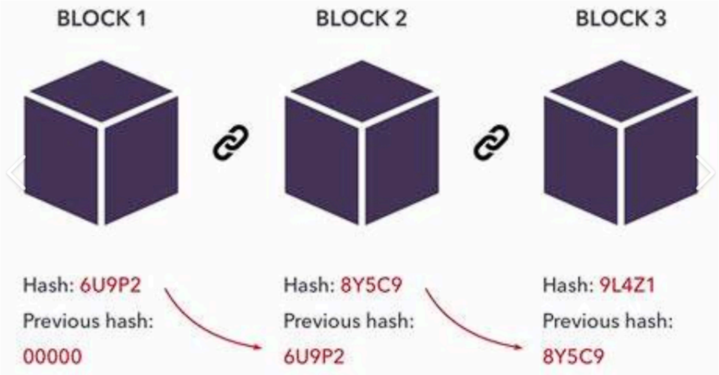
Introduction to Blockchain Technology

What is blockchain?

A method for introducing a secure and decentralized record of enormous transactions. It's a **digital ledger** that is distributed across a network of computers (called **nodes**), ensuring that **no single entity** has control over the entire **history of transactions** in **multiple places** at the **same time** via a **few processes**. There are **decentralization**, **distributed ledger**, **cryptography**, etc neither single entity from failures. Each **"BLOCK"** in the blockchain **contains** a **number of transactions**, and every time a transaction **occurs** on the blockchain, the record of the transaction is **added** subsequently to every **participant's ledger**.



Unique aspect of blockchain is achieved through **cryptographic techniques/hashes**, ensuring that a transaction is **recorded** in a **block**, it cannot be **altered retroactively** without **altering** all **subsequent blocks**, the unique characteristics makes blockchain **valuable** for the system that requires an immutable and transparent record of transactions.

Origin

First outlined in 1991 by Stuart Haver and W.Scott Stornetta, two researchers who wanted to implement a **system** where **document timestamps** could not be **tampered** with. Invention of **Bitcoin**, a **digital currency**, by an individual under pseudonym **Satoshi Nakamoto** in 2008, entitled "Bitcoin: A Peer-to-Peer Electronic Cash System," to create a **decentralized currency** free from central authority or government control.

1. **"Bitcoin: A Peer-to-Peer Electronic Cash System"** by **Satoshi Nakamoto**:
   A purely p2p version of electronic cash would allow online transactions/payments to be sent directly from one party to another without going through a financial institution.

Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double spending, therefore, a solution proposed to double-spending using a p2p network. The network timestamps transactions by **hashing** them into an **ongoing chain** of hash-based proof-of-work, forming a record that **cannot be changed** without **redoing** the proof-of-work. The **longest chain** not only serves as **proof of