4750bc4beda267bf03132fbe6c)
  5. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@3351e649f41b433f9bd4243c67c3f697)

# Chapter 3. Blockchain Functions

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@ebc40eedf3f24ac6af007fe3d04f93c5)
  2. [Smart Contracts](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@f412320eace9423fb674eadd8118084c)
  3. [Blockchain Security](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@b51fe96ee70c460d94db5d7d2f1959b5)
  4. [Public and Permissioned Blockchains](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@9891087256c440f3bfc65d7492893a24)
  5. [The Blockchain Transaction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@f43ac8b873a74859af3de0a87f61337a)
  6. [Consensus](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@177e1191bc254b33aac28ac2d4ce45fd)
  7. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@ce8663a813bb4af38f4785b86ebd88a3)

# Chapter 4. Blockchains and Governance

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@fd421df72f0848b1a7bf96e602450141)
  2. [Open Source Code](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@da506b1cceb54093af75a07845fb5a36)
  3. [Governance](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@43e86826079c4284852c1bed6e11186b)
  4. [Identity and Anonymity on Blockchain](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@9ebbad69d6be4e7e9180f8e86ff2c1bc)
  5. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@da27a79d56404477b3925c255eddf94a)

# Chapter 5. Blockchain Problem Solving and Future Trends

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@c8a9ccbd46b748b88d7c2867f42ea0d0)
  2. [Problems Blockchain Solves](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@ed004881d34343899d18744ba074c9ee)
  3. [Digital Currencies](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@8ec45efd8fb04b75b6268e0b18f06410)
  4. [Future Trends](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@af775e511d8f47d49a698b07e7164386)
  5. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@295e06b2437946df9a58b5ec475c88ea)

# Chapter 6. Blockchain Use Cases

* 1. [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@160833e7ed3848079dfea0f73128f459)
  2. [Blockchain in Practice](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@dae16c3b9a8b4a84a427752af12cd165)
  3. [Enterprise Solutions](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@e1cdd90003f34a3395a52438d3d5d318)
  4. [Public Sector Solutions](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@f35e0c4e6550431ea09ba040f0b53bf2)
  5. [Social Impact Solutions](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@561e3635558e41b1a49d4a1accbda125)
  6. [Platform Developer’s Solutions](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@8b50053f913e49ccbb0456700d9aae55)
  7. [End User Solutions](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@0215630608bb42ac8062133456146cdf)
  8. [Future of Blockchain](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@1693a69f91cf4e349b8467d6048958f4)
  9. [Summary](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@6e8e2af862604b129e957f5eb1fd1889)

# Chapter 1. Introduction to Blockchain

## [Learning Objectives](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+1T2020/block-v1:LinuxFoundationX+LFS170x+1T2020+type@sequential+block@aebb38d0b0e44fdc8ab82f3fa11cc9aa)

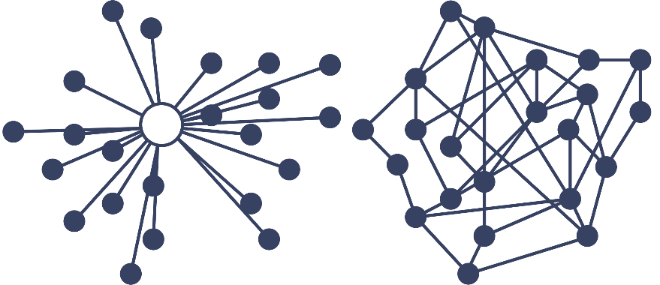
By the end of this chapter, you should be able to:

* Discuss blockchain technologies and the early Internet.
* Explain the difference between blockchain and the cryptocurrency Bitcoin.
* Discuss blockchain characteristics.
* Explain blockchain components.  
  - Explain what the block in blockchain is.  
  - Explain how blocks are chained together.  
  - Discuss the concept of immutability in a blockchain.
* Discuss about the roles and users in a blockchain community.
* Discuss use cases for general blockchain solutions.

## What is Blockchain

Blockchain is a peer-to-peer ledger system that allows peers to transact directly with each other eliminating the need for a central authority.

At its core, blockchain is a system for recording information about a transaction in a new decentralized way that makes it difficult or impossible to alter. These transactions are stored on sheets or blocks in a digital ledger that is shared among the participants of the network. Consensus on the transactions, brings the peer-to-peer network into agreement. Once the agreed-upon transactions blocks are recorded in the immutable ledger, trust becomes a fundamental component built into the system.



## The Early Internet

What started as a [DARPA](https://www.darpa.mil/) (Defense Advanced Research Project Agency) experiment in decentralized computing communications between two university labs in California in 1970, became the Transmission Control Protocol/Internet Protocol, INTERNET PROTOCOL SUITE (TCP/IP) developed as a standard in networking protocol or computer communication standards, and it is the backbone of today’s Internet.

With the TCP/IP protocols in place, users had the ability to link hypertext documents in an information system accessible from any node or computer using the TCP/IP protocol. The resulting information system or database is today’s World Wide Web.



With the birth of the World Wide Web, expanded usages of this new technology arose along with expanded business opportunities. Web servers, people who host and store the documents and web browsers, companies set up to help you view linked documents, help create a household need for this technology and the Internet explosion began.

## The Growth of the Internet

The Internet can be grouped into three distinct segments characterized by the way people interact with this new technology.

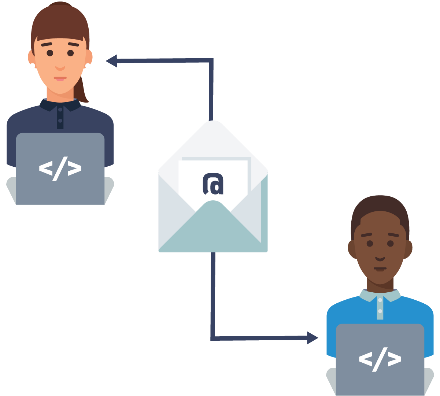
* Web 1.0 - Internet of Connection
* Web 2.0 - Internet of Information
* Web 3.0 - Internet of Value

Let’s discuss them in more detail.

## Web 1.0 - Internet of Connection

Characteristics of Web 1.0:

* Development of a host of web-based applications, which fostered in online services, such as email.
* Content from administrator.
* Managed by a central authority.
* Read-only, information was “pushed” to users.
* Email was the first widely adopted application on the Internet.



**Email, Electronically Transferring Information**

Computers and items for connection became necessities. Technology advancements in computers brought on changes, floppy disks became hard drives that stored MB that turned into GB that turned into TB. Internet speeds switched from kilobits to tens of megabits per second, to gigabits per second and RAM grew from hundreds of kilobytes to gigabytes and the dot-com bubble began. Companies appeared attempting to cash in on this new technology, most notable was a company called [Netscape](https://isp.netscape.com/) which developed the first commercial Web browser.



## Web 2.0 - Internet of Information

Characteristics of Web 2.0:

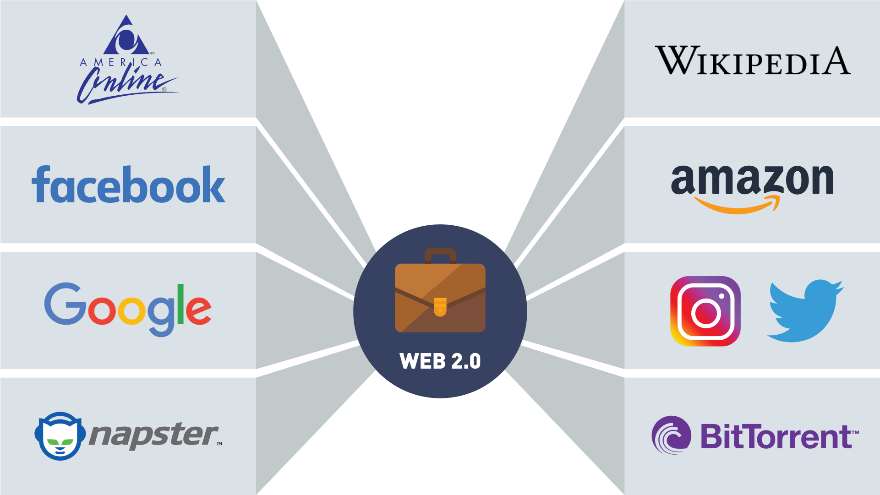
User-generated content.

Read-write, individuals can interact with information.

Information became siloed.

Data became a commodity.

Web 2.0 saw emerging marketplaces that brought together unrelated buyers and sellers in a seamless low-cost way. Data became a commodity collected, siloed and sold; we were giving up our information at a frenzied pace. Websites let users generate content, social networks became part of our lives.



**Social Networking**

## Real World Examples of File Sharing

**Use Case: Music Sharing Companies**



Approach: Music stored on many computers connected peer-to-peer. [Napster](https://us.napster.com/) software supplied its users with a centralized index of all music files and directed users to where these files were located on the connected peers' computers.

Result: The industry cracks down music sharing companies copyright infringement. Napster is forced to take down its index, shutting down the platform.



Approach: Music stored on many computers connected peer-to-peer. [BitTorrent](https://www.bittorrent.com/)'s software was purely decentralized, the files were stored as packet on the peers' computers and when a request was made for a song, the software would find the packet and send it to the user.

Result: The industry cracks down music sharing companies copyright infringement. BitTorrent is asked to shut down their platform. Since they do not control the software once downloaded to a peer, if two computers are running the software sharing can still occur.

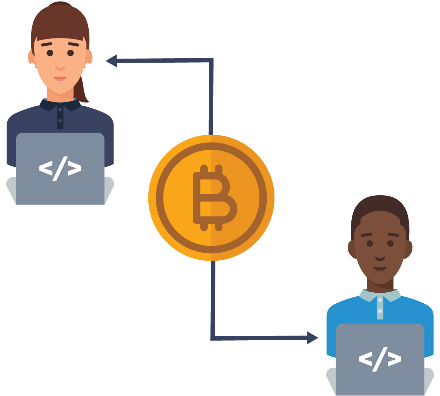
Music sharing companies gave us the first glimpse into peer-to-peer networking. Besides copyright infringement, the main challenge for the music sharing companies was file integrity. You could never be sure the file you requested was the file you would get and there was no one to complain to. Both Napster and BitTorrent are operating today, with different business models.

## Web 3.0 - Internet of Value

Characteristics of Web 3.0:

* Community interaction.
* More connected, open, and intelligent.
* Distributed ledgers or blockchain technology, smart contracts, machine learning and artificial intelligence.
* Identity and information will be held by the individual, breaking data silos.

Bitcoin is the first widely accepted application for the Internet of Value (just as email was the first big application for the Internet of Information).



**Cryptocurrencies, Electronically Transferring Value**

The Internet of Value represents a world where value is exchanged at the speed in which information moves today. The Internet is still the basic platform that these new technologies operate from. The new Web 3.0 browsers are being built to help you manage your cryptocurrency, keys, passwords and other blockchain features. Let’s now dive into blockchain basics.

## What Is Blockchain?

There's a new technology that has the power to revolutionize how you, businesses, and the world interact!

You've probably heard of it: it's called Blockchain!

But what is it? How does it work? How does it affect you?

Hearing the word "blockchain" is comparable to hearing the word "internet" in the early 90s.

It seemed elusive, like something that wasn't going to impact your daily life.

But here we are, more than 20 years later. Think about how the Internet has transformed businesses, commerce, communication, even music and video.

And that's not even touching on how the Internet has affected you.

We all carry the Internet in our pocket.

We can transfer money at the touch of a button, check the weather, get directions, and even have food delivered to our door.

The next technology to have that kind of impact isn't some of the buzzwords you hear.

It's not big data. It's not artificial intelligence. It's not even social media.

It's BLOCKCHAIN!

Now, let's discuss the vast promise blockchain has for every business, every society, and for everyone listening today.

And let's begin with an example we are all familiar with.

When you attach a file, a Word document, an Excel sheet, a PDF to an email, you aren't sending an original.

You are actually sending the recipient a copy, and that's a great way to move information around.

But it's not so great when it comes to things like money, stocks and bonds, music, loyalty points, intellectual property, tickets to a game or concert.

Then, sending a copy is suddenly a very bad idea.

Let's look at an example.

If I send you tickets to a concert, it's important that I don't send you a copy.

You now own the original asset and I can no longer use or sell those tickets.

Similarly, if I send you $100, it's important that I still don't own the $100, or have the ability to send it to anyone else.

Both of these examples illustrate a double spend problem, and those can be eliminated with blockchain.

So now, in this exciting time in history, every kind of asset, from tickets, to money, to music, can be stored, moved, exchanged, and transacted without an intermediary.

People everywhere can transact peer-to-peer and trust each other by using collaboration and cryptography.

How did we get here, to this place of trust and collaboration?

It began when Satoshi Nakamoto, whose true identity is still unknown, released a white paper in 2008, introducing a purely peer-to-peer version of electronic cash known as Bitcoin.

It is here that blockchain technology made its debut.

Even today people believe Bitcoin and blockchain are one and the same.

They are not!

Bitcoin, another alternative currency, utilize blockchain technology.

While an important one, Bitcoin is only one use case for blockchain.

Blockchain allows people to exchange assets and perform transactions without a third party.

Imagine a world where you don't need intermediaries.

While traditionally we have needed central authorities to trust one another and fulfill contracts, blockchain makes it possible to have our peers guarantee that for us. But how?

Assets, like we've discussed, are no longer stored in a central place, but distributed across a global ledger, using the highest level of cryptography.

When a transaction is conducted, it's posted across tens of thousands of computers around the globe.

These transactions are recorded as blocks.

Let's imagine a sheet of paper that has 25 lines.

When a sheet is filled up with 25 transactions, the block is validated via group consensus.

Once the page has been validated, it is added to a stack of previously validated sheets.

Each sheet on the stack can be assumed to be trustworthy because, once a sheet is validated, it can't be changed.

Because at this point, all the sheets are linked together.

And to link our sheets together, we embed information from the previous sheet of paper into the new, recently validated sheet.

In blockchain, our sheet of paper is equal to a block.

The act of embedding a previous block of information into the current block of information is called chaining, hence, the name blockchain.

In order to compromise or hack a blockchain network, someone would have to gain control of the majority of computers in that network.

This is extremely difficult to do.

There is no longer a single point of failure, and this is what makes blockchain infinitely more secure than what we have today.

Blockchain isn't just for assets, though. It extends to contracts.

These are called Smart Contracts.

And what are they exactly? What they sound like.

A smart contract self-executes and handles enforcement, the management, and performance of agreements between people.

Examples of smart contracts include insurance policies, copyrighted content, escrow and lending, wills, and trusts.

Smart contracts will revolutionize how we do business.

There are so many possibilities with blockchain; not just in the now, but with things we haven't begun to think about yet.

## Blockchain Is NOT Bitcoin

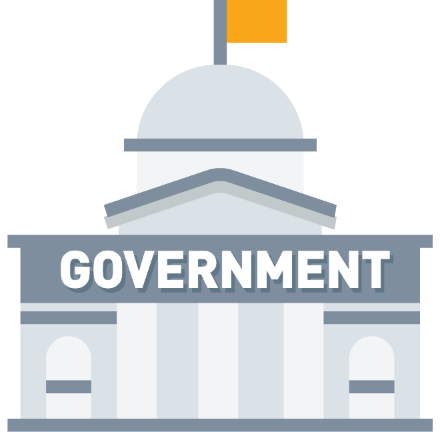
Bitcoin is a digital currency running on top of the Bitcoin blockchain. We will develop the concept of cryptocurrencies in later sections. Bitcoin has no borders and was created as a decentralized payment system, an alternative to currencies issued by governments. Let us look at what it means to be a currency.



 Purpose of currency:

* Medium for exchange
* Store of value over time
* Accepted as a measure of worth

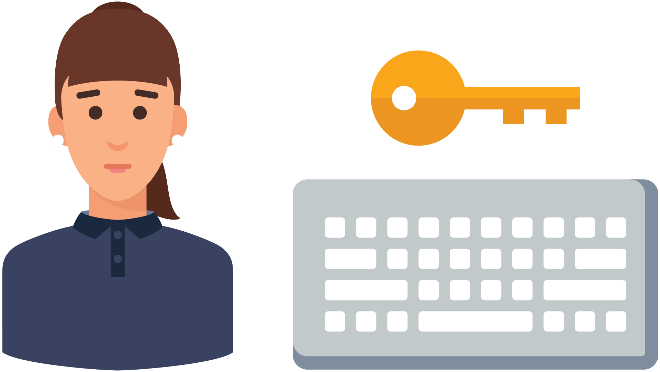
According to [Investopedia](https://www.investopedia.com/terms/f/fiatmoney.asp), “Fiat currency is money a government issues that is not backed by a commodity like gold. Fiat money is backed by the government and it has value because the government says so and the citizens believe it. The dollar is an example of the United States fiat currency”.



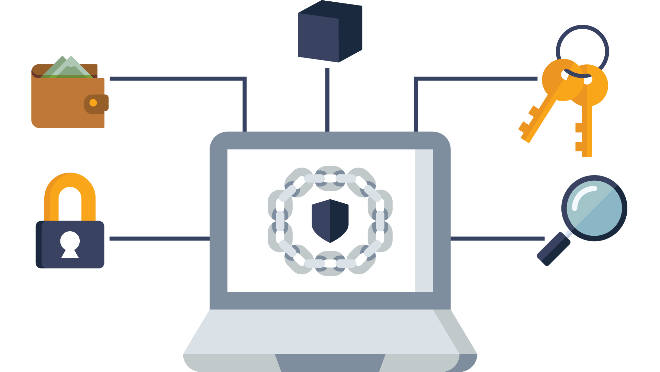
Apart from the government issuing Fiat, a centralized authority needs to keep track of assets. Banks, credit card companies’ stockbrokers are companies that represent a centralized authority that controls your money.



Bitcoin is transacted over the Bitcoin network, which is an open, public blockchain network. If you have an Internet connection and a Bitcoin wallet application, you can receive and send the cryptocurrency Bitcoin. Cryptocurrencies like Bitcoin have value because it can be proven with math. Unlike banks offering centralized services, you are responsible for your cryptocurrency.



[Bitcoin](https://bitcoinfoundation.org/) and cryptocurrencies are great use cases for blockchain, but they are just one component of the technology. Some privately run blockchains do not use cryptocurrencies as a function of their blockchains. Blockchain technology can easily transfer anything from property rights, stocks, and even artwork. There are many more use cases that utilize blockchain technology far reaching than currency transfer.



## Blockchain - Let’s Cover the Basics

In this section, we're gonna talk about what blockchain is: a real simple introduction to the basic concepts and principles behind blockchain.

When you hear about blockchain, you probably think of new, high-emerging tech.

Well, really, all blockchain is, is a combination of technologies that have existed for a long time.

They're simply combined in a new and creative way to give us an amazing new platform on which we can start to build solutions.

Blockchains are often referred to as digital decentralized ledgers.

And when you think of that, all you can think of is something we're all familiar with: a simple notebook.

When we talk about blocks and blockchain, all we're really talking about is a page of data.

We fill up this page with any kind of data imaginable, and, while most of the time today, we're talking about recording financial transactions on this page, or this ledger, or this block, we can record any kind of data we want.

In fact, you can think of a block as being exactly like this piece of paper, in that the paper doesn't know or care what kind of information you're recording on it.

And just like the paper doesn't know or care, neither does the block in a blockchain.

So, while most of what we're recording today are financial transactions, we could also record things like voting records and results, land titling, medical records, or even the opening notes to a Beethoven symphony.

What makes this blockchain notebook incredibly powerful is that all the pages are linked together.

So that if I go back and try and change any data on any page anywhere in this notebook, I break a link in a very obvious and easy-to-determine way.

This gives blockchain a key property known as immutability.

And it's what gives blockchain the security that we've come to know and depend on.

Blockchains also give us an incredible technology called smart contracts, which allow us to programmatically define the rules and steps that should be performed any time a certain type of event is recorded in our blockchain or in our magic notebook.

So, when you think about blockchain, don't get intimidated thinking it's something high tech, and new, and cutting-edge.

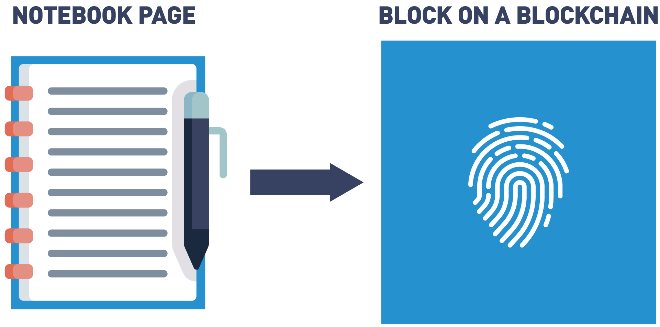
It's simply a new and creative way of combining old ideas together to give us an incredible new solutions platform.

And the rest of this series will take a deeper dive into these basic concepts and principles, but if the idea of this unchangeable linked notebook makes sense, congratulations!

You already understand all the major principles and concepts required to understand what blockchain is and why it's so powerful.

## Let’s Review an Analogy

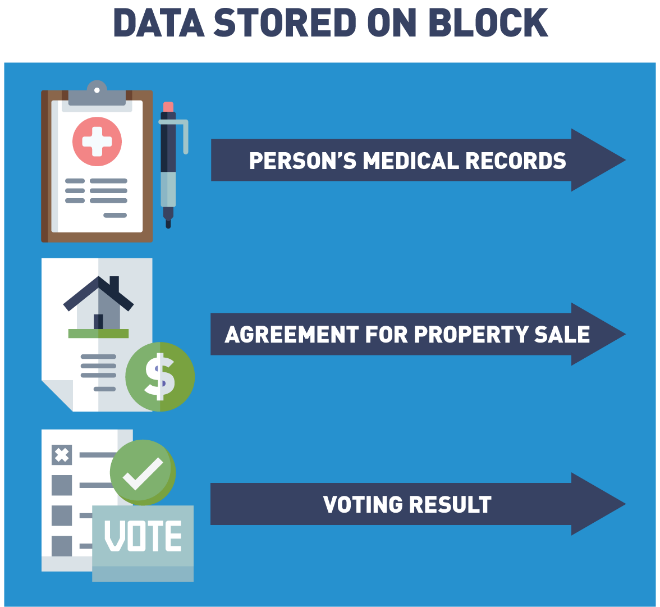
A block on a blockchain can be thought of much like a page in a notebook. Data is stored on a block, just like data is written on a page of a notebook. Similar to a sheet of paper, the digital ledger doesn't care what kind of information you're putting on it. So, while most of what we're recording today are financial transactions, we could also record things like voting records and results, land titling and medical records, etc. What makes this blockchain notebook incredibly powerful is that all the pages are linked together. So that if you go back and try and change any data on any page anywhere in this notebook, you break a link in a very obvious and easy-to-determine way.



## Let’s Review an Analogy: Data Stored

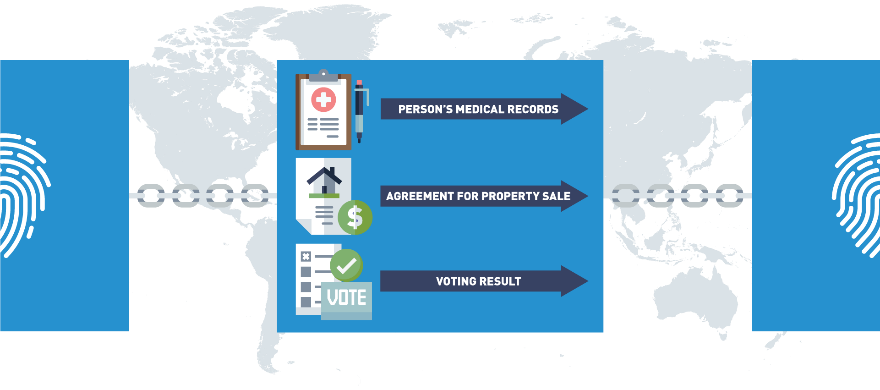
Any data can be stored on the same block. Examples of stored data include:

* Medical records
* Property agreements
* Voting



## Let’s Review an Analogy: Blocks Are Chained Together

Each block is chained or tied to the previous block by embedding the block with information from the previous block (we will go through this in depth later in the course).



## Let’s Review an Analogy: Blockchain Is Immutable

If the data is tampered with anywhere in the chain, the links will break in a very obvious way:



This provides immutability and security.

## Blockchain Features

Blockchain is a combination of several other underlying technologies that have never been combined so successfully. Here are some features that make this new technology so revolutionary.

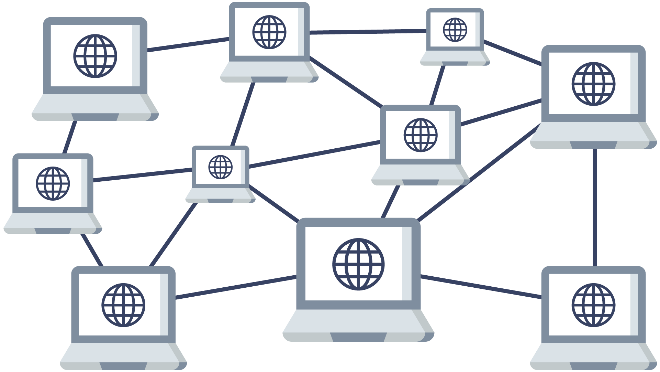
## Blockchain Features

## Blockchain Features: Distributed Ledger

A ledger is a collection of transactions. It is not a collection of assets. Assets are part of a transaction, but the ledger records the transaction. With a digitally distributed ledger or a blockchain, no one owns the ledger. The ledger is distributed among participants in the network, all running the same blockchain protocols. It is decentralized in that an identical copy of the ledger exists on every node/computer on the network. In 2009 with the publishing of Satoshi Nakamoto’s whitepaper [*“Bitcoin: A Peer-to-Peer Electronic Cash System”*](https://bitcoin.org/bitcoin.pdf) , [Bitcoin](https://bitcoinfoundation.org/) became the first application to leverage blockchain technology by recording the first asset transfer on a public blockchain ledger.

## Blockchain Features: Peer to Peer Network

The ledger is stored, updated, and maintained by a peer network. Nodes form the infrastructure of a blockchain network. They store, spread and preserve the blockchain data, so a blockchain exists on nodes. All nodes on a network follow the same rules of operation or protocols, but nodes have different roles. A full node contains a copy of the blockchain protocol, transaction history of the blockchain and aids in the maintenance of the blockchain. User node interacts with the ledger. With blockchain technology, a lack of a centralized authority is replaced with a peer-to-peer network.



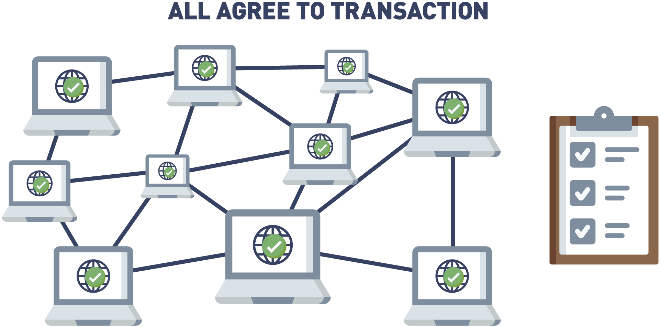
Blockchain networks can be public or private. A public blockchain is open to anyone with an internet connection and the appropriate application. A private blockchain grants access and rights to its users before they can interact.

## Blockchain Features: Transparency

In a blockchain, we can see all the transactions that have occurred on the shared or distributed ledger. A blockchain stores details of every transaction that occurred since the first transfer. This first transfer, along with some system information that we will discuss later, becomes the first block in our chain and is referred to as the genesis block. Since every node shares a copy of the agreed-upon ledger, there is no friction about the transactions, everyone has the same agreed-upon copy. Centralized systems are not transparent, the information about the ledger is controlled by one authority.

## Blockchain Features: Consensus

Blockchain ledgers are different from centralized ledgers because network participants have an agreement upon what is in the identical ledger. In order for the blocks to be added, all the nodes in the system come to agree as to what transactions are accurate and should be added to the chain of blocks. Since there is no central authority telling the nodes which transactions are valid, a new way to reach agreement or come to consensus is needed.



The way in which each blockchain comes to consensus is built into the protocol, they are the rules built into the code that determine how the nodes will add ledger transactions. The Bitcoin network uses a proof of work consensus model. This process of building agreement among a group of mutually distrusting participants is a benefit of blockchain consensus.

## Blockchain Features: Encryption

Encryption and cryptography are combined with blockchain technology to assure the information on the blockchain is authentic. In our previous real-world example, the music sharing companies not only had legal issues from copyright infringement, they also could not solve the data integrity problem. Music downloaded from these platforms was only as reliable as the anonymous person storing and sending it. The data had no integrity. Cryptography and blockchain offer a secure way to prove something is authentic. Instead of relying on third-party, trust is put into cryptographic algorithms that prove the provenance and authenticity of an attestation.

## Blockchain Features: Programable

Some blockchains offer an additional feature, smart contracts, that impose logic into the system. If you think of the Bitcoin blockchain as a calculator recording financial transactions, you can think of these blockchains as computers. The first of these blockchains is [Ethereum](https://ethereum.org/en/), created by Vitalik Buterin, Joe Lubin and team. Ethereum runs the EVM, Ethereum Virtual Machine, which has the ability to impose digital agreements or smart contracts. A smart contract is a program that runs on a blockchain that can impose contractual agreements. Typically, they work as digital agreements that enforce a specific set of rules. These rules are predefined by computer code, which is replicated and executed by all network nodes. Blockchain smart contracts allow for the creation of trustless protocols. This means that two parties can make commitments via blockchain, without having to know or trust each other. They can be sure that if the conditions aren't fulfilled, the contract won't be executed. Other than that, the use of smart contracts can remove the need for intermediaries, reducing operational costs significantly.

## d. Blockchain Actors

With new technology comes new opportunities and this holds true for blockchain. As this nascent technology matures, more positions are needed to fill this increased demand. A few of these new positions are listed here:

Entry Level and Internship Positions  
Mentorship programs and internships include assisting in the designing and developing blockchain distributed ledgers based on both public and private or permissioned blockchains. With open-source, development and success depend on an active community, Working Groups and Hackatons are available for the newcomer. Skills include basic blockchain knowledge and some familiarity with basic programming languages.

Blockchain Architect  
For a blockchain solution to be functional, it first needs to exist. The blockchain architect is the person or group who designs and implements the backend code of the blockchain. The blockchain architect works with blockchain engineers and technical leaders to identify the blockchain structure for specific use cases. Architects understand consensus mechanisms, cryptography systems, key management techniques and smart contract management. Qualifications include Go, Java, Javascript, or any modern programming language plus project management skills.

Blockchain Developer  
Developers design, write and upload smart contracts to the blockchain. Developer design workflows that operate with blockchains and smart contracts. Responsibilities can include project design, smart contract development and implementation, front end applications design and application deployment. Skills a developer needs can include Microsoft SQL Server, Visual Studio, .NET, MVC, AJAX, SQL, C, Python, Solidity, C++, C#, Javascript, Node.js, JQuery, SOAP, REST, FTP, HTML, XML, XSLT, XCOD, Neural-networks, Regression, Agile Scrum, MYSQL. Solutions are developed for Hyperledger Fabric, Ethereum, Quorum Ripple, Corda and other distributed ledger solutions.

Blockchain Operator  
Once the blockchain solution is designed and built, operators can join to create the peer network mentioned previously. The role of operators (nodes) is to store copies of the ledger and keep their copy up to date by distributing transactions and new blocks throughout their network via peer-to-peer communications.

Blockchain Regulator  
Many businesses operate under regulations regarding how their data should be stored and processed. This is the same for blockchain solutions. Regulators rarely interact with the blockchain directly, rather oversee regulatory requirements that need to be considered in blockchain development. With governments researching Central Bank Digital Currencies, governing these blockchains will become increasingly more complex.

Blockchain Project Managers  
Project managers are the first persons in an organization who are contacted when a company wants to bring blockchain in house and adapt it to their technology platforms. Responsibilities include deciding if a blockchain solution is a good fit for a business situation. If the decision is to move forward, Project Managers would organize key players and resources needed to execute the blockchain project.

Data Storage and Processing Managers  
The blockchain provides distributed, immutable storage with built-in integrity checking, but certain limitations exist that have to be managed. Data decisions need to be made dealing with maximum capacity based on the standard block size and block rate. Policies need to address how to provide integrity verification for large amounts of data, should data be stored off-chain with a hash of the data stored on-chain. eMembers of the peer network must execute the code to remain in sync with the current state of the network. If smart contracts commonly require large amounts of processing power to complete, devices external to the peer network may be used to augment the processing power of the network.

Blockchain Data Scientists  
Blockchain has the ability to collect large amounts of data. The Blockchain Data Scientist collects, interprets, analyzes and manages data. Developing algorithms that can analyze patterns in large data sets, the data scientist provides insight into the data collected.

Blockchain Quality Engineer  
This position is responsible for ensuring quality in all areas of blockchain development, such as automation frameworks and tests, manual testing and dashboards to support mobile, web and platform engineering. A quality engineer will need to research and advise on blockchain tools and develop quality assurance (QA) automated test standards, as well as define, write and implement test automation strategies for load performance tests. Applicants may also need an engineering management MBA degree.

Blockchain Legal Consultant or Attorney  
A blockchain legal consultant is responsible for advising companies on how to structure blockchain endeavors. Advise on how to structure and govern initial coin offerings, some of which are now coming under greater regulatory scrutiny. Attorneys will also be tasked with developing legal partnerships and contracts as blockchain technology offers "smart contracts", which are self-executing based on previously agreed terms. Duties can include designing legal partnerships that would connect the crypto ecosystem with existing financial structures, and supporting various merger-and-acquisition activities, such as negotiating and drafting legal agreements and performing due diligence on blockchain projects.

Blockchain Web Designer  
A decentralized community needs to keep the community that supports it informed. When including the use of cryptocurrencies in your business and when launching blockchains, there will be a greater need for websites to inform customers what a company is offering. A blockchain web designer will need to come up with original concepts, creative graphics and "mind blowing" user interfaces and dashboards to engage a diverse and inclusive community.

Blockchain Consultants  
Blockchain consulting companies and blockchain consultants are tasked with advising companies on designing and implementing blockchain solutions. This includes the documentation and maintenance of solutions, as well as their initial implementations.

NFT Graphics Designer  
NFT stands for a non fungible token, a special token that represents a unique ID that is linked to a piece of crypto art that cannot be replicated and is used to verify ownership of a piece. The collectable world is jumping in head first and creating non fungible tokens that represent art, music, memorializing events, newspaper articles and record releases. Designers are needed to turn these moments/media into art for the thirsty NFT market.

## e. Who Is Using Blockchain?

There are a lot of really interesting use cases being explored with blockchain right now.

When you first get into blockchain, one of the areas you'll see people immediately jump to are healthcare, how can we manage our personal health information on the blockchain, and voting an election reform, and both of these are really, really interesting ideas.

The goal behind the folks looking at blockchain for healthcare is the idea that it would be great if your medical records could be accessible with your permission to anyone else, anywhere on the planet.

So, if you're halfway across the world on vacation and you fall ill and end up in a hospital, wouldn't it be great if the doctor treating you had full insight into your medical records, knew any allergies you had, or knew any prescriptions that you were already on?

So, that's a very, very big area, and this ties a lot into blockchain and Identity Management.

How do we safely identify and protect people's personal data on blockchain?

So, a lot of interesting challenges there that are being worked out...

Blockchain for voting and election reform is a big topic right now, especially with all the conversation lately about how do we ensure the integrity of elections and demonstrate that our results haven't been tampered with or altered.

This is an area where blockchain could add a lot of value.

And then, of course, we hear a lot about traditional use cases, areas that are getting a lot of traction in blockchain right now.

Probably one of the biggest are supply chain, value chain relationships, which are very, very common in today's economy.

We have very, very few products or services that are offered completely and solely by one organization, without the involvement of other vendors, partners, suppliers, manufacturers, third-party organizations, etc.

So, blockchain to track the supply, the origin of the goods and services that we consume is a big use case.

Walmart is really pioneering this right now with a blockchain solution that they're using to identify the provenance of meat, where does the meat that is sold in a Walmart store come from, and the ultimate goal here is that, when we do have contaminated foods, and we have to issue a recall, currently, because we just don't have good insight into that supply chain, we recall and destroy a lot more food than we really need to, which is tremendously wasteful, especially with all the people on the planet that we could be giving that good food to.

So, these are some real, real exciting areas that we see being pursued in blockchain right now.

On the public sector, there's a lot of interest in blockchain for law enforcement purposes.

How do we allow different law enforcement agencies at the federal, state, county, city level, all to share data seamlessly, in a safe, secure fashion, which also protects people's personal privacy and anonymity, when appropriate.

Another area where blockchain is getting explored and is generating a lot of excitement is with artists, with people who produce content.

So, if you're an author, if you're a musician, if you're an artist, traditionally, you've only been able to get your work out to the public, in the hands of the public, by going through a third-party intermediary.

So, if I'm an author, traditionally, the only way I've been able to share my creation with the general public has been going to a publishing company and getting my works published.

Blockchain enables lots of interesting scenarios where I can sell that work directly to consumers, create new subscription models, and this is having very interesting effects on the work itself.

One of the things we're seeing is that traditional book publishing, there's anywhere between 12 to 18 months delay between when the author finishes the last word of a book and when it actually ends up into the consumers hands.

Well, if we can use blockchain to enable consumers to purchase this material or subscribe directly to the content creator, we take that lag time completely out of it.

And this allows authors and content creators to do some really interesting things.

You could write a fictional story and have one of your characters respond real-time to a news story that actually happened last week.

It makes for very compelling reading and it's just something we're not able to do today in the traditional publishing world because of that delay.

We also see a lot of interest in blockchain for fractional asset ownership scenarios.

So, we know we're 10-15 years away from having self-driving cars be a very regular thing, and when we all get our own self-driving car, if we're smart, we're gonna have that car take us to work and then turn around and perform ride-sharing services throughout the day, when we're not using it, in order to recoup some of that investment.

Well, in a blockchain world with coins and tokens, we can fractionalize that asset ownership, and I can take that same money I used to buy one self-driving car which operates in one city, and I could buy one percent of a hundred self-driving cars in the hundred's biggest cities on the planet.

And so, if they're ever weather conditions, where maybe my car couldn't operate locally, that's okay.

I have a much more diversified investment, which continues to offer returns, because I have a virtual guarantee that any major city that I have one percent of my car ownership in is currently in rush-hour right now, in peak times of people needing a ride.

So, there are lots of very very interesting use cases for blockchain in the real world right now.

I would encourage all of you to dig in a little bit more, and see some of the interesting things that are going on with blockchain in healthcare, in public sector, in insurance, in supply chain, value chain and more, see what other folks are doing, and then go out and make your own blockchain dreams a reality.

## Blockchain Users: Business to Consumer (B2C)

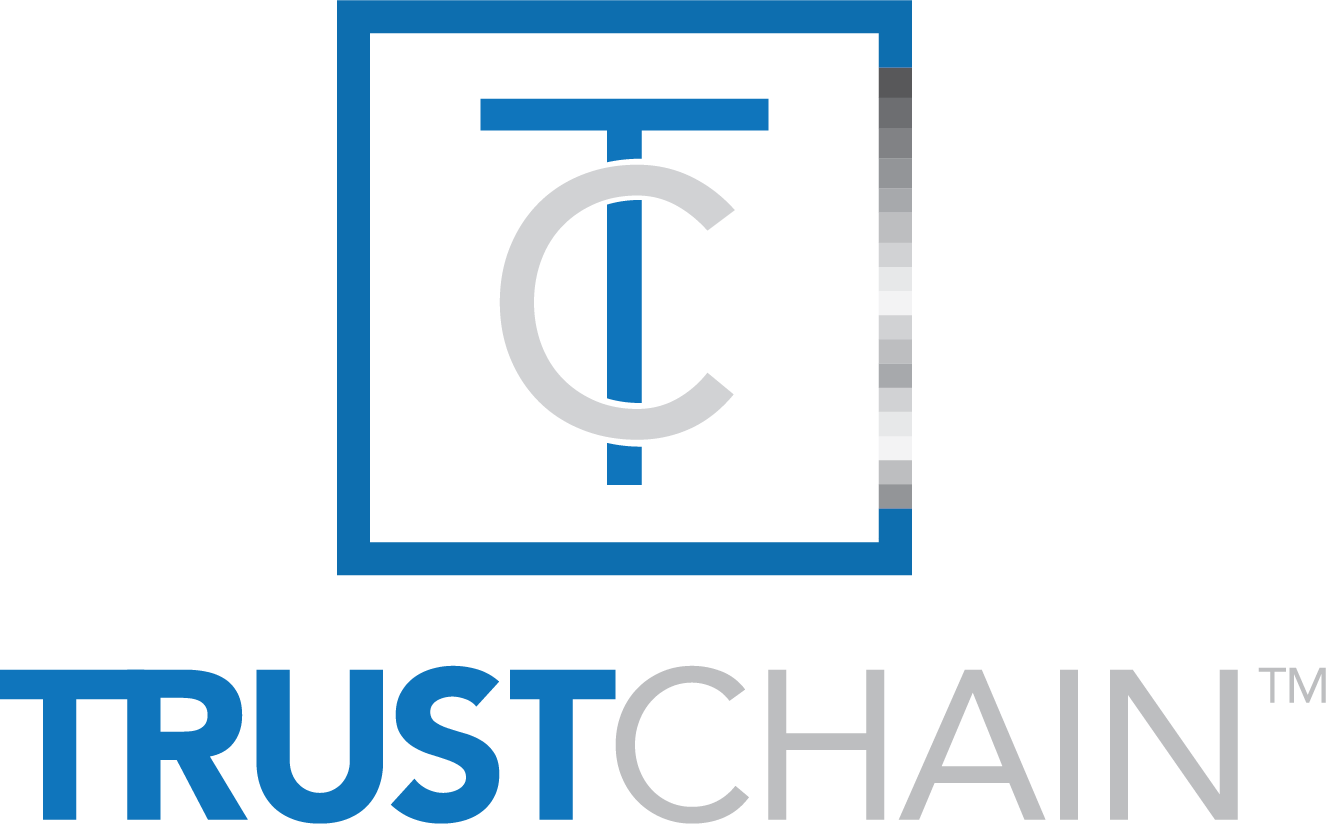
A Business to Consumer (B2C) is business or transactions conducted directly between a company and consumers who are the end-users of its products or services. Services being provided to the consumer is an area of interest to companies. Benefits to the consumer are transparency to the consumer, accountability from suppliers, ethical sourcing validation, labor verification and an immutable shared view.



Vertical markets see the advantage to the transparency feature in blockchain. Financials, logistics, charity funding, agriculture and ethical sourcing tracking are all markets leveraging blockchain to offer new insights to the customer.

## Real World Example of Supply Chain Business to Consumer

#### ****Use Case: Diamond Tracing****



 Approach: The blockchain collaboration, known as [TrustChain](https://www.trustchainjewelry.com/" \t "_blank)[TM](https://www.trustchainjewelry.com/" \t "_blank), will track and authenticate diamonds, precious metals and jewelry at all stages of the global supply chain, from the mine to the retailer/consumer.

Result: The jewelry industry can now track and authenticate diamonds and precious metals through every stage of the supply chain. Blockchain provides physical product verification, process verification and servers as an ethical oversight.

## Blockchain Users: Business to Business (B2B)

With a decentralized approach, blockchain removes the middleman (such as banks and brokers) to facilitate faster and less expensive payments. The shared ledger approach also eliminates the need for multiple paper copies that need to be verified and updated frequently.



Industries like finance, international trade and commerce, food supply chains are taking a serious look at blockchain.

## Real World Example of Business to Business Transaction

#### ****Use Case: Multi-party Invoicing DLT Labs****



* *Approach:* [DLT Labs](https://www.dltlabs.com/) along with Walmart Canada initiated a tracking program tackling the enormous problem in the freight and shipping industry of dispute resolution in disparate invoicing systems. The solution connects all points on the supply chain with a shared trusted ledger. This eliminates the root problems that cause invoice disputes. The carriers are a vital link in Walmart's supply chain backbone, and the DLT solution ensures invoicing systems run smoothly.
* *Result:* Demonstrable transformation of freight management and carrier relationships by implementing a shared DLT invoicing system with trust and transparency build-in, reduces disputes among organization by 95%.

## Unintended Use Cases

One of the coolest things about blockchain is some of the unintended consequences a good blockchain solution might have.

Let's look at an example.

Right now, there is a lot of attention being given to how blockchain might be used, in general, for supply chain relationships and, specifically, in the food supply chain.

Let's pretend that you have a very young child and your child has a serious pesticide allergy.

So, of course, when you go to the grocery store, you make sure that you only buy the produce in the produce section that has the USDA organic label.

It's just a little sticker that goes on the produce.

So, you buy your produce and you stand in the checkout line, and you feel pretty good that the produce you're buying actually is organic and has been treated with all-natural pesticides, and won't make your young child sick.

But sometimes you wonder, standing in line, how do I really know? How do I really know that someone didn't just print up a bunch of these stickers in their basement and apply them at the last minute?

The blockchain can go a long way to solving that problem and providing much greater insight.

Let's say, you're standing in the grocery store one day, and instead of the easily forged stickers, you're able to pull out your phone and scan a QR code.

When you scan that QR code, instantly on your phone the entire lifecycle of that piece of produce pops up.

You can see the date that it was planted on the orchard.

You can see after it was planted that an organic pesticide company came along and treated it with an all-natural pesticide.

You can even verify that the pesticide is natural by seeing the ingredient list in that pesticide.

And finally, you may see who actually delivered that produce to the store.

Let's say that there's a new green electric trucking company out there - Ed's Electric Trucking.

Ed has seen all the fancy new Tesla semis. He thinks they're great and he is committed to providing a transportation and logistics company which is also environmentally friendly.

Now, in today's world, you don't have a way of knowing about someone like Ed and his electric trucking company.

There just aren't good ways for folks in the B2B space to communicate their business value directly to consumers.

This is an interesting potential byproduct of using blockchain and smart contracts as a business solution.

Not only would I, as a consumer, have greater insight into the origin of that produce and feel much better standing in the checkout line, but maybe what Ed's doing resonates with me personally.

Maybe I'm a very green-conscious consumer.

I didn't know there was such an organization as Ed's.

And I have to buy a few more things while I'm at the store, besides produce. I have to buy some deodorant, some shaving cream, and some laundry detergent.

So, when I buy that deodorant, shaving cream, laundry detergent, I only buy the ones that were delivered by Ed's Electric Trucking.

This gives Ed and other B2B players a new and exciting way to establish brand and gain mindshare with the consumer that they currently don't have.

Certainly, it was not the original problem we went to solve with blockchain, but many times, when we apply blockchain to a real-world problem, we find out there are a lot of unexpected benefits just like this.

So, look for blockchain not just to transform the areas that we're aware it might make a difference in, but also to provide some new exciting and unintended benefits.

## Blockchain Users: Government to Consumer (G2C)/Government to Government (G2G)

The public sector is the part of the economy composed of both public sector and private sector enterprises that are operating on behalf of the public. It is the business of taking care of the people. Public sector includes all government, military, police, public health and welfare, infrastructure, etc. Public sector organizations leverage blockchain technology to move away from siloed and inefficient centralized systems. Current systems are inherently insecure and costly, with duplicated and in-efficient workflows. Blockchain networks offer more secure, agile, and cost-effective structures.



## Dubai Use Case

So, when people first start getting into blockchain, one of the areas they really start to look at is where is blockchain being used in the real world, what examples do we have of a blockchain being deployed and at what scale.

And there's really no better example of this right now than what's going on in Dubai.

Dubai, the largest city in the United Arab Emirates, is piloting a very ambitious program called smart Dubai 2021.

And what Smart Dubai 2021 aims to do is to move all governmental operations and record-keeping on the blockchain by the year 2021.

The goals of this program are increased transparency in the government operations, increased efficiency, and by proxy, providing better service to city residents and constituents and establishing Dubai as a thought leader, an industry leader in the blockchain space.

There are lots of really cool applications that tie into this.

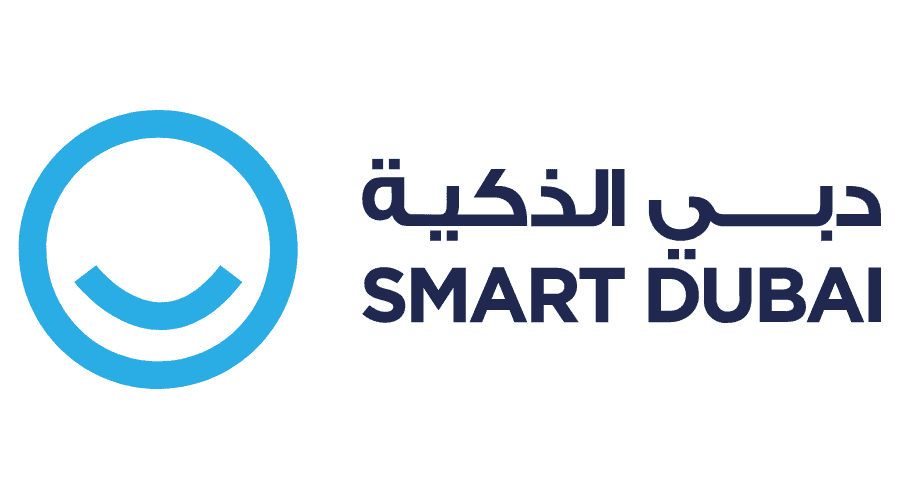
Blockchain is becoming the enabling layer for a lot of other Smart Dubai initiatives around creating a smart city.

So, projects like automated traffic signals, registration of citizens for governmental services, etc. are all being pioneered on blockchain.

So, if you want to see some of the really exciting use cases in blockchain right now, be sure to check out Dubai and what's going on with their Smart Dubai 2021 initiative.

## Real World Example of Public Sector Government to Government/Citizens

#### ****Use Case: Public Sector - Smart Dubai****



* *Approach*: To make Dubai the happiest city on Earth by recreating everyday experiences for residents and visitors of Dubai, making them more personalized seamless, efficient and impactful. The [Smart Dubai](https://www.smartdubai.ae/) strategy will be using three essential pillars: government efficiency, industry creation and international leadership
* *Result*: By the end of 2021, Dubai government will go completely paper-free, eliminating more than one billion pieces of paper used for government transactions every day. When successful, Dubai will be the first blockchain-powered government driving the future economy.

## More Use Cases

Blockchain is having far reaching effects in major industries. Let’s take a look at three major industries having a positive influence from this new way to organize our information:

* Supply chain  
  Blockchain is used in supply chains to gain transparency and traceability. This has a special advantage for global health when used in traceability to the food supply chain. By tracking food on a blockchain, you are building trust into the system through the transparency feature. In the event of a problem, blockchain technology can pinpoint the exact location and cause of a food health issue. By instantly tracing food issues back to the source, all contaminated lots can be tracked and dealt with instantly.
* Global trade  
  Blockchain is making headway in the global trade markets with transparency and immutability that offer new relationships among distrusting parties. Blockchain makes sure all relevant parties have access to the same information. Trade barriers are reduced, from settlement issues, to compliance and regulation, information is being shared and accessed in new secure ways. A distributed ledger technology (DLT) opens up a new avenue for securely sharing data that guarantees accuracy while removing banks as deal facilitators. Smart contract and microfinancing alternatives allow all size companies to seamlessly trade across borders.
* Digital identity (self-sovereign identity and verifiable credentials)  
  Users of the Internet were all but unaware that all the information given so freely when interacting with companies online, was being stored and sold. Data became a siloed commodity owned by everyone but you. With new blockchain technology, individuals can reclaim their identity information breaking the need for data silos, endless passwords and hackable databases. Self-sovereign identity (an individual identity holder controls their credentials and can use them without permission) gives secure identity verification designed to bring trust and personal control so people can prove things about themselves using trustworthy verifiable digital credential.

## f. Chapter Summary

In this chapter, we discussed:

* The history of the Internet and how blockchain technology relies on Internet technology as the foundation for building the future of the blockchain evolution.
* The difference between blockchain and Bitcoin.
* How blockchain functions.
* Blockchain concepts and characteristics: distributed ledger, peer to peer, transparent, consensus, encryption, immutability, programmable.
* Blockchain users and their roles.
* Emerging general use cases.

# [Chapter 2. Blockchain Mechanics](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@chapter+block@90a929834b3b4bc18c6b3e7c662f152c) / [Introduction](https://learning.edx.org/course/course-v1:LinuxFoundationX+LFS170x+2T2021/block-v1:LinuxFoundationX+LFS170x+2T2021+type@sequential+block@1088154e18e0450f9bb3a418121008c1)

## Chapter Overview

In this chapter, we will discuss distributed ledger technology, some of its basic components and how it differs from previous ledger storage techniques. We will gain an understanding of how previous computer technologies like file sharing, data storage and cryptography are wrapped into the blockchain foundations. We will also see how hashing functions and Merkle tree data structures help us immutably store and prove facts without revealing all the details.

Next, we will learn how these blockchain techniques are fundamental in public-key cryptography where key pairs support privacy and authenticity by only exposing the public key while keeping the private key secure. We will examine how this agreed-upon, append-only distributed ledger guarantees transparency and immutability, where trust in the information contained in the ledger is a given.

## Learning Objectives

By the end of this chapter, you should be able to:

* Discuss how cryptography allows for distributed ledgers to work on a global scale.
* Explain how blockchain is using cryptography.
* Define key terms of cryptography and understand basic concepts.
* Explain the concept of zero-knowledge proofs (ZKPs) and Merkle trees and how they are used in blockchain technologies.
* Discuss how blockchain utilizes transparency.
* Explain what it means for blockchains to be immutable and how chaining provides this capability.

## Introduction to Blockchain Components

In this section, we're going to discuss the core components of the blockchain, starting with a high-level view of the blockchain components.

The first blockchain component that we're going to look at it's the ledger.

However, before we discuss the ledger, let's discuss briefly the history of the ledger.

The story of blockchain is tightly coupled with the story of accounting.

Historically, humans started off with no way to prove ownership and we began with a single entry accounting system.

The single entry accounting system for the first time in human history allowed us to prove ownership of the asset.

The ledger was associated with an owner.

The single entry accounting worked for centuries.

The issue with single entry accounting is that it mandated that there was a single authority, which is the reason why there was the necessity for a king or a queen to control the ledger.

In order to have trade, right, at the international level, we needed to have at least two authorities.

For instance, for England to do trade with France, we had the owner of the ledger, the single entry ledger, in England for instance, doing trade with the king or queen of France, who also had their ledger.

And so, we needed a new form of accounting, and that's where double entry accounting came in, which was in use up until very recently, now, within the last 40 years.

What blockchain is? Blockchain is the very first implementation of triple entry accounting, where we have an asset being recorded on the ledger in the context of a transaction.

The third entry and triple entry accounting is cryptography, where we have a cryptographic account of the transaction stored permanently and immutably on the ledger.

That's what the ledger is. A ledger is a collection of transactions.

It is not a collection of assets. Assets are part of a transaction, but the ledger records the transaction.

In blockchain, the differentiator is that no one owns the ledger, or all of the participants own the ledger.

The ledger is distributed. It is, in other words, it's decentralized. So, there's a copy of the ledger that exists on every node that exists on the network.

Said simply, the ledger is a distributed immutable record of a collection of transactions.

Bitcoin is the most popular asset.

It was the first asset to be recorded as a transaction on a blockchain ledger, and it remains the most popular, at least in terms of market share.

That's what the ledger is.

As we move to more modern blockchains, we start to look at blockchains such as Ethereum, which not only records the asset on the blockchain,

Ethereum and other public blockchains like Ethereum, they also allow you to have a permanent and immutable collection of code, also known as a smart contract, that runs on the blockchain.

The ledger stores the assets, the transactions that are on the blockchain, and it also holds the code.

The code that's stored on the blockchain is a smart contract.

Again, a smart contract is a program that runs on the blockchain.

The blockchain is a network.

The blockchain is a ... A public blockchain is equivalent to the Internet, complete with its own set of protocols, etc.

Private blockchains are more synonymous with an intranet.

It is... just like we have a use case for public blockchains and public internet, we also have use cases for private blockchains and an intranet, if we're gonna stay with that analogy.

On each of these networks there are nodes. The nodes are going to be synonymous with the computers that make up the network.

On the nodes, they are put together by a collection of protocols.

Each blockchain has its own set of protocols that defines not only the nodes, but how those nodes communicate with each other.

So, when you hear the term "peer network", that's what we're talking about; we're talking about how that blockchain is put together, what are the protocols that define the peers, and how those peers interact.

Those peer networks, they store the ledger, they provide the updates, they effectively, they maintain the ledger.

That's what a peer is on a peer network.

On a public blockchain, it's public, and, by definition, that means that anyone is allowed to join and participate in that public blockchain.

On the private blockchain, however, the situation is a little different.

On the private blockchain, we have to control who can access your network.

Just like no company would allow... no company would allow anyone to access their private intranet, in a private blockchain you have to be asked to join, or you have to have permission to join.

A membership service is a gateway that allows users and components to enter a private blockchain.

A membership service or a membership service provider provides the services of user authentication, user authorization, component authentication, component authorization, more specifically Identity Management.

If we're talking about a private blockchain, then we have to control who the members are, and not only the members.

We have to also control the nodes that access the blockchain.

As we're talking about membership, one of the key components of membership is we have to allow certain users to access the blockchain.

In order to do that, we have to have a definitive identity for each user.

In the blockchain, that's achieved in two ways.

One, we use PKI, to provide a public key. And that public key represents the identity of the user.

But we also need a private key that each user can hold to prove their identity.

That private key is stored in a wallet. And a wallet is a collection of the user credentials. It is effectively ... it encapsulates private keys.

Additionally, in the blockchain... blockchains are self-contained, or they should be self-contained.

And so, one of the key aspects for developing on a blockchain or using a blockchain is we have to have a way to interact with the blockchain.

And one of the ways that we do that is through events.

So, when something occurs on the blockchain, either public blockchains... most public blockchains anyway, and certainly all private blockchains, they have events that are fired when certain key actions are triggered.

This allows end-users, other systems, other users, different components to be able to act off of the events that come from an update on a blockchain.

## Basic Terms

Some terms that are mentioned in the video we will get familiar with in later chapters, but are worth a brief description for clarification:

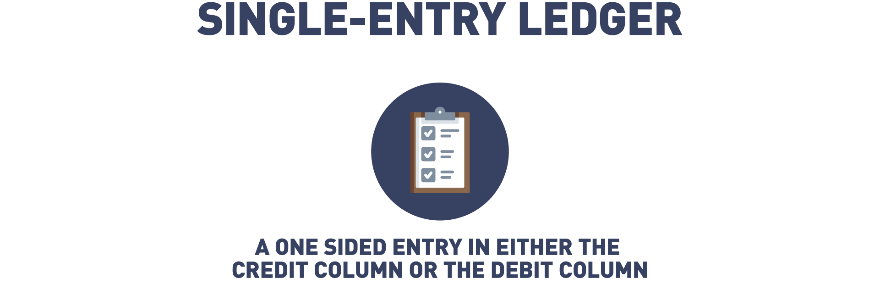
* *Ethereum*  
  Ethereum, like Bitcoin, is a decentralized open-source blockchain. Unlike the Bitcoin Network, the Ethereum Protocol has the ability to run smart contracts. The cryptocurrency running on the Ethereum network is called Ether.
* *Smart Contracts*  
  A smart contract is computer code that can be run on a blockchain that executes the conditions of an agreement.
* *Public Blockchains*Public blockchain has no barrier to entry. Anyone with a cell phone and a digital wallet can transact on the blockchain.
* *Private Blockchains*  
  Private blockchains or permissioned blockchains have a Membership Services piece that grants access to users before they can interact with the blockchain.
* *Digital Wallet*A digital wallet is a computer application that represents a traditional wallet. It gives the user the ability to store and transact cryptocurrencies through the Internet.
* *Public Key Infrastructure (PKI)*  
  Public Key Infrastructure is the identity management system for instilling trust into the electronic transfers of information or value. It is the technology used to authenticate users and information by issuing a set of key pairs (public and private).

## b. Ledgers and Basic Accounting Systems

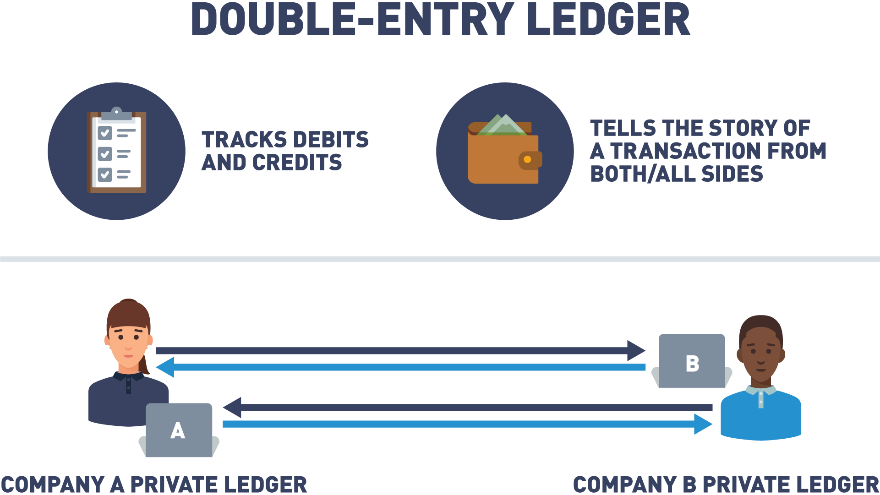
To understand how blockchain came about, we must first understand the following key principles:

* Ledgers
* Basic accounting principles in history

## History of Ledgers



* Ledgers first appeared around 3,000 B.C.
* Single-entry only.
* Chanakya, an Indian leader, creates the first documented accounting standards.



* Double-entry ledger appears in 1340 A.D.
* Tracks debits and credits.
* Tells the story of a transaction from both/all sides.
* The Italian Luca Pacioli, recognized as the father of accounting and bookkeeping, was the first person to publish a work on double-entry bookkeeping and introduced the field in Italy.



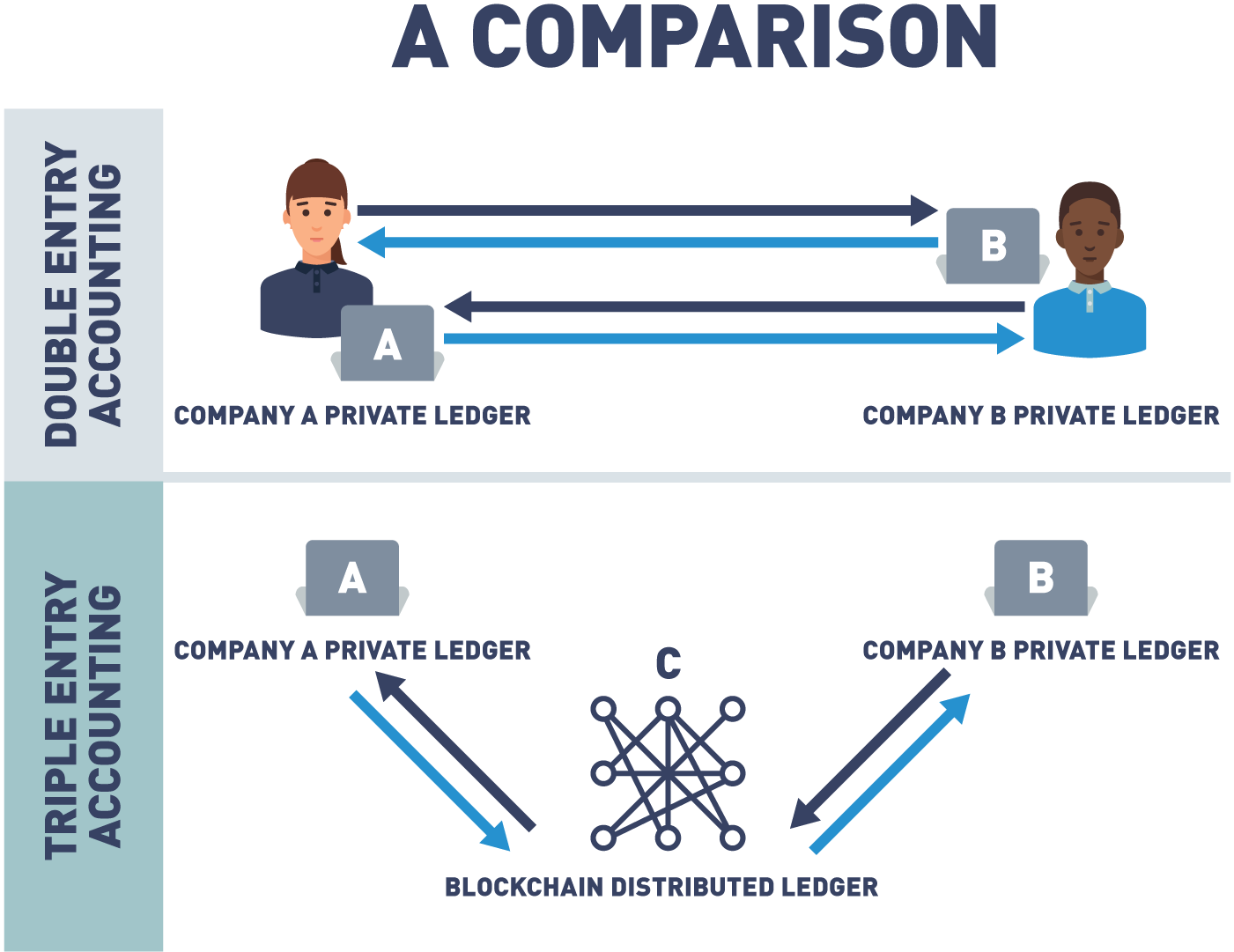
* Triple-entry accounting is an enhancement to the traditional double-entry system, in which all accounting entries involving outside parties are cryptographically sealed by a third entry.
* Debits, credits, and an immutable link to all past debits and credits.
* Triple-entry ledger appears in 2008 in a white paper by Satoshi Nakamoto (a.k.a., Blockchain).
* Some features include: tamper-proof records, distributed ledgers, double-entry+cryptography, validated, secure, and private, digitally signed receipts.

## History of Ledgers (Triple-Entry Example)

As described in the [Reddit post](https://www.reddit.com/r/Accounting/comments/2biy4h/triple_entry_bookkeeping_using_cryptocurrencies/),

"A seller books a debit to account for cash received, while a buyer books a credit for cash spent in the same transaction, but in separate sets of accounting records. This is where the blockchain comes in: rather than these entries occurring separately in independent sets of books, they occur in the form of a transfer between wallet addresses in the same distributed, public ledger, which creates an interlocking system of enduring accounting records. Since the entries are distributed and cryptographically sealed, falsifying them in a credible way or destroying them to conceal activity is practically impossible".

## Double vs. Triple-Entry Accounting Comparison



## Island of Yap

Idea for the following was obtained from the LinkedIn article [*"Explaining Blockchain: The Distributed Ledger of the Island of Yap"*](https://www.linkedin.com/pulse/explaining-blockchain-distributed-ledger-island-yap-riva/) by Anthony (Tony) Riva.

The concept of distributed ledge's has been around for hundreds of years.

To understand the basics of blockchain, it's important to understand how distributed ledgers work.

A distributed ledger is a collection of data with no central administrator that has been agreed upon by consensus.

To understand more details, let's take a trip way back, back into 500 AD on the island of Yap.

The Yapese people used a very unique form of currency: the Rai stone.

Each of these stones weighed around 200 kilos, making them difficult to move.

To exchange them as money for goods or services, they divided the stones into sections and then, they would announce to every adult on the island who owned what part of each stone.

Each adult had to keep a mental ledger of ownership.

Every time any Yapese conducted a trade, an announcement was made to the entire tribe.

Each member of the tribe would then update their mental ledger.

In today's description, this would be called a distributed ledger.

All Rai stone ownership was known to everyone, and that knowledge was updated whenever a transaction was made.

So why didn't just one member of the tribe keep track of the Rai stones?

What if that main recordkeeper was sick, unable to do their job, or was found to be dishonest?

If the only copy of the ledger was changed by any means, wealth would be lost or gained unfairly.

The Yapese knew their distributed ledger system safeguarded it from tampering.

Since all the Yapese knew who owned what, it would be very difficult to fool everyone on the island.

Even if one tribe member moved away, the tribe still had multiple copies of their mental ledger.

In this regard, the ledger was fault tolerant and could not be easily changed, corrupted, or lost.

The tribe also decided that stones didn't have to be on the island to hold value.

One day, a stone fell into the ocean and the Yapese decided that this stone, even though it could not be seen, still held value and could still be traded.

This system of consensus by the majority of the adults on the island with no central administrator is one of the first examples of distributed ledger.

## Island of Yap: Recap



The group shared the ledger.



The group agreed to the ledger.



 All changes or transactions to the ledger were announced to the group.



 The group agreed to update all ledgers.

Let's review an example on Yap Island:

Alice agrees to trade Bob her stone by the pond in exchange for all of his cattle.

Alice and Bob announce their transaction to the tribe.

Everyone updates their mental ledger.

From this point on, they agree that the stone by the pond is owned by Bob.

## Island of Yap: Corruption Is Near Impossible

Alice tries to corrupt Carol, so that Carol’s ledger shows that Alice never gave up ownership of the stone.

* Centralized ledger, only one place to cheat.
* Decentralized ledger, Carol will be outvoted by the rest of the tribe, and her version of the ledger will not be accepted.

If Alice wants to cheat, she will need a way to convince 51% or more of the tribe to accept an alternative ledger.

## The Evolution of Distributed Ledgers

We've gone through history and seen how ledgers have evolved from stone tablets to double entry ledgers.

We even took a look at the earliest known example of distributed ledgers.

Now, let's discuss the evolution of the ledger in the modern world.

Over the last few decades, ledgers have moved from the written form to the digital form, also known as computers and digitalized databases.

This has allowed us to be interconnected - a truly global economy.

Today, this recordkeeping system is being revolutionized again,

going from the digital realm or centralized databases keeping track of transactions to information being stored across different locations across the globe.

This system of record, information across a global network, is what we call decentralized ledgers.

A decentralized ledger can be described as a ledger of any transactions or contracts supported by a decentralized network from across different locations and people, eliminating the need of a central authority.

When a recording is made, every participant is notified of that recording.

They now own an identical copy of the record of transactions, identical to every other node on that network.

Every piece of information on a blockchain can be traced back to the exact moment when it was created.

So, if an audit is necessary, information can be found and deemed reliable.

This allows blockchain to serve as a truly reliable storage of public records and transactions, and therefore eliminating the need for a centralized authority of database of records.

Now that we understand the foundational principle of blockchains - distributed ledgers, it's important to understand another important key element that makes blockchain so revolutionary: trust.

Trust is created through two key principles: collaboration and cryptography, both of which will be discussed shortly.

## Decentralized Ledger

As we can see by looking at the evolution of the ledger, we are entering a new phase where no one centralized authority has control of the details in the ledger, everyone does. Transactions are only added to the ledger if everyone agrees. Once consensus is reached, no one can go back and deny a transaction. So essentially you are not trusting one person or entity to control the ledger, you're trusting everyone. Let's look a little further into how that trust is achieved.

## c. From Distributed Ledger to Cryptography

A great way to think of blockchain is to think of blockchain as another layer of the Internet, a layer that enables secure and trusted records and transactions to occur, which is why a lot of people associate the word blockchain with the word trust.

Blockchain creates a layer of trust between untrusted parties.

Without blockchain to create trusted records and transactions, a third party intermediary is necessary.

Blockchain eliminates the need for a centralized institution to act as intermediary, and instead uses cryptography and collaboration to create that trust.

In an earlier chapter of this course we spoke about collaboration through distributed ledgers.

Now, we will be taking a further look at how cryptography is used in blockchain to create the second element necessary for blockchain to be a source of trust.

Information on the blockchain is stored on the ledger using cryptography.

It can be accessed using keys and cryptographic signatures.

Now, let's take a look into some of the key concepts of cryptography that are utilized with blockchain.

## Rose Greenhow

So, one of the reoccurring themes in blockchain is that while blockchain itself may be new and cutting-edge, it's built out of nothing more than very old tried-and-true technology.

And one of the areas this is most true is in cryptography.

When a lot of people first get introduced to cryptography they think it's a very new cutting-edge science.

And while there are new developments being made everyday, cryptography itself is a very, very old science.

One great example of this is the story of Rose Greenhow.

Rose Greenhow lived in the Washington DC area during the Civil War, and she was actually an undercover spy for the Confederate Army.

Rose Greenhow was a wealthy socialite, and she threw many fancy, elaborate parties, and was also invited to many, many high-end social events.

Many of these events were attended by members of the Union Army, generals and strategists, who Rose pretended that she didn't know much about military strategy and wasn't too interested in the war in general.

In secret, at all these events she was soaking up all the information she could.

Rose actually developed her own method of encrypting data.

She came up with her own encryption scheme and she used this scheme to mail letters down to the Confederate Army, revealing the plans of the Union Army, troop positions, general high-level strategy, etc.

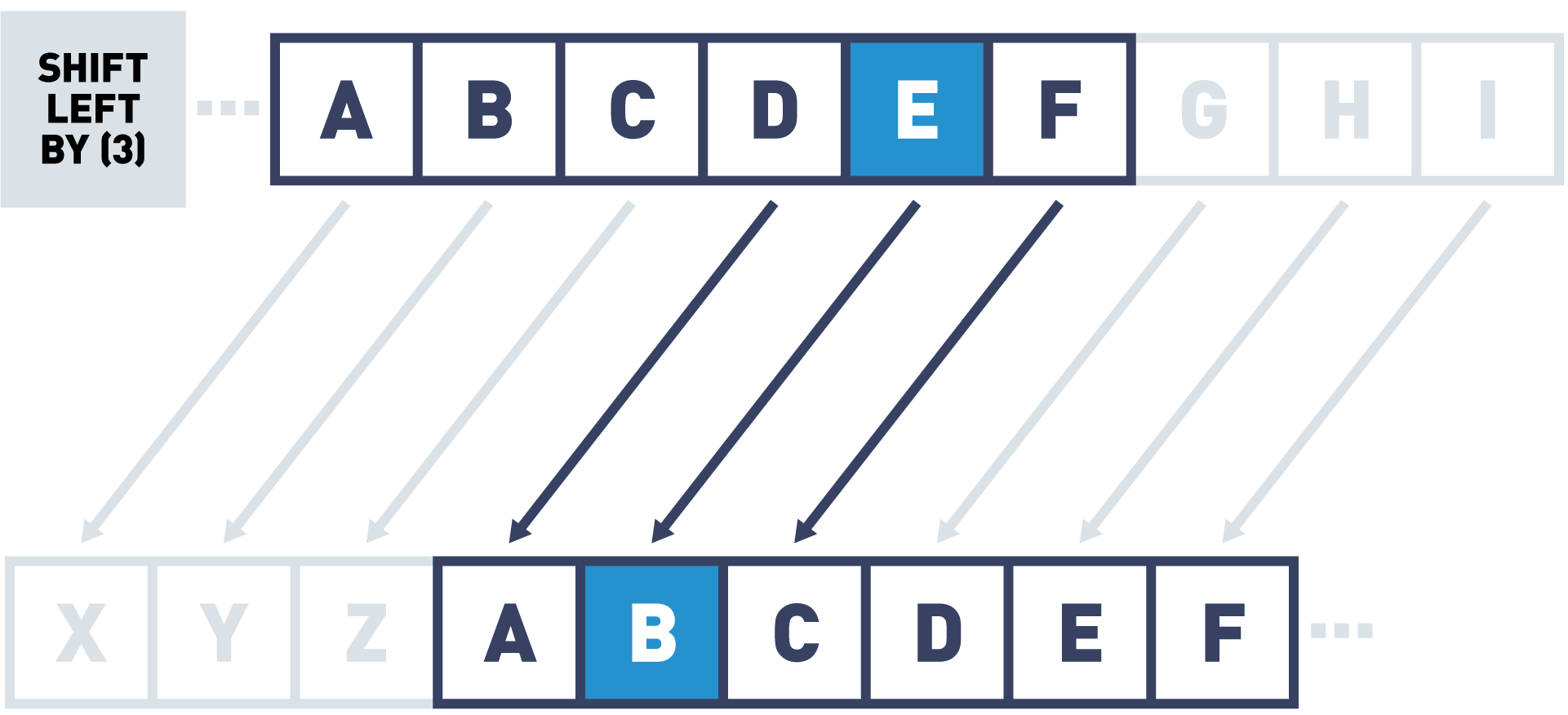
Rose did this for a number of years, until finally her encryption method was broken and she was actually one of the very first people who were arrested and prosecuted by the Pinkertons, who later went on to form the Secret Service.

So, when you hear about blockchain and all the new cutting-edge things that it's doing, don't forget that it has its roots in very old technology, and we're just combining old technology in new and creative ways to create a brand new solutions development platform.

## Early Use of Cryptography: Caesar Cipher

As described in the [*"Ancient Uses of Cryptography: Four Examples that Pre-Date the Internet"*](https://medium.com/tokenring/ancient-uses-of-cryptography-four-examples-that-pre-date-the-internet-14679ae4f509) article by Melanie Shapiro,

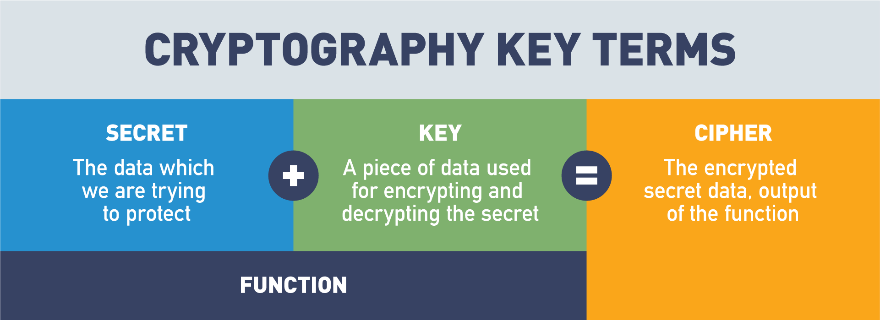
"A little over 2,000 years ago, Julius Caesar developed a simple system to send secure information to his troops. It was all about substituting certain letters for others, typically by shifting the letters by a predetermined number. That algorithm is what we would call a cipher, and since Caesar’s invention, cipher keys are much more secure and advanced. If you want to encrypt and decrypt information, you are going to have to choose a kind of cipher to do so".



## Cryptography Key Terms

Let's explain the key terms used in cryptography:

* Cryptography  
  Cryptography is a technique used to secure the communication between two parties from a third. The term cryptography is derived from two ancient greek terms, “kryptos” which means “hidden” and “graphein” which means “to write”.
* Secret  
  The data which we are trying to protect.
* Key  
  A piece of data used for encrypting and decrypting the secret.
* Function  
  The process or function used to encrypt the secret.
* Cipher  
  The encrypted secret data, the digital secret, the output of the mathematical function or a cryptographic algorithm.
* Encryption  
  This is the process from plain text (ordinary text) to cipher text (random sequence of bits).
* Decryption  
  Encryption is the reverse process of converting ciphertext into plain text.



 The Secret and the Key are passed into the Function to create the Cipher.

## Cryptography Basics

So, one of the technologies that plays a pivotal role in blockchain and making blockchain secure and immutable is cryptography.

If you've never heard of cryptography before, cryptography is simply designed defined as the science of sending messages back and forth from one party to another, in the presence of adversaries or bad actors, participants who may wish to corrupt or change the contents of a message.

And there's some key concepts you should understand about cryptography before we really dive in and talk about how cryptography protects the data on the blockchain.

Most cryptographic functions are what we refer to as two-way functions, meaning that we can take a secret and we can encrypt it, and then we can use that same process to decrypt the secret at the other end.

Some of the key objects or concepts to be aware of in cryptography:

first of all are the secret, what data are we trying to protect, what is the message we're sending back and forth that we want to do so in a safe, secure fashion, and try and guarantee that it hasn't been tampered with.

We take our secret and we combine it with a key, a piece of data that is consumed by a function, our third component.

When the function receives the key and the secret, the function uses the key to change the data in the secret, to create the fourth and final component, which is known as the cipher.

And the cipher is simply the encrypted message, the thing that, if it were intercepted, if it were seen by anybody else, any non-participants or bad actors, they wouldn't know what the secret was, they wouldn't know what the message was, and they wouldn't know how to decrypt that cipher back into the secret.

Now again, most cryptographic functions are two-way, meaning it doesn't do me a lot of good to encrypt data if the receiving party at the other end can't decrypt it.

And oftentimes, all this involves is using the same key or a reverse key into the same function, with the cipher's input to get the secret back out.

So, when we talk about cryptography, keep in mind those four basic components: the secret, the key, the function, and the cipher.

Then, understand that cryptographic hashing is a very special class of cryptographic functions, which, unlike all their cousins, are only one way, meaning that in a cryptographic hash I can put information or a secret into it, the key and the function are contained as one entity, and what I get back out is encoded data.

I get what's called a hash or a hash result, that is an encryption based on the data that I put in.

That is mathematically impossible to reverse-engineer and this is what makes it a one-way function.

If you put any data into a cryptographic hash and you share that hash with me, there's absolutely nothing I can do above and beyond random guessing to try and recreate that input to tell me what your original secret was.

Cryptographic hashes are used in scenarios where we want to prove a piece of data without sharing that data.

So, password management systems use this all the time.

We don't want to store a username and password…we want to store a username and the hash output of a password.

That way, I don't know what any of my users' passwords are, but I do know that if they put the wrong password in, they're going to get the wrong hash back out, and I can easily validate that a user's password is correct or incorrect, without having to know what it is.

We're going to dive in a lot deeper in a later module, to show you exactly through a demo how cryptographic hashing works, and how it provides security and immutability on the blockchain.

But just understand these two big takeaways:

Number one, that cryptography is normally a two-way function with four components: the secret, the key, the function, and the cipher, and then finally, there are a subclass of cryptographic functions called cryptographic hash functions, which exist as one-way functions, which can only encrypt the secret, and again, which are mathematically impossible to decrypt the secret in any other method other than randomly guessing input until you're able to recreate that hash output.

These are key components which enable security and immutability on the blockchain.

## Cryptographic Functions

Simple example function:

* Secret = "Blockchain technology is transformative"
* Function = Swap each letter in the secret with a new letter according to the Key
* Key = "+2"
* Cipher = "Dnqemejckp vgejpqnqia ku vtcpuhqtocvkxg".

As mentioned in the [*"Ancient Uses of Cryptography: Four Examples that Pre-Date the Internet"*](https://medium.com/tokenring/ancient-uses-of-cryptography-four-examples-that-pre-date-the-internet-14679ae4f509) article by Melanie Shapiro,

"(...), in the 1970s cryptologists [*Whitfield Diffie and Martin Hellman*](https://news.stanford.edu/2016/03/01/turing-hellman-diffie-030116/) made a landmark invention: asymmetric key encryption. It’s the concept that both HTTPS (the popular protocol used to access a secure web server discussed previously) and the secure element within a token rely on to keep your information completely private.

The principles behind it are genius. Instead of a shared key that codes and decodes information, the key for encrypting the information is different from the key that decrypts it — that way there is no longer a shared, secret key. With this invention, in order to share a secret message, you no longer even have to know the person you are sending it to. Most importantly, for people like us who care deeply about safe authentication and identity protection, the private key itself is never communicated at all, and that means no more shared secrets".

## Byzantine Fault Tolerance

As you start to get more and more into blockchain, one term that you're gonna hear a lot is Byzantine Fault Tolerance.

And if you're wondering what that means, that very strange, cryptic sounding term, that can be explained with a story.

In 1982, a mathematical problem was formulated called the Byzantine Generals Problem, and without getting too detailed into the mathematics behind it, essentially, with the Byzantine Generals Problem was a story about two empires fighting a battle.

One Empire exists within the confines of a walled city, the second Empire has several generals surrounding that walled city, each with their armies waiting to attack.

And the balance of power is very, very evenly matched between these two empires.

And it is such that if all the generals agree to attack at the same time, they will have enough power to overcome the empire in the city and win the campaign.

If they, for whatever reason, they're not able to coordinate their attacks simultaneously, and even one general misses the message, well then, the Empire surrounding the walled city is going to lose the campaign and the war.

So, the Byzantine Generals Problem explored a really key area of cryptography, in fact, the heart of what cryptography is, which is how do I exchange messages with someone in an environment where they may become corrupt.

You have to understand the only way these generals had to communicate was to send a disguised messenger through the city with a message to deliver to the other generals,

and so, this always posed the risk, if you're a general and you're getting amessage from one of these messengers, how do you know or how can you believe with greater than random certainty that the message you're getting is reliable and it hasn't been tampered with.

And this, again, was a problem that was postulated back in 1982 and went unsolved until 2008, when the anonymous Satoshi Nakamoto proposed a solution to this problem.

Essentially, what the solution entails is all of the generals hiring as many mathematicians as they can.

When we talk about Byzantine Fault Tolerance, we're talking about the generals having more access, having more mathematicians, or more computing power on their side than the empire within the city.

So, if I'm the generals and I can hire more mathematicians, then that means that I can encrypt my messages going through the city with greater and greater complexity, and if the balance of power becomes greatly tipped in the favor of the generals outside the city,

they can use encryption methods which are so complex, that there really is no real chance, or a very small chance that any of the mathematicians inside the city will be able to intercept that message, decrypt it, change the contents, re-encrypt it, and do so in a time quick enough that the generals don't notice anything has happened.

And so, this is Byzantine Fault Tolerance.

This is the fault tolerance mechanism that we see used in most public blockchain solutions right now, including Bitcoin, including Ethereum, although Ethereum is working on transitioning to a system called Proof of Stake, which we'll talk about in a later section.

But, when you hear Byzantine Fault Tolerance, often abbreviated as BFT, all we're talking about is this idea that blockchains become secure if they have more computing power, more hash power in the chain than off the chain.

Solutions like Ethereum, 16,000 plus nodes, or Bitcoin, the largest blockchain network that there is, with over 30,000 nodes, offer us a high degree of security and immutability, because the computing power on those networks is so great that there's really no reasonable chance that an adversary could ever control as much or more computing power to attack the data on that blockchain.

And if someone ever were to, chances are the economics behind it would be such that they would spend more to simultaneously harness all that computing power, than they could possibly stand to gain.

And so, this is Byzantine Fault Tolerance.

It is the root of cryptography, which is simply the study of how do we send information back and forth securely in the presence of adversaries and bad actors, and it's what underlies all the security and immutability that we get in blockchain.

## Types of Cryptography in Blockchain

To understand cryptography in blockchain, you must be able to differentiate between various cryptographic techniques used. There are four main ways blockchain leverages cryptography:

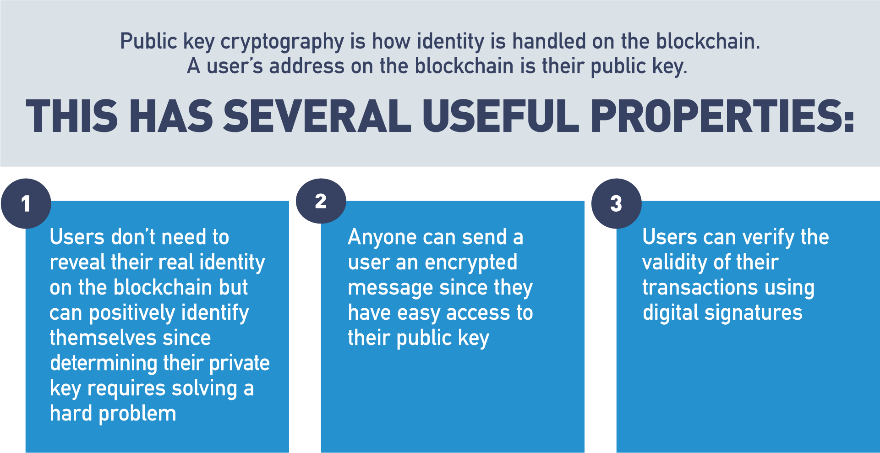
* Public Key Cryptography  
  This encryption method uses a pair of keys: an encryption key, and a decryption key, named public key and private key, respectively. The key pair generated by this algorithm consists of a private key and a unique public key that is generated using the same algorithm.
* Zero-Knowledge Proofs  
  A new approach to protecting digital secrets that prove knowledge of a secret without revealing it.
* Hash Functions  
  This type of encryption doesn’t make use of keys. It uses a cipher to generate a hash value of a fixed length. The function converts plain text (no matter the size) into a hash of fixed size. It is nearly impossible for the contents of plain text to be recovered from the cipher text. Think of it like trying to recreate a human from a fingerprint, a fingerprint uniquely represents a human no matter the size of the human and you can’t reverse engineer a fingerprint to recreate the human.
* Merkle Trees  
  A data storage technique that compresses or packs data for storing blockchains with a tamper-free component built in. Merkle trees are built upon hashing principles in that each hash becomes a part of the next hash to build a tamper resistant data storage model.

## Public Key Cryptography

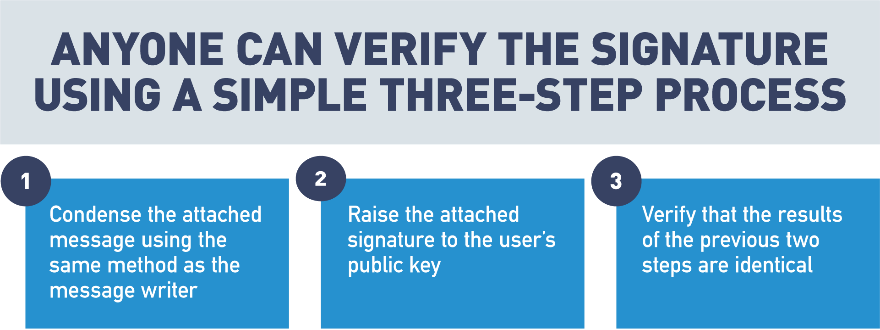
Identity in the blockchain is based on public key cryptography. A person’s address on the blockchain is their public key.

Transactions on the blockchain include their public key and are digitally signed with the sender’s private key:

* A digital signature verifies that someone in possession of the private key authorized the transaction.
* Digital signatures can be easily verified using the corresponding public key, which is included in the transaction.



## Identity: RSA Public Key Cryptography



## Identity: Specific Identity Implementations

Ethereum is a public blockchain that anyone can participate in. [Hyperledger](https://www.hyperledger.org/) is an example of an enterprise blockchain where participants must be granted access to engage in the blockchain.

* Ethereum  
  A user’s identity is an address based on their public key.
* Hyperledger  
  Identity is managed by X.509 certificates. Certificates are only shared with parties involved in the transaction.

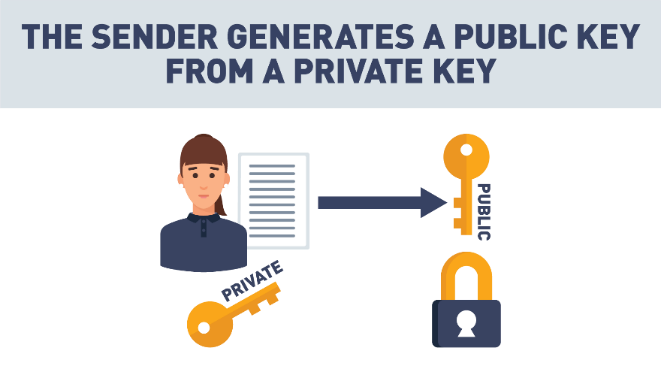
Public key cryptography uses a pair of a public key and a private key to perform different tasks. Public keys are widely distributed, while private keys are kept secret.

Using a person's public key, it is possible to encrypt a message so that only the person with the private key can decrypt and read it. Using a private key, a digital signature can be created so that anyone with the corresponding public key can verify that the message was created by the owner of the private key and was not modified since.

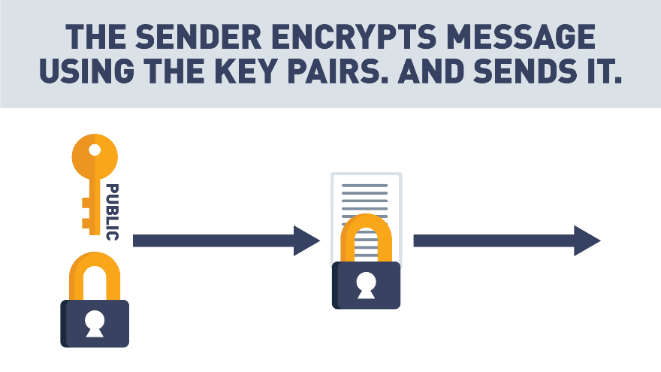


Digital Signatures and Key Pairs

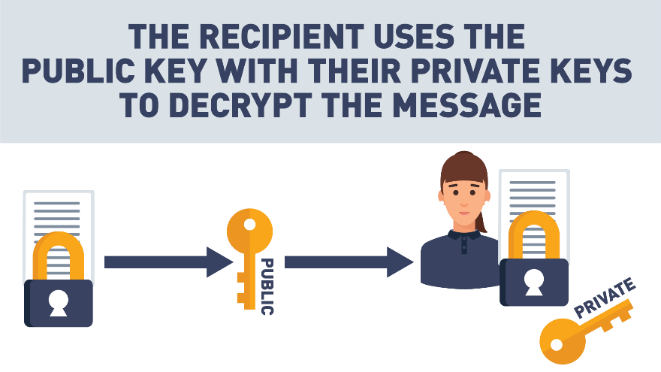
Step 1:



 Step 2:



 Step 3:



## Private/Public Key Cryptography

So, I want to talk a little bit about cryptography and some cryptography principles.

The first is easy and hard problems, and this is really what public - private key cryptography is based off of.

There are some problems that are very easy one way, but difficult another way.

And a great example of this is multiplication.

So, if I give you a pen and paper and a lot of time, I can ask you to multiply any two numbers,

and you can probably come up with an answer fairly easily.

What's a lot more difficult is if I give you the product of two numbers and ask you to factor that and find out all the numbers you can multiply together to get that result, that's a much harder problem.

This is how public - private key cryptography works.

We use an easy mathematical problem to generate a message, but we make it very, very hard for that message to be decrypted by someone who shouldn't decrypt it,

and public and private keys play a big role in blockchain.

I want to show you right now.

I've got two different wallets. I've got a wallet that you're probably used to, that you keep money in,

and then I've got a hardware wallet, a digital wallet that you'd use to trade something like bitcoin or ether.

Now, what you're used to, in your regular wallet, is putting money in here and you keep your money in here, and if you lose this, whoever finds it gets your money and they get to spend it.

In blockchain, with cryptocurrencies, it's a little bit different.

The only thing that ever gets kept in here is your private key.

And a private key - public key work together so that I can use a private key to sign, digitally sign, any message.

You can then take the public key, which anyone can know, and use that to verify that my specific private key was the one that actually signed that message.

And so, you can know without a doubt that I'm the one who created that message,

that no one was pretending to be me sending this message.

And this is how it works trading digital currencies back and forth.

If I decide I'm going to pay you three bitcoins and I'm gonna do it out of this hardware wallet or any software wallet, any kind of digital currency,

the only thing that's kept on this device is my private key and it never leaves,

and so, that means if I want to send you some money, all I'm doing is accessing the private key on this device to sign a message that gets transmitted to the blockchain,

that says I'm paying you some bitcoin or some ether or some litecoin,

and my public key is used to verify that that message actually did come from the wallet that says it did.

And this is how we trade cryptocurrencies back and forth,

and that's what makes the concept of a digital wallet a little bit different.

If I happen to lose this, I haven't lost the money that's in it, because there's no money ever in it.

All that money is is just a record on the blockchain, and, as long as I can recover that private key somehow, I can just get a new device,

start using my private key again and keep right on trading.

So, that's how public and private keys fit into blockchain; a very key component just to understand that they're used to sign and verify any transactions that you make.

## Zero-Knowledge Proof (ZKP)

Zero-knowledge proofs authenticate parties without the need to widely transmit private information online. A zero-knowledge proof (ZKP) is a cryptographic method that allows a party (the prover) to prove to another party (the verifier) that a given statement is true, without conveying any additional information.

Let's review an example. Let's say there are two toy cars, identical in shape and size, except, one is red and one is blue. Jerry, who is color-blind, holds the toy cars behind his back. Jerry then shows one of the cars to Sam. Jerry then hides that car behind his back and shows Sam the other car. Sam can consistently detect the switch because the cars are different colors, but he never has to reveal the color of the cars to Jerry in order to prove the secret.

## Zero-Knowledge Proof (Cave Example)

In this section, we're going to talk about a cybersecurity idea in blockchain known as "Zero-Knowledge Proofs".

Zero-Knowledge Proofs are exactly what the name implies - they allow me to prove to another party that I have a certain piece of knowledge without revealing what that knowledge is.

Zero-Knowledge Proofs are often referred to as zk-SNARKs, which stand for "Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge", or zk-SNARKs for short,

or, if you want to be simple, we can just call them Zero-Knowledge Proofs.

A great example of a Zero-Knowledge Proof is what we call the cave door analogy.

Pretend that there's a circular cave, with only one entrance or exit and at the back of this circular cave there's a door which can be unlocked using a secret code entered onto a keypad.

If I want to prove to you that I know the unlock code without revealing that unlock code to you,

all I need to show is that I can walk into one end of the cave, open the door, and come out the other end.

If I've successfully demonstrated that, then you know without a doubt I've been able to unlock that door,

but yet I haven't revealed that unlock code to you.

This simple cave door analogy is a Zero-Knowledge proof, and if you can remember this cave door analogy,

then you'll understand why Zero-Knowledge Proofs are so important in blockchain.

For example, a user may make a request to send another user some money.

The blockchain naturally wants to make sure, before it commits this transaction,

that the user sending the money has enough to send.

However, the blockchain doesn't really need to know or care who is spending the money, or how much total money they have.

Being able to answer a question of "Does a user have enough money to send to another user" without knowing who the user is, or exactly how much they have,

is one of the primary use cases for Zero-Knowledge Proofs in blockchain.

So, when you hear about zk-SNARKs, don't get intimidated.

It's not a complex idea. Just think of Zero-Knowledge Proofs and the cave door,

think about how important it is in blockchain to be able to prove a claim without actually revealing the information behind that claim,

and you'll understand Zero-Knowledge Proofs.

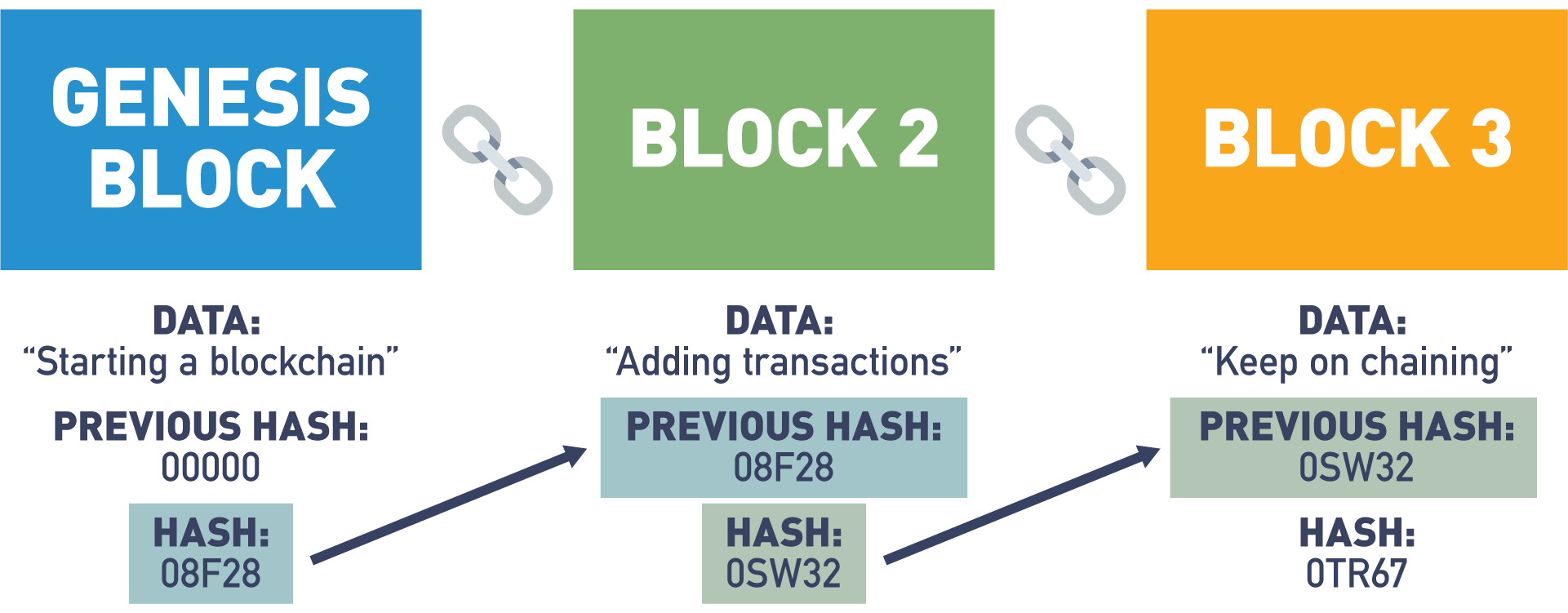
## Hash Functions in Blockchain

When using blockchain, the need to trust a central authority to verify the accuracy of data is removed and replaced by trust in a cryptographic hashing function. With data integrity guaranteed by algorithms, trust becomes part of the system.

Blockchain provides users with data integrity in a trustless environment. This is accomplished using cryptography in a way that moves the burden of trust from data processors to cryptographic algorithms.

Hash functions are featured heavily in blockchain. A hash function is a mathematical equation with four important properties:

1. Hash functions can take anything as input and create an output with a fixed size. This makes it possible to condense anything into a piece of data of a fixed size and is how messages are condensed for digital signatures.
2. It’s easy to calculate a hash, but hard to determine a hash input from the output. The best option is to keep trying inputs until one produces the desired output.
3. Inputs that differ by a single bit produce hashes that differ by half of their bits on average. This prevents someone from finding a desired hash input using a “hill climbing”.
4. It is infeasible to find two inputs that produce the same output when hashed. Since a hash can take any input and produce a fixed output, it makes sense that multiple different inputs will create the same output. A good hash function will make it so that you have to try a large number of inputs before finding two that produce the same output.



## Cryptographic Hashing Demo

In this section, we're going to talk about cryptographic hashing.

A cryptographic hash is a special kind of cryptographic function known as a one-way function,

whereas most cryptographic functions are two-way, meaning that data cannot only be encrypted, but can be decrypted at the other end.

Hashes, or cryptographic hashes, are very unique in that once data is encoded, it can never be decoded.

Let's look at a small example of this.

If we take a simple sentence, "Let's eat, grandma!", this input sentence will generate a unique 32-character hash output.

This hash output can be thought of as an identifier for this input data.

In other words, I can input this data into a hash function to generate this output, and I can share this output with anybody, anywhere,

safe in the knowledge that it is mathematically impossible for anyone to reverse-engineer the original input from this hash output.

In fact, the only possible way that anybody could ever determine what this input was,

was by randomly inputting data into a hash function until they were able to recreate the same output.

Another unique function of a cryptographic hash is that any small change to a dataset, no matter how small,

will result in an entirely different hash.

If I remove the comma from this sentence, not only do I entirely change the meaning of the sentence,

I also generate a new 32-character hash output.

This 32 character hash output remains a fixed length no matter how big the input dataset is.

To demonstrate this, let's take the entire first book of Tolstoy's War and Peace, and paste it into our hash function.

You'll see that the most time-consuming part of this is not generating the hash,

but simply moving all the data from the clipboard into the input text box.

I've generated a unique 32-character output or identifier for this input text.

You'll notice that this is the same length hash that I got when I had a short one sentence input,

and if I change anything in this extremely large dataset, even one character, I will generate an entirely new hash output.

Let's take a look at this by removing the period at the end of this sentence, and replacing it with a question mark.

You'll see that by doing so, I've generated a completely different and unique 32-character hash output.

Hash functions not only play a critical role in blockchain security by enabling blocks to be linked together in creating immutability between data on the blocks,

they can also be used as a great way to identify and verify large sources of data.

Let's look at some other use cases.

If I want to build a system which authenticates users, but I don't want to store their password directly in a database where it might be stolen or corrupted,

a hash function can be a great solution.

Let's say I have a very simple password, password123, but I don't want to store that password inside my authentication database.

I can simply ask the user to type in their password and store the hash output from that password.

Any subsequent time a user tries to log on, I don't need to know their password. I just need to verify that they're able to recreate this hash output.

If they can't do that, because they have the wrong password, then I can simply choose not to authenticate them into my system.

Cryptographic hashes can also be used to validate source code or real large datasets in the real world.

Let's look at self-driving cars for an example.

Let's say we've developed a self-driving car with the intelligence to hash the source code every time it starts up.

My self-driving car, every time I start the car, can take its source code, run it through a hash and generate a unique identifier.

It can then share this identifier or 32-character hash output with every other self-driving car it has contact with,

or even validate this against a database or some other centralized data store, or perhaps an immutable record on the blockchain.

This gives me the confidence that if the source code for my self-driving car ever gets hacked and makes it unsafe to drive,

I'll know it right away. I'll try to start my self-driving car, which will immediately hash its source code,

share that hash output with other self-driving cars or validate it against an authoritative source,

and instantly we can see I've got an entirely different hash output letting me know before I've gone anywhere

that my self-driving car is not safe, and putting that self-driving car into a limp or shutdown mode,

where it simply refuses to go anywhere until the compromised software has been updated and replaced with the right version.

Cryptographic hashes give us a way to link all blocks on the blockchain together in an immutable fashion,

such that if any one particular block or the data on any block is changed or altered to any degree, no matter how small,

we will know it instantly and we will break the link between that block and every subsequent block.

If you haven't watched the module in this course that covers the in-depth walkthrough of Proof of Work consensus,

you might want to check it out now.

We actually show how cryptographic hashing is used to link together all blocks on the blockchain,

creating a data store that is immutable, secure, and extremely trustworthy.

## Lab 1: Hashing

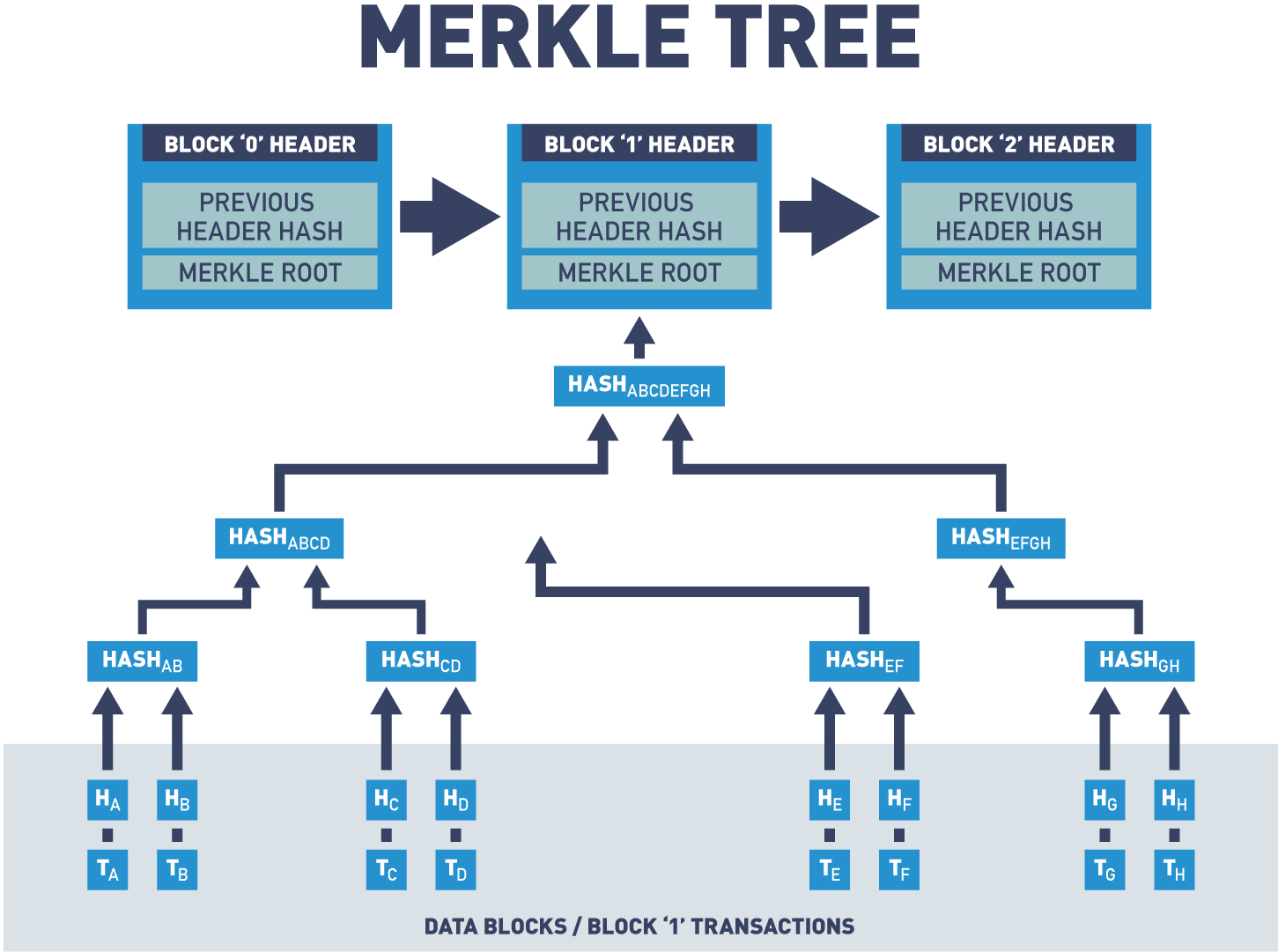
Next, let's engage with an interactive lab. This lab is an actual hands-on demonstration of taking data and creating a hash output. Enjoy!

[Start Lab](https://blockchaintrainingalliance.com/pages/lab-hash)

## Merkle Tree

A special type of data storage structure based on hash functions is called a Merkle tree:

* + - It is structured as a binary tree; the leaves contain the values to be stored and each internal node is the hash of its two children.
    - It provides efficient lookups and protection against forgery since verifying a transaction is included in the tree. Can be accomplished by sending only the transaction, the hash contained in each node between the transaction leaf node and the root, and the hash values used to create each hash sent.
    - Looking up a transaction in a Merkle tree with three levels includes sending two transactions (the desired one and the other child of its parent) and three hashes (the transaction’s parent, the root, and the root’s other child).



Merkle trees are a data structure that allows authenticated storage with efficient data retrieval.

## Merkle Tree, Validation of Data

Blockchains use Merkle trees for fast and efficient validation of data.

Merkle trees summarize the entire set of data in a block by creating a root hash of that data.

A root hash is created by repeatedly hashing pairs of child nodes until only one node is left.

Let's take a look at a diagram of a Merkle tree.

This diagram consists of four transactions represented as transaction "A", transaction "B", "C" and "D".

The transaction data is then hashed and the resulting hashes are stored in leaf nodes represented as H of "A", "B", "C" and "D".

Nodes are repeatedly hashed in pairs and the resulting hashes are stored in child nodes until only one node remains: the Merkle root or root hash.

The reason you wouldn't want to hash the entire set of data as a string is because it's not efficient when validating transactions.

In this example, if Alice needed to show Bob that this transaction was valid, let's say transaction "C", Alice would need to send the entire list of data to validate the block.

But when using a Merkle tree, Alice only needs to send the nodes that contain the transaction that she is looking to validate.

Let's say Alice wants to validate transaction "F".

Alice would only need to send the data of transaction "F" and four hash values to Bob.

Bob would then calculate the hash value based on the given information.

And if the root matches the block, the transaction can be assumed to be valid.

Merkle trees allow for validation of a specific transaction without requiring the entire set of data.

## Ethereum vs. Hyperledger Fabric

The use of public keys for identity management is a logical choice since knowledge of a public key is necessary for verification of digital signatures. Both Ethereum and Hyperledger Fabric use digital signatures on transactions and blocks to verify the identity of the creator and that the signed data has not been modified since signing. Public key cryptography is used in the blockchain as a method for managing users’ identities without revealing real world identities.

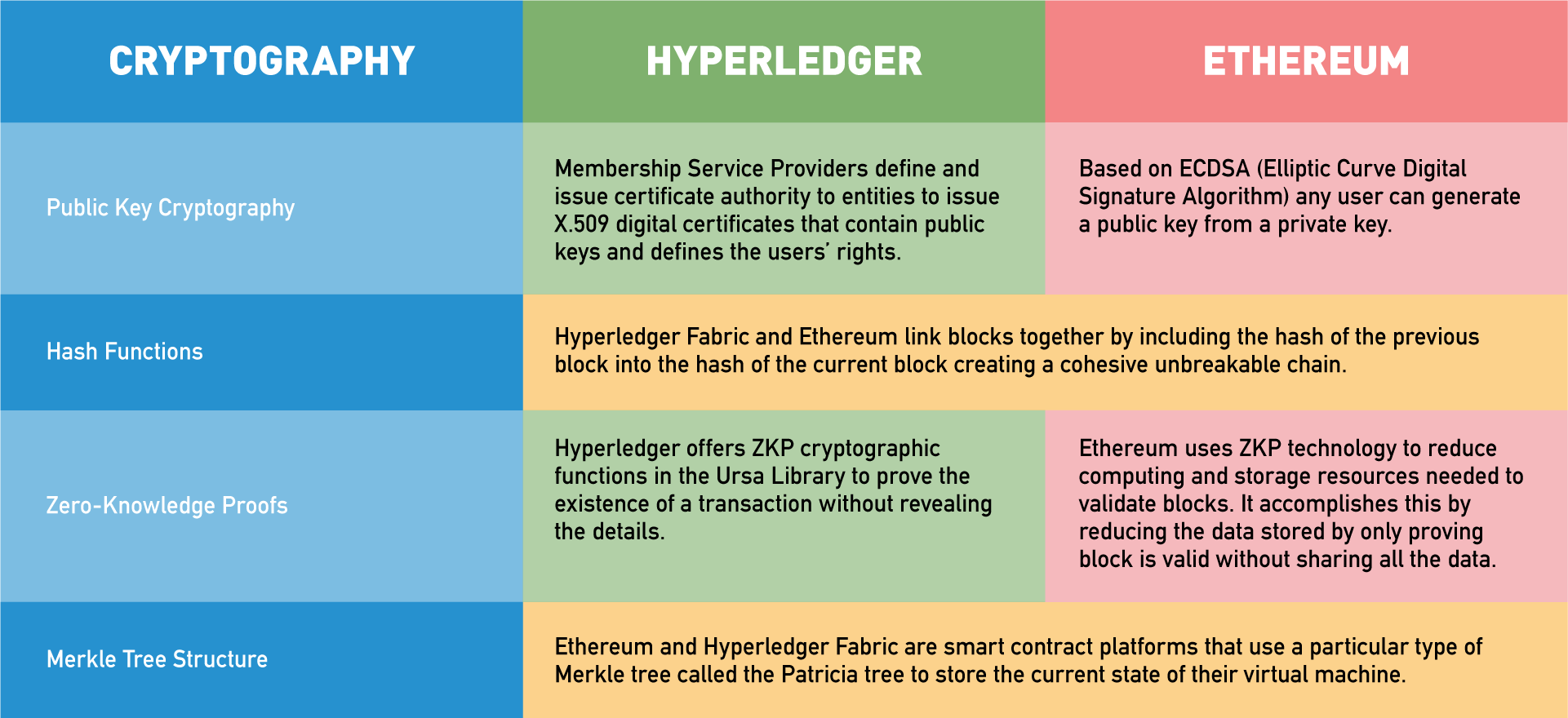
In Ethereum, users are identified by an address that is directly related to the user's public key. This provides identity verification while preserving anonymity.

In Hyperledger Fabric, users are identified via [X.509 certificates](https://www.ssl.com/faqs/what-is-an-x-509-certificate/). These certificates provide several pieces of information about the user, but one of these is also the user's public key.

Hash functions are at the core of all blockchain technology. One of the primary uses for hash functions is chaining blocks together. In both Ethereum and Hyperledger Fabric, blocks include the hash of the previous block to tie the blockchain into a cohesive whole.

Zero-knowledge proofs are to increase the privacy of users. Currently, Ethereum is working on a layer 2 solution, where a second layer blockchain would store large amounts of data and only prove the batch of information is valid to layer 1 (mainnet). Hyperledger Fabric has a pluggable cryptographic library that supports ZKP techniques for enhanced privacy measures.

Merkle trees are a data structure that allows authenticated storage with efficient data retrieval. Both Ethereum and Hyperledger Fabric are smart contract platforms that use a particular type of Merkle tree called the Patricia tree to store the current state of their virtual machine.



## Disintermediation - Trust Through Transparency

Now, that we have an understanding of how collaboration and cryptography work together to create a layer of trust,

let's discuss how blockchain has a powerful disintermediation effect.

Nodes on the network are allowed to place data directly onto the database that is shared.

This eliminates the need for an intermediary to do such task.

As we discussed, developers can create a distributed ledger on a blockchain,

and use cryptography to give people secure storage space on that ledger.

This creates a very different world than what we have come to know today,

because for the first time in the digital world, people are allowed to own their own data.

You may be asking yourself: "How is this different from the current technology?".

Today, organizations use our data, sell our data, store our data, and exchange our data.

Data is an extremely powerful asset and to own your own data is a vast change to the current system.

In a world powered by blockchain, having ownership of your database is truly empowering.

You can now share the information that you choose with any organization that you desire.

The power of blockchain and owning your own data is the ability to transact peer-to-peer.

With the current system of intermediaries, when you go out to eat at a restaurant,

and you pay with your credit or debit card, you're not paying the restaurant directly.

Instead, a database record at your bank is being debited and the database record at their bank is being credited.

In this example, the value and data in the database belong to the intermediary, and not the individual.

You are then dependent upon them to secure and validate your transaction, which shifts the power to the intermediary.

But in a blockchain world, the individual has their own ledger record, and the secure key that allows them to access it.

You no longer need an intermediary.

A transaction is simply debited from your record and credited directly to somebody else's record.

Blockchain allows you to have the power to control more than financial transactions.

Later, we'll take a deeper dive into how the healthcare industry may utilize blockchain to allow people to own their own health records.

And with blockchain, you can determine who can access those records.

We talked about how cryptography and collaboration create trust.

Now, let's examine how peer-to-peer transactions with blockchain create transparency which can lead to even another layer of trust.

This layer of trust can aid to curb corruption and even create cost savings.

Currently, the data and records owned at centralized organizations are stored and hidden within a database.

These databases for the most part are inaccessible and do not communicate in an effective manner with other systems.

This also makes it easier for a corrupt individual or persons within the centralized organization to manipulate these records.

Because of the possibility for manipulation and foul play, there are many flawed and expensive safeguards put into place.

These safeguards are done through regulations and legal requirements creating expensive overhead and delays.

Besides the mounting cost to safeguard these internally, there are also many necessary safeguards from external forces,

because they are individual points of critical data storage, bad actors have to identify only one point of failure to access.

The combination of one central point of storage and the storage of valuable data make these centralized databases a very enticing target for malicious persons.

With blockchain, these malicious actors would have to identify and target numerous points of failure until they control more than 51% or what's known as consensus.

In most blockchains, it could be up to hundreds of thousands of points of weakness that would have to be exposed to be corrupted.

It's nearly an insurmountable task.

Not only must a centralized organization worry about bad actors, but they must do so, while still allowing access and synchronizing with other centralized organizations.

Having to constantly update and synchronize data across many centralized databases is not efficient.

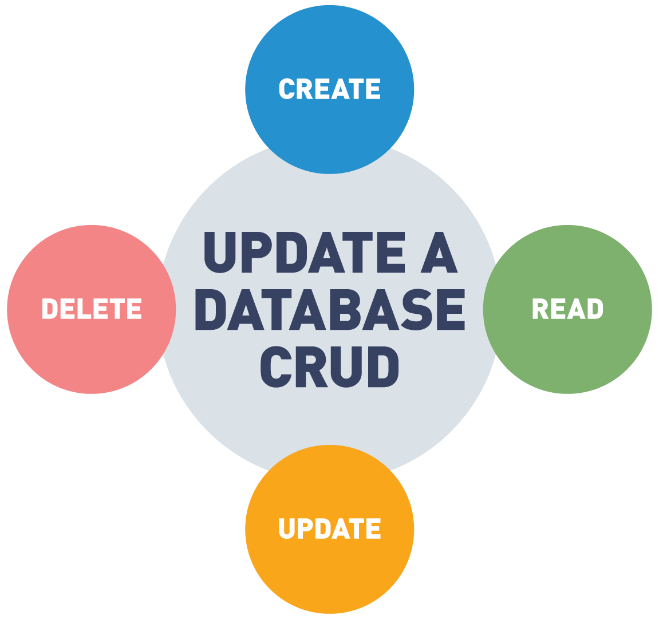
The cost is high and the vulnerability to bad actors is amplified.

All the more reason blockchain solutions are so exciting.

Blockchain offers efficiency. It also creates trust and financial benefit through transparency, security, immutability, and accessibility through ownership.

## Traditional Database Updates

A database is a collection of interrelated data organized so that individual records or groups of records can be retrieved to satisfy various criteria.



Traditional databases use the CRUD method which has four main operations:

* Create - add a new value to the database.
* Read - read a value from the database.
* Update - change an existing value in the database.
* Delete - remove the value from the database.

## Transparency of Traditional Databases

Traditional databases do not retain historical information:

* Only the most recent versions of each value are visible.
* Deleted values are not visible in the database.
* This limits the transparency of data contained in the database.
* Values can be modified or deleted after creation.

The CRUD update model of databases allows data to be changed or removed from the database. This means that the visible data in a database is not an accurate historical record of the database. The existence of an update operation means that each value in the database is only the most recent version of that value and could have had different values in the past. The delete operation means that values can be removed from the database.

This limits the transparency of data in the database since values can be modified or deleted after creation.

## Blockchain Updates

The blockchain is designed to be a data structure that only allows appending:

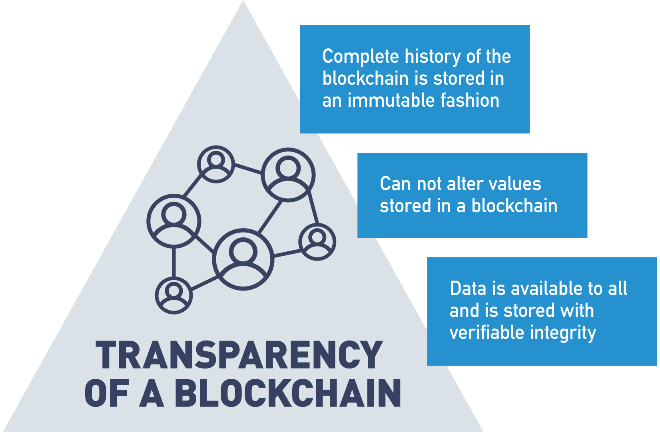
* The past history of the blockchain is visible and immutable.
* Updates to the blockchain can be performed by including them in new blocks added to the blockchain.

The blockchain is designed as a data structure where each block in the chain locks in the value of the previous block and so on, back to the first or *genesis* block. This means that the blockchain is an append-only data structure without support for modification or deletion.

The entire history of the blockchain is publicly visible and stored in a distributed and decentralized fashion. Values in the blockchain can be “updated” by appending a new version of that value in a later block, but the complete history of the value is preserved.

## Transparency of a Blockchain

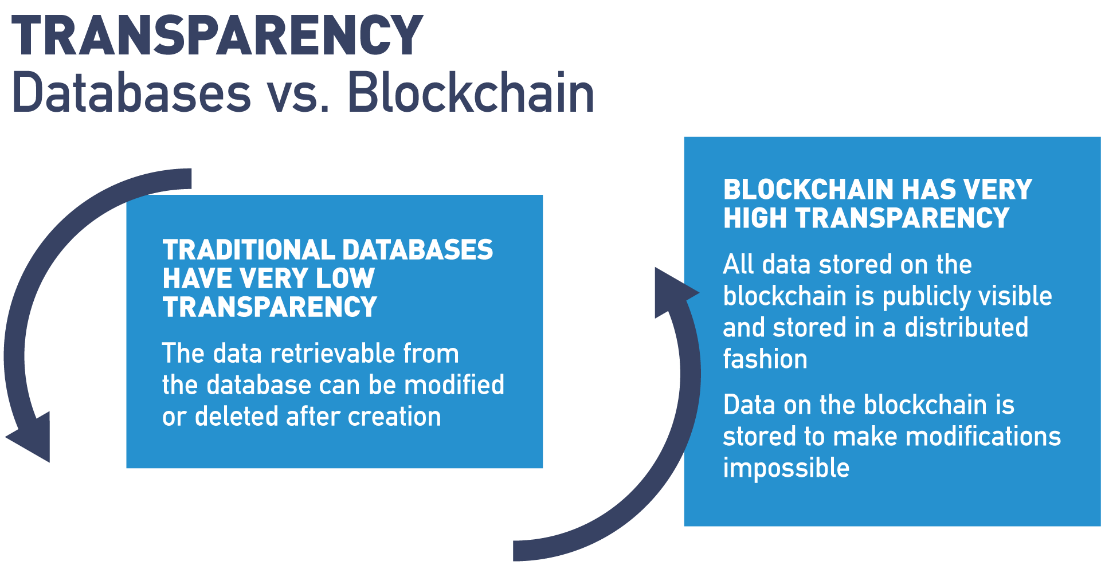
The blockchain is designed so that its entire history is visible and unchangeable. Transactions in the blockchain cannot be modified after creation, and their complete history is publicly visible. This means that the blockchain is a completely transparent data structure with the useful property that the integrity of the blockchain is easily verifiable by any user.



## Transparency: Databases vs. Blockchain

Traditional databases and the blockchain were created for different purposes and have different levels of transparency. Traditional databases have low transparency since values can be modified or deleted; however, this changeability allows them to store data in an efficient manner, with only the most relevant versions of each value retained in storage.

The blockchain is publicly visible and immutable, meaning that it has very high transparency. Its append-only structure and decentralized storage sacrifice storage efficiency for trustworthiness of the stored data.



## Transparency

One of the big benefits we get in any blockchain solution, public or private, is the idea of transparency.

In a public blockchain solution like Bitcoin or Ethereum, because they're anonymous and we have no concept of identity, we have no way of treating users differently.

And so all data on a public blockchain is transparent and visible to all other participants.

And this is really important. It adds a lot of value, when we're talking about exchanging currency or tokens of monetary value,

because as long as we can protect the anonymity of the participants, it gives us a great way for anybody to validate the details of any transaction on a ledger

that is immutable, permanent, can't be changed and offers a high degree of trust.

This also has a lot of other really interesting potential use cases that we're starting to explore with public blockchain technology,

voting and voter registration, tracking election results.

These kind of things are really, really great use cases for having that level of public transparency.

On public blockchains, there's another kind of transparency, which adds a lot of value as well, and that is full transparency into the smart contracts that execute on that blockchain.

These, oftentimes, are smart contracts which have the potential of working with other people's money,

sometimes large amounts of other people's money, and if you're going to entrust your hard-earned money into someone else's smart contract,

it's really nice to have the ability to be able to go and review the source code of that smart contract,

either yourself or by hiring a developer to audit that smart contract, to make sure it's written fair,

that there are no security vulnerabilities, and that it does what it claims to do.

On the private blockchain side, what we have [is] permissioning.

And we can control how open or closed our data is.

We still have a lot of benefits of being able to share data in shared infrastructure that private blockchain creates.

There are a lot of business networks and value chain relationships where the participants would love to be able to share more data

and more information with consumers about the business processes and the steps that go into delivering the goods and services we consume.

What has traditionally prevented solutions like this is the lack of any kind of shared infrastructure.

While technically conventional technology may be able to address these concerns,

it is oftentimes human beings not being able to come to consensus on questions of who owns the platform, who builds the platform, who maintains the platform.

And, if a participant leaves the ecosystem, do they take all or part of the platform with them.

That have prevented us from using conventional technology in these respects.

Using private blockchain solutions like Hyperledger, can help alleviate many of these concerns and create a truly shared technical infrastructure between organizations.

This allows participants in a supply chain or value chain to track data and to share that data about how goods and services are produced,

created, consumed, and distributed into the open market, that we haven't really been able to do practically before.

It's estimated that over 1/3 of the food we produce goes to waste, because we don't have this kind of clarity into our supply chain with food or into the supply chain with other types of products.

Think about if a meat gets recalled because it might be contaminated, how much perfectly good meat gets thrown away just to err on the side of caution.

Where if we just had greater resolution and visibility into where that product originated from and how it went all the way from production to our plate.

Then, we could issue more accurate recalls and reduce the amount of food waste.

Consumers might value your product more, your offering more, if they're able to see where it comes from.

If you're manufacturing and distributing clothing, and you can demonstrate to consumers that all of that clothing was

sourced from parts of the world with fair labor practices consumers are going to

have more trust and faith in the brand and ultimately willing to pay a higher price for it.

So, these kind of transparency solutions are one of the biggest benefits of blockchain and one of the areas being most explored,

because right now there are a lot of consumers who would simply like more visibility into the products and services they purchase.

And blockchain, both public and private, can be a great solution for doing that.

## Immutability in the Blockchain

Blockchain is designed to be an authoritative ledger of the history of the network.



This history may include financial transactions and business agreements where modifications to the ledger may have wide-reaching business impacts. Blockchain is based on an untrusted network, so trust that the blockchain has not been modified needs to arise from the structure of the blockchain itself, rather than from trust in the organization storing a certain copy.

## Immutability

So, one of the big benefits of blockchain is what's known as immutability.

The idea that once a record gets committed to the blockchain or to the ledger, it's there permanently.

In blockchain, it's a little bit different than a database, and we'll talk about this in another module, in a lot more depth,

but in a traditional database we're used to being able to edit records and delete records, and that's not the case in a blockchain.

In a blockchain, there are only two possible operations: we can create a new record, and we can go back and we can read existing records.

And this contributes to this idea of immutability - once data is written to the blockchain it can't be edited, it can't be deleted.

All you can do is add another record to the blockchain indicating that something about that data has changed.

This immutability is what makes blockchain such a secure solution, and it's what makes it a trustless solution.

Meaning, I don't have to trust anybody that I'm interacting with and trading with, as long as we both trust in the blockchain and the data on the blockchain.

Then, we can transact back and forth seamlessly.

The way immutability works is blockchain is, by design, a very, very inefficient solution.

We are asking a large number of nodes, all the nodes on a network, to perform exactly the same work as every other node.

So, essentially, we are duplicating work over and over and over again.

The reason we do this, is because the only way to potentially change data on a blockchain,

is by changing the record on that blockchain for 51% or more of the nodes, or the miners, or the participants trying to come to consensus on the blockchain.

And when you get to very, very large-scale networks, Ethereum with its 16,000 plus nodes, Bitcoin with over 30,000 nodes, this becomes a very, very, very tall order.

Trying to hack or change a record on over 15,000 of those Bitcoin nodes would be quite a bit of work.

So, it's not impossible to change the record on a blockchain, but it is impossible from a practical standpoint.

Now, one thing that should be noted when we talk about immutability is: all that blockchain gives us is an accurate recording of what was posted.

It is still up to whoever is posting that information on the blockchain to make sure it's accurate and true.

In other words, blockchain doesn't do anything to validate that the information on it is correct.

All it validates is that that information hasn't been changed since it was originally written.

Another way to think about that is, I could post a record on blockchain that says: "The sky is orange".

That doesn't make the sky orange.

We all know the sky is blue. All that does is that means I can't ever come back later and claim I didn't say that or I said it was blue, and someone changed the record.

That immutability that blockchain gives us is that that data was captured as it was originally written and hasn't been changed.

Another part of this immutability is cryptographic hashing, and the way that we chain all these blocks together.

If you haven't watched the module on Proof of Work, I would suggest you go watch that now, to see a really in-depth overview, a hands-on demo of how blocks are actually linked together in such a way.

And, if I ever went back and tried to change any information or prior transaction on the blockchain, I would break that chaining in a very

obvious way that anyone would immediately spot if they came and querried that data off the blockchain.

So, when you hear about immutability, just understand that all we're talking about is the virtual impossibility of anyone ever being able to come along and change any data on the blockchain.

And also, along with that, understand that we only have two possible operations on blockchain: we can only read data and we can only add new data.

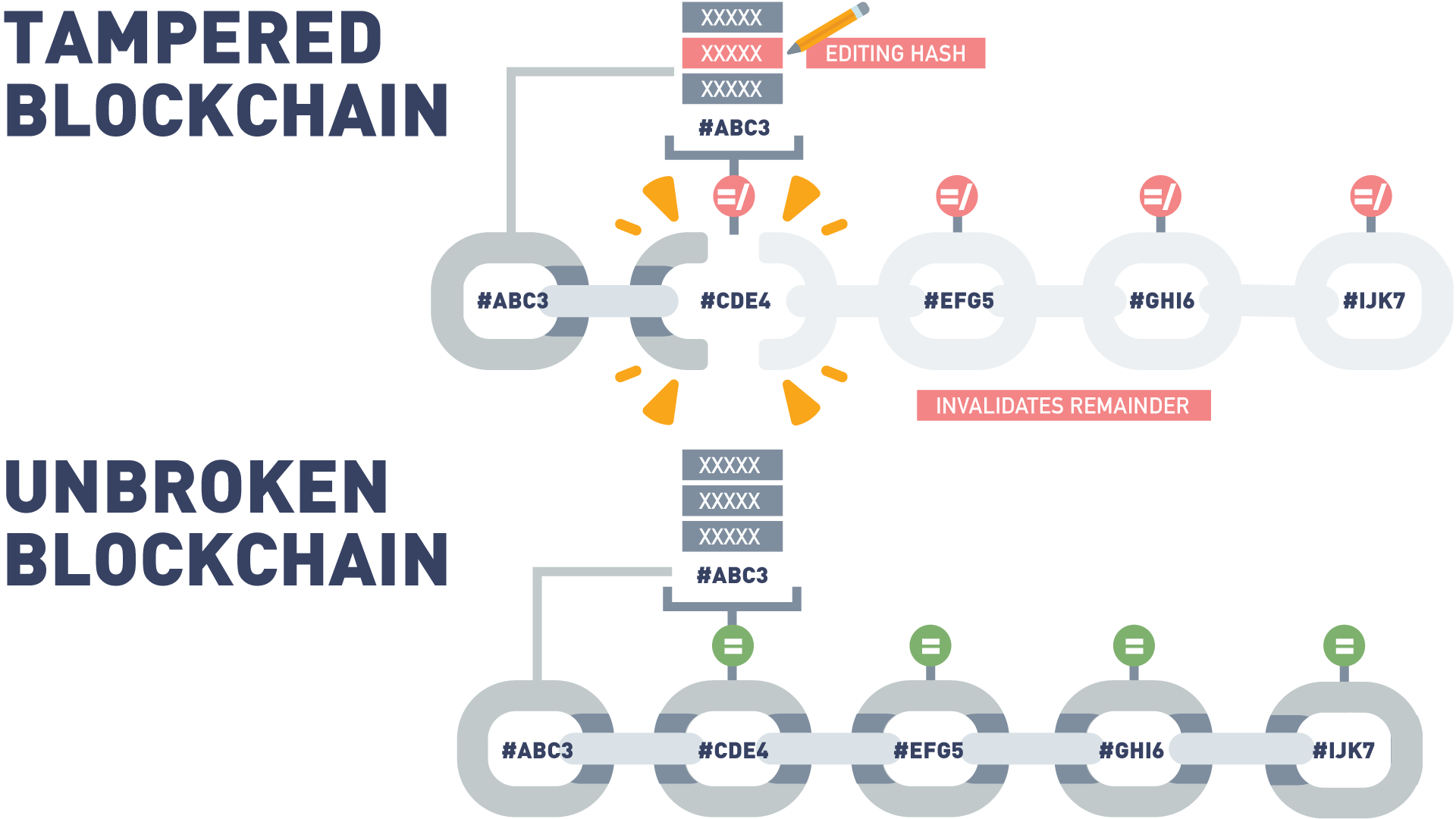
There's no concept of an edit or delete, and this is what we talk about when we talk about immutability on the blockchain.

## Immutability in the Blockchain Is Essential

The blockchain needs to be immutable. If someone can change the blockchain after the fact, then it is no longer a trusted historical ledger. The blockchain is designed so that immutability is cumulative; each piece is linked to every other piece, creating a cohesive whole that is more difficult for an attacker to modify.

* At the bottom level, transactions are digitally signed by their creators. An attacker can’t forge a transaction unless they steal a private key.
* A block structure is predefined. Attackers can’t modify it to suit their purposes.
* The chain part of the blockchain is achieved using hash functions. Each block includes the hash of the previous block, creating a clear link between each block in the blockchain.
* Each block is digitally signed by its creator. The creator is selected through the blockchain’s consensus protocol, making it difficult for an attacker to be a legitimate creator.

All four of these features help to make the blockchain resistant against changes occurring after the fact.



## Why Is the Blockchain Immutable?

Each transaction cannot be forged or modified because it is mathematically infeasible to forge a digital signature. The structure of blocks is publicly defined, and invalid blocks will be publicly rejected.

Each block “locks in” the value of previous blocks by including their hash. Attackers cannot find another block that will produce the same hash.

A block cannot be forged or modified, because it is digitally signed by the creator. The creator of a block is either publicly known (Proof of Stake) or difficult to become (Proof of Work), making masquerading as the real creator difficult or impossible.

Now, let’s take a moment to discuss how each of the features mentioned contribute to the immutability of the blockchain.

At the bottom level, each transaction is digitally signed. This means two things about transactions:

* Existing transactions can’t be changed after the fact, because the signature will no longer match.
* Fake transactions can’t be created since an attacker can’t create a valid digital signature for a transaction between other parties.

Both of these contribute to the immutability of the blockchain since they limit the range of transactions that an attacker has to work with if he wants to create a fake but valid blockchain.

Next, the block structure is publicly defined in the protocol. This limits the types of modifications that an attacker can make to a block when trying to modify the blockchain.

Third, each block contains the hash of the previous block. This is what ties the blocks of the chain together. Remember from earlier that one of the properties of a hash function is that it is extremely difficult to find two inputs to a hash function that create the same output. Since a block contains the hash of the previous block, it’s difficult to find a different version of the ledger’s history that matches the most recent block, as that would require finding two different versions of the previous block that have the same hash.

Finally, each block is digitally signed by its creator. Since the creator of a block is selected via a consensus algorithm, it’s difficult for an attacker to become the legitimate creator of a given block. If an attacker is not the legitimate creator of a block, it’s impossible for them to create a digital signature that others would accept.

## Immutability Mechanisms

Now, let’s look at how different blockchains implement immutability.

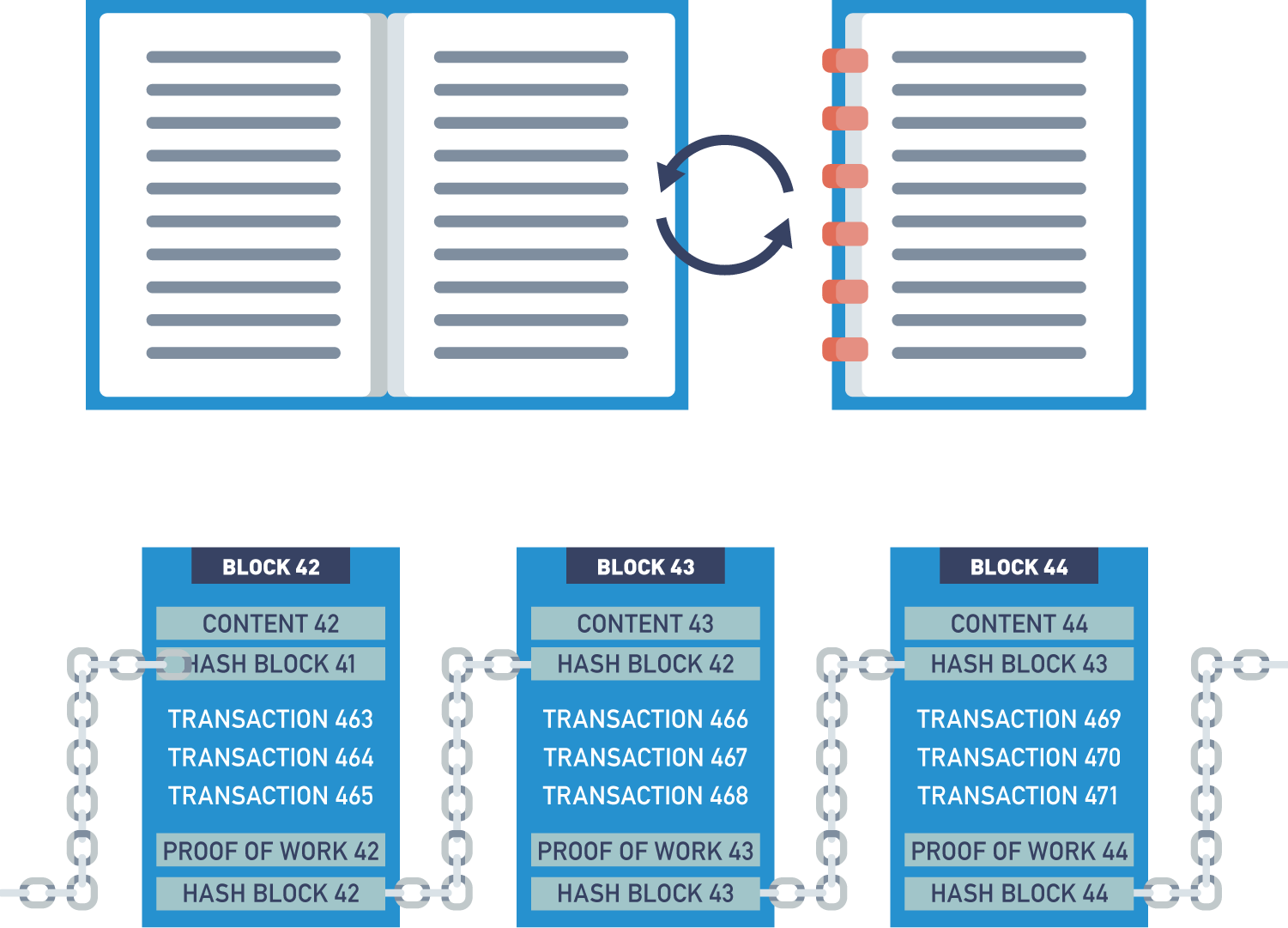
In Ethereum and Hyperledger, the immutability mechanism is the one that we’ve described previously. Each transaction and block is digitally signed and are linked using cryptographic hashes.

[Corda](https://www.corda.net/), a multi-party application development platform, relies on its notary service for immutability. Each Corda network has one or more notary services that verify transactions. Each transaction is considered separately and, if approved, is signed by the notary service. Transactions signed by a notary are finalized and cannot be modified after the fact.

## Hashing and Chaining

It is important to understand the value of locking in the previous block by including its hash in the next block. As explained in the article [*“A Gentle Introduction to Immutability of Blockchains”*](https://bitsonblocks.net/2016/02/29/a-gentle-introduction-to-immutability-of-blockchains/) by Antony Lewis,

* Creating a ledger of transactions with blocks that refer to previous blocks is a much better idea than numbering the blocks sequentially.
* In a book with pages, 1, 2, 3, etc., it would be easy to tear out page 25 and replace it with another page.
* The integrity of the book has been manipulated and altered. However, there is nothing about the new page number that ties it (chains it) to the content of the previous page.
* Instead, in a blockchain, blocks are referenced by their hash and each block explicitly specifies which block (hash) it is building on.



## Hashing and Chaining with Proof of Work

We will cover the Proof of Work in a later module. However, this demo uses the Proof of Work to show how blocks are chained together and how tampering with information in a past block destroys the integrity of the proceeding blocks.

In this video, we're going to talk about Proof of Work consensus, and how Proof of Work consensus is used to create an immutable chain of block data on the blockchain.

Let's take a small example. Let's say we have some block data and, in the interest of simplicity, we will keep this very, very short.

But understand that in real life a block will most likely contain many more than just three transactions.

But on this block, we have a full block consisting of three transactions.

When it's time to validate a full block, all nodes on the network will start to guess a nonce value.

What we're looking for is a nonce value such that when we take our input transaction data and we combine it with a nonce value, we get a hash output that matches the difficulty.

In this case, the difficulty is a hash output that begins with four leading zeros.

Every node on the network will guess a different nonce randomly until they happen to find one that matches the difficulty.

So, our node may start off by guessing "123", and we see that that does not produce a hash output that matches the difficulty of four leading zeros.

Let's try again. Let's try "973321". Once again, that does not give us a hash output that matches the desired difficulty.

[So], rather than guess over and over and over, let's go ahead and ask a computer to do our work for us.

The computer is now guessing different nonce values randomly until it happens to guess "22294", which you see produces a hash that begins with four leading zeros.

Because this hash is the output of input to a hash function consisting not only of the transaction data, but also of the nonce, changing either one will result in a new hash output which does not match the difficulty.

If I change "22294" to "22295", you'll see I get a hash output which no longer meets the difficulty criteria.

However, if I change any of my transaction data, I'm also altering the input to the hash function, and thus generating an output which is completely different from the desired outcome.

In this way, nodes or miners on a network can validate they have the same transaction data.

If we all start randomly guessing nonce values for this set of data and my node guesses "22294" first,

all it has to do is simply share this nonce value with every other node on the network.

Each node can then try this value "22294" on their copy of the transaction data.

If any of the data has been altered or changed in any way, no matter how insignificant, or the node in question simply did not record the data correctly, this nonce value will not work for them.

For example, if we change "4.2 bitcoin" to "4.9 bitcoin", you'll see that the proposed nonce does not work.

In this way, all nodes can come to consensus or agreement on what the right set of transaction data should be.

When a node thinks it's guessed the right nonce, it simply shares that value with the rest of the network.

If the majority of the network, 51% or more, agree with this nonce, because they have the same transaction data on their copy of the block,

then the miner who guessed this number first, will be rewarded and the network will come to consensus or agree that this is the correct version of the block data.

Any nodes which have different data for the current block will then discard the copy of the current block they have and replace it with a copy that the rest of the network has agreed upon.

This method is used to create blocks of data chained together.

So let's say, that we have some transaction data here on the blockchain.

We're gonna go ahead and mine our first block.

Now, you'll notice what we've added in this demo is the hash of the previous block, which is something we store in the header of every block on the blockchain.

Because this is the first block, I simply get a previous hash value of zero.

Combining zero with the current transaction data on the block and the proposed nonce gives me a hash that meets the output difficulty.

When it's time to validate the second block, the process repeats itself.

I simply take the hash of the previous block stored in the header of the current block, combine it with the transaction data in the current block, and try to guess the correct nonce.

In this case, it's "12314"

As discussed in our video on cryptographic hashing, changing any of the input data, no matter how slight, will result in an entirely different hash.

So, if I try and change "2.3" to "2.9", I no longer get an output which matches the current difficulty.

This new output is also reflected in the header of the next subsequent block.

If I have a series of blocks that have all been mined and linked together, then you can see that going and changing data on any block will break every subsequent block after it.

So, if I try to alter this record "David pays Charlie 2.3 bitcoin" and I want to change it to "5.3 bitcoin", I change the hash output of the current block.

This is reflected in the current block and also in the next block.

Now, the nonce proposed for the next block does not work when combined with that block's transaction data and with the hash of the previous block.

This means that if I want to alter data on the blockchain, not only do I need to remind the block that I changed the data on, I also need to remind every subsequent block in the blockchain.

This is no trivial task when you consider that major blockchain such as Bitcoin and Ethereum contain millions of blocks.

And this process must be repeated on tens of thousands of nodes worldwide in a very short amount of time.

This ability to link together blocks in an immutable fashion and to cause attackers to have to undertake so much work, repeated so many times is what gives blockchain its security and immutability.

I hope you've learned a little bit more about Proof of Work consensus and how blocks on the blockchain are linked together through watching this video.

## Lab 2: Blocks

Next, let's engage with an interactive lab. This lab will give you the chance to examine blocks on the blockchain. Enjoy!

[Start Lab](https://blockchaintrainingalliance.com/pages/lab-blocks)

## Chapter Summary

In this chapter, we discussed:

* Distributed ledger technology (DLT) and how it differs from previous ledger storage techniques.
* Cryptography and how it is used in several different ways in blockchain.
* Hashing functions, zero-knowledge proofs (ZKPs) and Merkle tree data structures and how they are used to secure the blockchain.
* Public-key cryptography and public/private key pairs to support privacy, authenticity and security.
* Ledger immutability and transparency result from combining these technologies for an agreed-upon, append-only distributed ledger system chained together by blocks.

# Chapter 3. Blockchain Functions

## Chapter Overview

In this chapter, we will examine some blockchain functions. First, we will discuss smart contracts, how blockchains have the ability to store and run computer code (applications) that can execute the terms of an agreement. These digital promises stored and executed on the blockchains remove intermediaries and ambiguity thereby reducing conflict.

We will discuss how blockchains secure their information and how these techniques differ from traditional database security. Next, we will discuss different types of blockchains (public, private and hybrid models) and design considerations that need to be made before blockchains can be developed. We will examine the steps in a basic blockchain transaction from start to finish.

Finally, we will discuss blockchain consensus models and why they are the backbone to the blockchain operations. We will compare Proof of Stake and Proof of Work models and be able to discuss the differences and environmental considerations. Also, we will discuss private blockchain structures and how permissioning mechanisms are the basis for their ConsenSys models.

## Learning Objectives

By the end of this chapter, you should be able to:

* Explain what smart contracts are and how they work.
* Discuss the benefits of using smart contracts.
* Compare blockchain security vs. standard security.
* Indicate the differences between private and public blockchains.
* Understand when to use a public vs. a private blockchain.
* Analyze the flow of a transaction in blockchain.
* Discuss consensus mechanism used in blockchain.
* Explore different methods of achieving consensus (Proof of Work, Proof of Stake, etc.).

## What Are Smart Contracts?

When we make a transaction on the blockchain, understand that there are only three types of transactions we can make.

It's what I refer to as the onion model of blockchain.

And we're gonna start off with a very outer layer of that onion.

The thing that most people are familiar with when they get into blockchain.

The thing we do all the time when we trade cryptocurrency back and forth.

That is two or more parties coming together and using the blockchain to record an announcement of the exchange of monetary value.

So, we might use blockchain to record the fact that I paid you three Bitcoin for your used car or I paid you twelve blockchain for your vacation home.

This is the model that many of us start off with in blockchain and again, it's the one that's most familiar and gets talked about the most right now.

But if we peel away that outer layer, and we take away the idea of a monetary transaction or exchange,

well, then we're left with two or more parties using blockchain to record an important announcement.

And we see many valid use cases around this. Let's say, that I go to the doctor for my annual physical.

My doctor checks me out and he says: "Hey Kris, I want to put you on a new prescription for XYZ".

Well, maybe we're keeping our medical information on the blockchain, and so my doctor and I both agree that's a pretty good thing to add to my medical record.

In this case, we still need two or more parties, because it's very important that I, as a patient, don't have the ability to go and update my own medical record without a licensed medical professional involved.

And it's also equally important that my doctor is not able to update my medical record without my consent or permission.

And so, that's two or more parties coming together, recording an important announcement or important data point,

but nowhere in that example has money or anything of monetary value exchanged hands.

Now, if we peel away the very last layer, you can take away this idea of having to have two or more parties.

And what we're left with is just a single party announcing an important or significant event.

And this is the simplest type of blockchain transaction, and also the most powerful.

So, if we think about blockchain, perhaps for managing a supply chain scenario,

we might have a grower of organic produce announce or commit a record to the blockchain that says they've planted a crop.

And we might have an organic pesticide company come and treat that organic produce.

We still have a single organization or single entity making the announcement of something they did, and this is really the heart of blockchain.

When you understand that it becomes really easy to understand where smart contracts fit in.

And smart contracts are just computer code. They're codified logic that we can use to respond to any kind of event that gets captured on the blockchain.

So, in that organic produce example, if I want to notify someone when that fresh produce has been treated with organic pesticide, so I can create traceability for the end consumer,

well, I might have a smart contract that manages that that defines the rules and the steps that get taken when that particular type of event occurs.

And so this is all smart contracts are.

There's oftentimes a misconception when people hear the word contract that they think of a legally binding agreement between two or more parties.

It's not necessarily the case. Smart contracts don't have to be legally binding and they don't have to involve multiple parties.

In fact, if you come from a programming or development background you can think of a smart contract easily enough as a "class".

If you're not a programmer or don't come from a developing background, you can just think of a smart contract as a set of rules that get executed every time a certain type of event happens.

And this is where the real power of blockchain comes in that not only do we have a permanent, unchangeable record of all the different events that have occurred,

we can also write computer code, very, very non-subjective computer code, that defines exactly how that process is going to be managed and what steps are going to be taken when that event occurs.

This allows us to ensure process consistency.

It allows us to ensure that processes that are normally fulfilled by intermediaries or middlemen can now be satisfied on their own without the need for human intervention.

And this leads to much more lean and efficient organizations in way of organizing human effort.

So, when you hear smart contract just think about computer code that you write to respond to certain types of significant events.

If you've ever worked with workflow or business process automation or management tools, you can think of smart contract is just being another workflow tool, albeit one that has the power to work with money.

So, don't get too intimidated when you hear smart contracts, and don't get too caught up in the language.

While they sometimes may be legally binding, they're not necessarily legally binding in the same way a true contract is.

They're just a way for us to bake our own logic into block chain solutions to ensure that consistency of execution.

## Smart Contracts Recap

Let’s review some basic information about smart contracts:

* They are computer programs that execute an action based on terms and conditions.
* Can also be known as chain code.
* Smart contracts have terms recorded in a computer language instead of legal language. Smart contracts are not necessarily legally binding and do not need multiple participants.
* Each step of a smart contract can only be implemented after the execution of the immediate former step.
* Smart contracts are written in Solidity, a programming language designed for developing smart contracts that run on the Ethereum Virtual Machine.
* The smart contracts act as a foundation to build decentralized applications (DApps).
* Once executed, the information in the smart contract is stored on a blockchain which cannot be altered.



## What Do Smart Contracts Provide?

These are some of the features that smart contracts provide:

* Accuracy  
  Replacing human intermediaries with executable code ensures the process will always be performed the same.
* Cost savings  
  Replacing intermediaries often provides significant cost reduction.
* Efficiency  
  Removing process intermediaries often results in significant process efficiency gains.
* Backup  
  A blockchain and smart contract deployed to it can provide a permanent record, allowing for auditing, insight, and traceability, even if the creator is no longer in business.
* Autonomy  
  Smart contracts can be developed by anyone, no need for intermediaries such as lawyers, brokers, or auditors.

## Campaign Finance

One area getting a lot of attention in blockchain right now, is campaign finance and campaign donations.

There are some really interesting and compelling use cases, ways that we might be able to use blockchain to do new and innovative things around campaign finance that just aren't practical or feasible today.

Let's say that I'm running for office and you want to support my campaign.

Well, today, you might write me a thousand dollar check and I get all 1,000 of those dollars right away to use for my campaign.

Maybe, we want to use blockchain to empower voters to keep their politicians more accountable to the promises they make on the campaign trail.

Consider the following scenario.

Let's say, that instead of donating a thousand dollars to my campaign, you write a smart contract on the blockchain to manage campaign donations.

You give that smart contract a thousand dollars instead.

That smart contract has been programmed with the terms and conditions under which it is able to release those funds to my campaign.

I might get two hundred and fifty dollars of that thousand initially to use for my election.

If I get elected, the remaining $750 can stay in the smart contract.

Otherwise it gets refunded to you.

Let's say, I do get elected. Perhaps, I don't get the second 250 of that thousand dollars until a year in office, if I fulfilled three or more of my campaign trail promises.

And then, we might say as a final step, I get the remaining five hundred dollars to use for my reelection, only if I fulfilled 80% or more of my campaign promises during my first four-year term in office.

This would put a lot of power back in the hands of voters and would keep politicians accountable for the promises they make, long after election day.

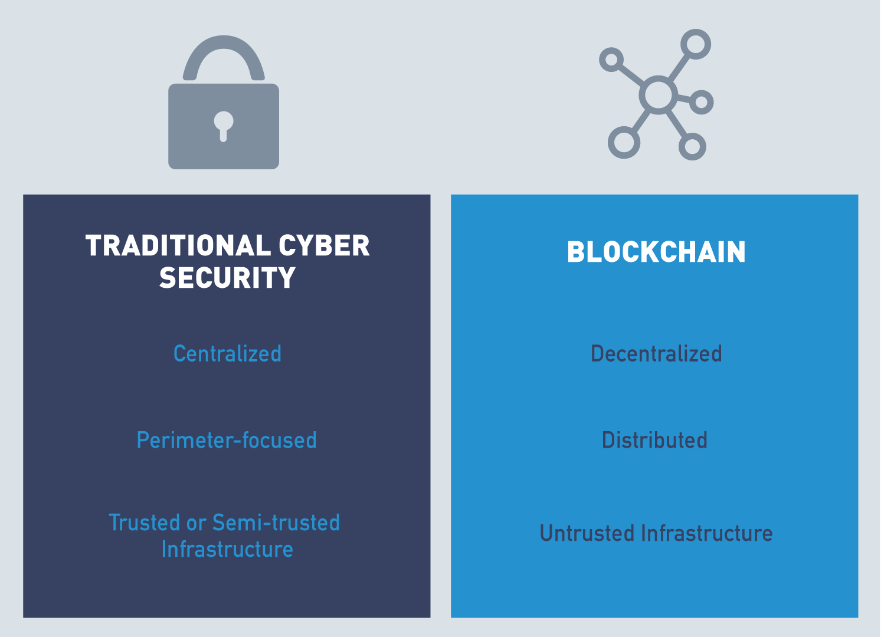
While things like this certainly are possible with conventional technology, they're not very practical or feasible.

Blockchain and smart contracts puts this power back in the hand of voters and has the potential to radically reform campaign finance and campaign donations.

## Blockchain Security vs. Standard Cybersecurity

Blockchain is commonly called the future of computing. It takes a very different approach to data storage and processing and requires a very different perspective for security.

In this section, we will discuss some of the ways that security differs in traditional and blockchain environments.



## Introduction to Blockchain Security vs. Standard Cybersecurity

In this section, we're gonna talk a bit about how traditional cybersecurity is different than security concerns on the blockchain.

In traditional cybersecurity that we're all used to, we're very perimeter-focused meaning that we put trusted data and assets behind a very, very strict perimeter.

We assume that anybody we let inside that perimeter is trusted, either partially or fully, to see that data and access those resources.

And the brunt of our focus goes into making sure that we keep the right people from getting beyond that perimeter,

that we only let authorized users through, and then we prevent any attackers or strangers from breaching that perimeter and accessing the data and resources we store behind it.

This is very different from blockchain, which runs on a massively decentralized peer-to-peer global network.

Just by its very definition, you can see that there is no perimeter on blockchain, and so our security has to come from somewhere else.

Our security on blockchain comes from the incredible amount of repetition.

The fact that we ask every node to keep the same copy of the ledger and periodically to reach consensus, majority consensus, on what the proper data in that ledger should be.

This makes the amount of work that an attacker would have to do practically impossible, if they wanted to change, hack, or alter the ledger.

Another big difference between traditional cybersecurity and security on the blockchain is with conventional applications and conventional technology.

We either develop all our own code or application parameters in-house or we rely on a single, third-party trusted provider who creates that line of business system for us.

All the code, generally, comes from one place, and if we don't write it ourselves, all we have to have is trust and faith in the third-party organization providing it to us.

In blockchain, things are a little bit different, because oftentimes in blockchain solutions we're writing smart contracts which call code from other smart contracts, which we may or may not write ourselves.

And which may come from a variety of different authors.

So, whereas in the traditional cybersecurity space, we really only need to rely on our own due diligence to write secure code or the due diligence of a third-party application provider.

In blockchain, we have to expend a lot more time and energy, making sure that the code and the contracts we're calling is safe and secure.

In other words, we want to make sure that we're not inheriting any security vulnerabilities from other smart contracts that have been written by unknown and potentially untrusted parties.

And the final big difference between conventional cybersecurity and blockchain is that in conventional line of business systems all of our applications either run directly on trusted hardware that we own as an organization or run on hardware with a high degree of trust provided by a third-party such as Microsoft Azure or Amazon Web Services.

In the blockchain world, all of our code runs on untrusted resources, untrusted hardware that can be owned by anyone, anywhere in the world.

This means that we must ensure that all the trust in a blockchain solution comes from the protocol and the code layer itself, because it's a much more massively distributed network, and we don't have direct control or access into the hardware that supports it.

So, there are a lot of differences between conventional cybersecurity and security in the blockchain space that you're going to want to understand before you deploy your own blockchain solutions.

## Blockchain Security Environment

One of the primary differences between cybersecurity in a traditional computing environment and on the blockchain is the environment itself and what it is and isn't designed to do.

**Standard Cyber Security**

* The traditional computing environment is a company network fully or, at least mostly, under the control of the company's computer security staff.
* While many organizations are making the shift to cloud-based environments, they still have a high degree of control over the security and configuration of their rented systems.
* Traditional networks are highly centralized, and the focus of cybersecurity on these systems is primarily perimeter-focused. All systems and authorized users on the network are trusted or semi-trusted, so the focus is on preventing attackers from entering from outside the network.

**Blockchain Cyber Security**

* Blockchains are designed to be decentralized, distributed systems running on untrusted hardware.
* While security in traditional environments is designed to provide security by putting all data in one place and building walls around it, security in blockchain is based on ensuring that data is protected from modification by copying data to as many locations as possible to make modification of all copies infeasible.
* Traditional infrastructure focuses on confidentiality and integrity, while blockchain is designed to provide integrity and availability.

## Security Attacks

Both traditional computing environments and blockchain have security considerations associated with them. In many cases, the same attack is possible against both paradigms, but the details of how to implement it vary.

Next, we will discuss how a few different attacks can be launched against traditional computing environments and blockchain.

## Security Attacks: Denial-of-Service (DoS)

A denial-of-service (DoS) attack is when an attacker makes it impossible for a system to serve its users as designed. This can be accomplished by exploiting a flaw in the system, but, more commonly, is accomplished by performing legitimate actions at a rate higher than the target can handle. To be effective, denial-of-service attacks typically focus on a system’s weakest link or bottleneck.

* In traditional environments, denial-of-service attacks target a company's web server to prevent customers from accessing the company's services. This can be accomplished by making more connection requests than the server is capable of supporting.
* In blockchain, a denial-of-service attack involves submitting more transactions to the blockchain than it can handle. Since many blockchains have fixed-size blocks created at a fixed rate and are stored in a distributed fashion, they have a maximum capacity that a determined attacker can exceed, rendering the blockchain unusable.

## Security Attacks: Endpoint Security

Traditional infrastructure and blockchain environments also differ with regard to endpoint security. Endpoint security originated to ensure multi-layer prevention against malware, fraud, cyber threats, and attempts to extract private information.

* In traditional cyber, endpoints are under the control of the enterprise and have some level of heterogeneity. Heterogeneity can be dangerous because an attacker has more options for finding a vulnerability to exploit.
* In blockchain, endpoints are the nodes and may be completely homogeneous. While homogeneity means that a flaw in one system is a flaw in all of the systems, blockchains built in trust and data privacy feature guard against such attacks.

## Security Attacks: Code Vulnerabilities

Computer applications are dependent on reliable code. Another way that traditional and blockchain cybersecurity differ is in the level of trust in the code used in a company's applications.

* In traditional cyber, the company writes most of the code, and vulnerabilities can arise only from code that the company controls.
* In blockchain, anyone can write a smart contract, and a flaw in the smart contract or the underlying platform code can have wide-reaching consequences. But since smart contracts can be vetted and reused, such inconsistency can be edited from the code. We will discuss an example of a vulnerability in a platform's code in a later section covering the Ethereum network.

## Security Attacks: Intentional Misuse

Both traditional and blockchain environments are vulnerable to attacks based on intentional misuse of the system.

* In traditional cyber, insider attacks or intentional misuse of the system by clients are possible. In fact, a denial-of-service attack is a specific type of intentional misuse.
* Blockchain systems are set up to incentivize network participants, miners or nodes, to do something that is in their best interest. A weakness is, if more than half of the network's processing power is controlled by a single group, the network is controlled by that group.

## Security Attacks: Data Protection

Finally, traditional infrastructure and blockchain differ in their goals regarding data protection.

* In traditional cyber, data is siloed, and access is strictly controlled by the owners, placing responsibility for confidentiality, integrity, and availability in their hands.
* In blockchain, data is distributed, and the blockchain is relied upon to provide integrity and availability.

## Lab 3: The Blockchain

Next, let's engage with an interactive lab. In this lab, you will be examining a blockchain containing five blocks, each labeled according to position. Enjoy!

[Start Lab](https://blockchaintrainingalliance.com/pages/lab-distributed)

## Different Types of Blockchains

When we try to understand the main difference between a public and private blockchain, it is important to understand the terminology.

* Public Blockchains  
  A public blockchain is open to anyone to join. Public blockchains are decentralized where no one has control over the network. These immutable and censorship resistant networks are ideal for participants who don't trust each other, but still interact in a network and take part in consensus. Bitcoin and Ethereum are examples of public, permissionless blockchains.
* Permissionless Blockchains  
  Public blockchains are permissionless. Anyone effectively can join the blockchain network. No one is prohibited from joining.
* Private Blockchains  
  Private blockchains define a user's rights on the network prior to admittance. Restrictions are placed on activities such as who can write to the ledger and what transaction they can participate in. Hyperledger and Corda blockchains are examples of private permissioned blockchains.
* Permissioned Blockchains  
  In a permissioned blockchain, participation is permissioned by an organization or a consortium of organizations. This empowered entity defines who can participate in certain transactions and consensus on the network.

## Public (Permissionless) Blockchains

When people get into blockchain, there's a natural discussion about what type of blockchain, because blockchain comes in many different types and flavors.

And normally, we hear about public versus private blockchains.

One factor that often gets left out is also the idea of an open versus closed blockchain.

And it's important to consider both parameters, so you know where on a possible solutions quadrant your idea falls.

When we talk about public and private, what we're really talking about is who is able to write data onto that blockchain or onto that immutable ledger.

The open versus closed brings in to consideration who's able to read that data.

And so, we can talk about solutions which are public and open, public and closed, private and open, private and closed.

When we talk about public blockchains, what most folks are really talking about is a public open blockchain.

It's a blockchain where anybody can come write data to the blockchain, anybody else can come read that data.

So, public blockchain platforms like Bitcoin, Ethereum, Litecoin tend to get talked about a lot right now,

and these are what we also refer to as permissionless blockchain platforms meaning that they really strive to, by design, increase and protect the user's anonymity.

And, if we don't know who a user is, if we don't have a way of identifying individuals, and we don't have any way of creating permission or access rules around that user,

and it's what we get, is the system where anybody can commit data to that blockchain and anybody can come along and read data to that blockchain.

So, there's a perception that public blockchain platforms like Ethereum can't be used to build permission scenarios or to control access to data.

The truth is they can, they just don't give you the built-in tools that a private or permission blockchain platform will.

So, you can always use these open public platforms to build a permission solution,

but just understand that it's upon you, your architects, and your developers to create that permissioning model,

and that all starts with some kind of identity management system.

So, when you think of public blockchains, you think about blockchain platforms like Ethereum, understand, by default, by the very nature and design, they're designed to protect anonymity.

And, if we don't know who a user is, then we really have no way of creating permissions, role-based access and controlling what data they can read or write.

In a lot of situations, this is desirable, this is why we see cryptocurrencies based on public blockchain platforms, because having that anonymity is important.

And, if a user has a currency, something of value, they should be able to exchange it and spend it, and do what they want with it, just like anybody else.

We don't want to treat any class of users differently than any others in those scenarios.

So, that's a public blockchain, and that's a very different animal from the private permission blockchain.

## Public Blockchain Benefits

In the [*"Public vs. Private Blockchain In a Nutshell"*](https://medium.com/coinmonks/public-vs-private-blockchain-in-a-nutshell-c9fe284fa39f) article by Demiro Massessi, public and private blockchains have different features. The benefits of public blockchain are:

* Ledger is distributed  
  The database is not centralized like in a client-server approach, and all nodes in the blockchain participate in the transaction validation.
* Immutable  
  When something is written to the blockchain, it cannot be changed.
* Open Consensus Model  
  No restriction placed on who can operate a node in the network.
* Open read and write  
  Anyone can participate by submitting transactions to the blockchain, such as Ethereum or Bitcoin; transactions can be viewed on the blockchain explorer.
* Low barrier to entry  
  Transactions on a public network can be accomplished with an Internet connection and a cell phone.
* Secure due to mining (51% rule)  
  With Bitcoin, obtaining a majority of network power could potentially enable massive double spending, and the ability to prevent transaction confirmations, among other potentially nefarious acts. This has never been accomplished as the computing power needed to stage such an attack would be too costly.

## Private (Permissioned) Blockchains

So, the counterpart to a public blockchain is naturally a private blockchain.

These are platforms like Hyperledger or Hashgraph.

And private blockchains are more specifically known as permissioned blockchains.

When we talk about private blockchain or when you hear folks talking about private blockchain solutions, they tend to be talking about things on the private and closed end of the spectrum.

We want to control who can write data to this blockchain, and we want to control who can read data from this blockchain.

And in order to do that, the first step is identity.

If we don't know who a user is, it becomes difficult, if not impossible, to define rules about what data they can commit to the ledger and what data they can consume from the ledger.

So, when we talk about private blockchain, just think about permission blockchain, the blockchain which right from the beginning is an idea of who you are.

And this is very different from a public platform like Ethereum, in which the platform tries to protect and maximize anonymity.

And we, by design, don't know who a user is, unless we build in that kind of identity management scheme.

Private blockchains tend to come with identity management tools or a modular architecture, where you can plug in your own identity management solution.

This can be anything from an Active Directory deployment to an OAuth solution using Google, Facebook, LinkedIn, etc.

Just understand the idea behind the private blockchain.

All begins with understanding who a user is, because once we understand who a user is,

we can determine what role they're in, and we can use that role to determine what information should they and should they not have access to.

This also changes the incentive for good behavior a little bit in a public blockchain, because we don't know who a user is.

We rely on economics and game theory incentives to ensure that everybody in the system behaves honestly and according to the rules.

We set up situations through group consensus, which we discuss in other sections in this course, through which honest participants are economically rewarded,

where dishonest ones only incur work or cost, with no possibility of ever recouping that cost.

In a private permission blockchain, we rely on the fact that we know who a user is.

So, in a corporate scenario, blockchain for the business, blockchain for supply value chains, because we know who an individual is, what organization they're associated with and what their role is,

we also assume that they're going to behave fairly aboveboard, because if not, we know exactly who's misbehaving and they know that they're gonna suffer the consequences for that.

So, public and private blockchains, two very, very different offerings.

A lot of people make the impression, get the impression that they compete with one another, when really they don't.

They just serve to provide different types of solutions and enable different kinds of products and offerings to be built on top of them.

## Private Blockchain Benefits

The following benefits of a private blockchain have been described in the [*"Public vs. Private Blockchain In a Nutshell"*](https://medium.com/coinmonks/public-vs-private-blockchain-in-a-nutshell-c9fe284fa39f) article by Demiro Massessi:

* Ledger is distributed/immutable  
  The database is not centralized like in a client-server approach, and all nodes in the blockchain participate in the transaction validation. When something is written to the blockchain, it cannot be changed.
* Compliance support  
  As an enterprise, you likely would have compliance requirements to adhere to, and having control of your infrastructure would enable this requirement more seamlessly.
* Faster transactions  
  When you distribute the nodes locally, but also have much less nodes to participate in the ledger, the performance is faster.
* Consensus more efficient/more options  
  More options for consensus models. Enterprise or private blockchains have less nodes and usually have a different consensus algorithm, such as BFT vs. POW, which we will discuss in a later section.
* Better scalability  
  Being able to add nodes and services on demand can provide a great advantage to the enterprise.
* Enterprise permissioned  
  The enterprise controls the resources and access to the blockchain, hence private and/or permissioned.

## Security: Public vs. Private Blockchains

As discussed in [*"The Difference Between Public and Private Blockchain"*](https://www.ibm.com/blogs/blockchain/2017/05/the-difference-between-public-and-private-blockchain/) article by Praveen Jayachandran, both public and private blockchains share many similarities:

* Both are decentralized peer-to-peer networks, each maintaining a shared append-only ledger of digitally-signed transactions.
* Both maintain transaction replicas in-sync through a protocol referred to as consensus.
* Both provide certain guarantees on the immutability of the ledger.

On the other hand, as described in [Investopedia](https://www.investopedia.com/news/public-private-permissioned-blockchains-compared/#:~:text=The%20primary%20distinction%20between%20the,and%20maintain%20the%20shared%20ledger), the main difference between a public and private blockchain is related to who is allowed to participate in the network, execute the consensus protocol, and maintain the shared ledger:

* A public blockchain network is completely open and anyone can join and participate in the network.
* A private blockchain network requires an invitation, and must be validated by either the network starter or by a set of rules. Private blockchains are usually set up as permissioned networks, placing restrictions on who is allowed to participate in the network, and only in certain transactions.

## Public and Private Comparison

Now that we've talked about public and private blockchains what those terms mean and again you can think of that as permission versus permissionless blockchains we can talk about some of the differences and what each different type of platform aims to provide so public blockchains are really good for scenarios where protecting the anonymity of users is important or adds value to the solution and they're really great platforms for solutions where all users should be treated equally and we can see this because most public blockchains right now are used to support crypto currencies and this is a real great use case with a crypto currency like Bitcoin litecoin ether etc we don't want to have any permissioning or role based access anybody should be able to own some Bitcoin anybody should be able to trade it with anybody else and it's not necessarily important that we know who the participants in any transaction are and so that leads to an open permissionless model with full transparency this is very different in the corporate world where we see private blockchains being adopted at scale because the concerns are the opposite in a corporate scenario in a business scenario anonymity is a bad thing I want to know exactly who all the participants are and I need to know who they are because I don't want full transparency I don't want to share all my business data with all the participants in my business network or the general public at large I want to control who sees what type of information under what circumstances and I also want to control who is able to contribute that information onto the blockchain so I might use a private blockchain solution to manage supplier vendor relationships where only myself and my suppliers can see the price that I'm paying for a certain item only a particular supplier gets to see the details of the contract that I have and not the details of a contract I have with any other suppliers and I may wish to share some of this high-level macro data with consumers so that they can see the origins of the products they're buying but of course I'd want to hide the financials behind all that so when we look at how these two solution types differ we see the public blockchains tend to focus more on b2c or business to

consumer scenarios whereas private

blockchain offerings like hash graph

hyper ledger really lend themselves well

to be to be scenarios supply chain value

chain relationships or creating any kind

of shared infrastructure between

enterprises so just to understand that

well there's a big misconception that

these are competing offerings they're

really not and many real-world use cases

use components of both it's important to

understand the difference between them

and it really all starts with Identity

Management so when you're thinking in

your head about public versus private

blockchain to understand that it all

begins with Identity Management and in a

private blockchain I know who all the

participants are right from the

beginning in the public blockchain again

I don't know who those participants are

and that's not to say that I can't build

a permission solution on a public

platform but myself my architects and my

developers are gonna have to develop the

logic and the mechanisms behind identity

management so that's public and private

blockchain two very very different

animals they serve vastly different

purposes and as you start to dig into a

lot of real-world use cases what you're

going to find is many use cases make use

of both types of blockchain integrated

seamlessly

you

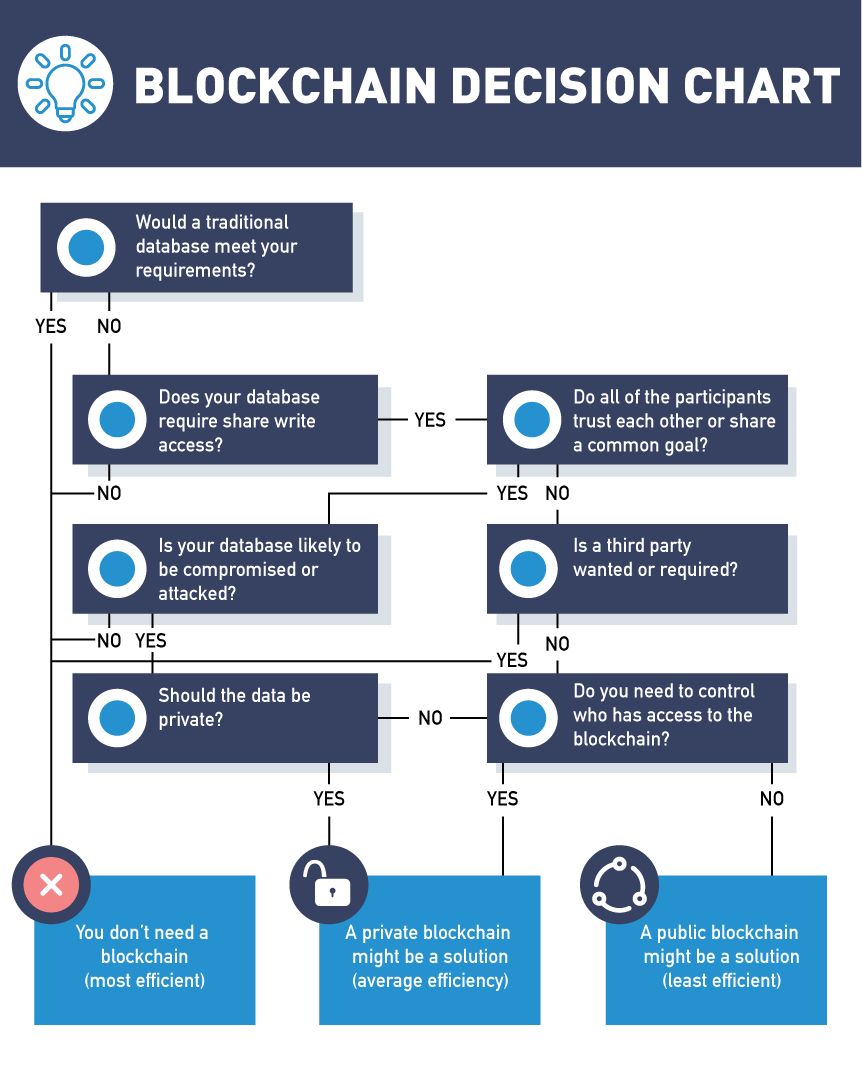
## Is Blockchain Right for You?

Consider the following aspects before deciding if blockchain is for you:

* Blockchain decisions  
  When it comes to decision making around which blockchain model to use, it's important to determine which platform and structure is right for your business.
* Intermediaries  
  Blockchain implementations can eliminate middlemen from work processes.
* Central authority  
  Some organizations rely on one person having control, with blockchain decentralized work flows without a single point of failure are emerging.
* Multi-party interaction  
  Blockchain provides a trust layer that opens up new opportunities for participation.
* Visibility  
  Blockchain transparency reduces dispute over what is in the ledger since agreement occurs before transactions are committed and everyone can have access to the ledger.
* Write access  
  Public permissionless blockchains do not prohibit who can add blocks to the chain, where private permissioned can restrict access.
* Value transfer  
  Blockchains can differ in the ways participants transfer value.

## Blockchain Decision Chart

Let's take a look at the blockchain decision chart:



## Questions to Ask When Deciding on Blockchain

Discussion on these topics need to be addressed when making the decision to adopt a blockchain strategy for your company:



## Blockchain Is Like Hot Sauce

One important thing to keep in mind as you're evaluating blockchain, is it's like hot sauce.

And I want to give credit where credit is due for this analogy.

This was inspired by a friend of mine, Samson Williams, a thought leader in the blockchain space.

But Samson loves to talk about blockchain being a lot like hot sauce.

Hot sauce is surprising because you can put it on a number of things and it really tastes good.

But there are some things hot sauce just doesn't belong on.

You'd never put hot sauce on an ice cream.

Understand that blockchain is like hot sauce - there are a lot of areas you can add a little splash of blockchain to, and end up making a much better and more compelling solution.

But there are some areas that you can't add blockchain to without making things a lot worse.

And just like hot sauce, there is a big misconception that blockchain is an all-or-nothing proposition,

that if we're going to architect and develop a solution on blockchain,

we either have to build something that exists entirely on blockchain, or we need to build something that exists entirely on conventional technology.

The truth is that like hot sauce, blockchain is really best when it's combined with something.

If you were hungry and poured yourself a big bowl of hot sauce, not only would you be unsatisfied and unhappy, but you wouldn't want to repeat the process again.

That's the same way with blockchain.

If you are looking at building a solution completely and entirely on the blockchain,

without exploring some of the amazing potential that you get by combining blockchain with conventional technology, then you probably got a bad recipe.

So, when you're thinking about blockchain, and how to add it to your personal and professional life, understand that it's not a meal all by itself.

It's just a little splash of hot sauce you put on something that already works pretty well.

## Blockchain Considerations

Companies must consider many factors when deciding how blockchain will fit into the business model. There is not a “one size fits all” blockchain but rather a well-thought-out approach that considers many options.

* Governance  
  Governance of blockchains deals with a system for managing and implementing changes to blockchains. Chapter 4, “Blockchains and Governance”, covers how blockchain is managed and how changes are implemented.
* Smart contract functionality  
  Each organization has unique needs when deciding on smart contract functionality. Such decisions as to the design, implementation and testing of these digital promises need to be determined. Can smart contracts replace or accentuate current work flows? Can a previously written smart contact fit in the current business model?
* Consensus algorithms  
  Consensus models change depending on the type of blockchain environment you are creating. In order for a blockchain to work, the participants must come to an agreement as to the transactions in the ledger or the state of the ledger. This agreement is called consensus. Public blockchains have miners that are incentivized to reach consensus whereas private blockchain set up enterprise nodes that agree to the state of the ledger. We will discuss consensus models later in this chapter.
* Integration  
  Consideration for current work flows and how to bridge blockchain platforms with existing systems are a vital conversation. How can blockchain add value to your current organization while leveraging your existing systems?
* Cryptocurrency requirements  
  As mentioned previously, a cryptocurrency is a medium of exchange digitally enabled by blockchain. Cryptocurrencies secure and keep track of assets on a public blockchain. Enterprise blockchains generally do not use the cryptocurrency feature. Decisions on if and how your blockchain will exchange value among participants are important discussions that need to occur during the design phase.
* Cost modeling  
  New questions arise around consortium type blockchains, one consideration is how to split the technology cost among the participants in a fair and equitable manner.

Overall, blockchain decisions will be made by actors in the blockchain system:

* System Architects design and build blockchain networks. This includes decisions on node types and membership rights.
* System Developers design the smart contracts for the community to use.
* System Operators interact with the software, permissioning and onboarding participants. This includes storing the ledger, creating wallets and managing credentials.
* Blockchain Users join a blockchain by setting up a wallet and/or receiving permission to engage in a particular function.

## Blockchain Transaction Flow

In this section, we'll discuss the blockchain transaction flow.

Regardless of the blockchain technology, the flow of data through a blockchain will remain relatively the same, at least at a high level.

It will all start with a blockchain user performing an operation that should be stored on the chain.

The operation will trigger a smart contract, or it will trigger chaincode.

In either case, the code will be executed on the actual blockchain itself.

The output of this code will result in a transaction, and that transaction is what we intend to be stored on the blockchain.

Next, the blockchain operators using protocols that will be specific to the specific blockchain that's running on will spread that transaction throughout the blockchain network.

The block creators will collect all of the transactions and it will create a new block.

In most blockchains, this is the job of the miner and is the output of the mining process.

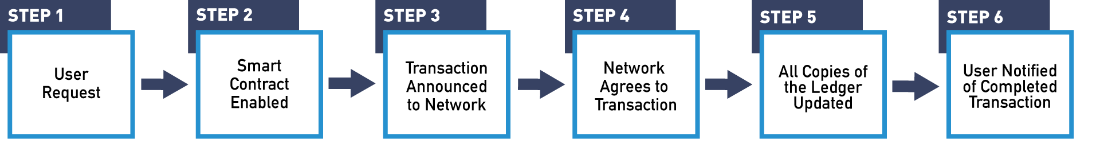
The block operators will spread the block through the peer network to update the ledger copies once consensus has been achieved.

And this results in the block operators again executing code included in transactions within the block to update their own copy of the internal state of the blockchain.

The users will be notified of an event from a blockchain creation via an event.

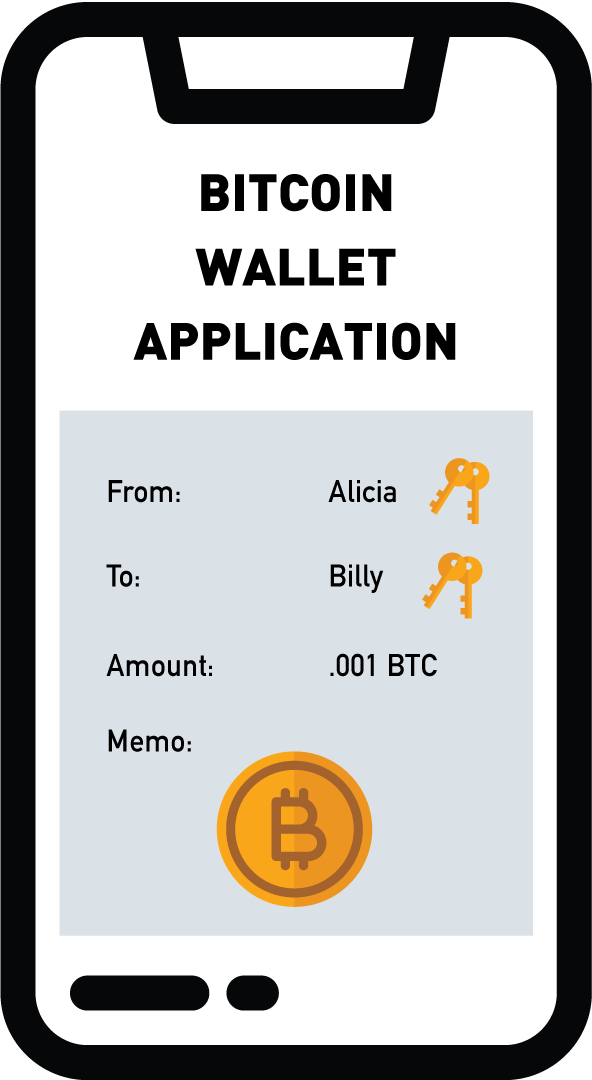
## Blockchain Transactions

Although blockchains can differ in form and function, all follow a basic transaction flow.



**Step 1: Operation Initiated**

Let's follow the step with a basic Bitcoin transaction. If Alicia wants to send Billy some bitcoin, she would go to a Bitcoin wallet application either on her phone or computer where she has some Bitcoin stored. Wallet applications are usually free downloads that create a digital wallet to purchase and/or store your cryptocurrency.



It is worth mentioning here that cryptocurrencies are a function of a particular blockchain, the Bitcoin blockchain only transacts bitcoin and the Ethereum blockchain can only exchange Ether. To initiate the transaction, Alicia would make a request of the system to send Bitcoin to Billy’s wallet by specifying the amount and Billy’s Bitcoin address.

**Step 2: Smart Contract Triggered**

When Alicia sends her transaction to the network, it triggers a smart contract that checks with the nodes on the network to make sure Alicia has the currency to spend, and that she hasn't already spent it. Once checked, the transaction is added to a proposed block.

**Step 3: Operators Spread Transaction**

Proposed block is communicated to the network through the peer-to-peer protocols.

**Step 4: Consensus**

In order for the Bitcoin network to validate a block, nodes or miners must validate the correctness of a block by completing a math problem first. The node that completes the Proof of Work equation first is rewarded with some newly minted Bitcoin. Once a solution for the equation is reached, the other nodes can easily check its accuracy, thereby accepting the new block onto the blockchain.

**Step 5: Spread The New Block**

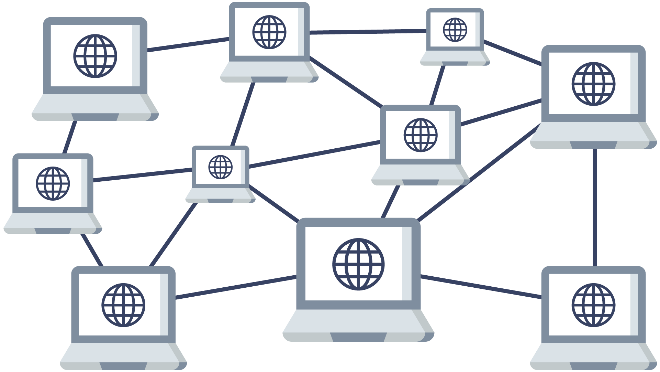
This block is spread throughout the network through the same peer-to-peer communications we used for the transaction. When block operators receive a copy of the new block, they add it to their copy of the distributed ledger. This guarantees that all members of the peer network agree on the current state of the blockchain’s distributed ledger.

**Step 6: Transaction Completed**

The user’s wallet monitors for the creation of new blocks that include transactions associated with the user. When a block containing the completed code from the user’s operation is received, an event is created to notify the user that the operation is complete. When the block containing Alicia’s Bitcoin transfer to Billy is added to the blockchain, an alert will be sent to the affected wallets that the transaction was accepted and completed.

## Consensus in Blockchain

Consensus in blockchain is moving away from a centralized system where one entity keeps track of the ledger towards a decentralized system where trust does not rest with one entity but all entities and what they decide together. The blockchain is a distributed and decentralized system, which means that it needs to have a way of tracking the official current state of the system. Since the blockchain can include financial transactions and business agreements, it is important that all parties involved are in sync regarding the terms of the agreement.



In this section, we will discuss the details of how a blockchain network comes to agreement on the contents of the blockchain.

## Consensus

So, one of the most important components to blockchain is this idea of group consensus.

As we talked about in an earlier module, blockchain is inherently a very inefficient system.

We're asking multiple nodes, sometimes tens of thousands of computer nodes, to all repeat the same work: they're all keeping a copy of the same data.

And the reason we agree to this tremendous inefficiency is because if we can get all or most of those nodes to agree on what the truth is.

We can have a lot of trust that that's actually the truth, that that record hasn't been tampered or altered or changed in any way.

So, consensus is one of the underpinnings of blockchain.

There are several different methods we use right now to have all these nodes reach consensus.

Essentially, when we talk about consensus, you can think of every block in a blockchain as being like a sheet of paper.

It's got a fixed amount of space. We write a transaction on every line, and when that sheet of paper is full,

it's important that we all, as a group, come together and compare our different sheets and select the sheet or the version of paper that the majority agree with.

And so, there are several different methods that we use to come to consensus on a block.

The oldest and most widespread method is what's called Proof of Work.

And you'll see in another module, we actually dive into Proof of Work hands-on, and show you live what Proof of Work looks like and how it actually works.

Proof of Work has served us well for the past nine, almost ten years, got its start in Bitcoin, and it's used in every major public and most private blockchain offerings.

But we're also starting to see some of the limitations of Proof of Work.

One of the big limitations behind Proof of Work right now is how big and how fast it can scale.

Currently on Proof of Work blockchains, we're able to process somewhere between 50 and 20 transactions worldwide per second,

which may sound like a lot until you realize that modern payment processing networks like Visa can scale up to over 70,000 transactions a second.

So, in order to compete with conventional technology, blockchain really needs to add a few more orders of magnitude to that transaction rate.

There are many proposed alternative consensus methods for how we might be able to reach that kind of scale.

There are things in production right now like Tangle which use a blockless solution,

and they're also new and emerging consensus methods like Proof of Stake, or Proof of Activity that we're currently examining to take the work out of Proof of Work.

We also have another module in this course where we talk about Proof of Stake, and then another one where we compare Proof of Work versus Proof of Stake.

So, if you're curious about any of the details of how these consensus mechanisms are actually implemented,

be sure to check out those modules.

But the takeaway point to understand is that, it's this consensus, it is this idea of asking all of these nodes, potentially tens of thousands of nodes,

to all repeat the same work and then periodically come together and agree on whatever the majority select the right version of the truth to be.

That gives blockchain that high level of trust and that makes it such a secure record store.

## Introduction to Consensus in the Blockchain

As discussed in the [*"Blockchain Consensus and Fault Tolerance In a Nutshell"*](https://medium.com/coinmonks/blockchain-consensus-and-fault-tolerance-in-a-nutshell-765de83b8d03) article by Demiro Massessi,

"The blockchain is designed to be a shared, synchronized historical ledger, meaning that there needs to be a final decision at some point on what should and shouldn’t be included in the official record. Since blockchain is decentralized, there is no "higher authority" that can rubber-stamp and finalize the contents of a blockchain block.

The method that [*Satoshi Nakamoto*](https://en.wikipedia.org/wiki/Satoshi_Nakamoto), creator of the Bitcoin network, invented to achieve consensus is based on scarcity. In one way or another, blockchain consensus algorithms boil down to some kind of vote where the number of votes that a user has is tied to the amount of a limited resource that is under the user’s control. Based on the economic law of supply and demand, collecting enough of an asset to have a controlling share will drive up the price of the asset enough to make achieving that level of control unfeasibly expensive".

## Consensus Mechanisms

© 2021 edX Inc. All rights reserved.  
深圳市恒宇博科技有限公司 [粤ICP备17044299号-2](http://www.beian.miit.gov.cn/)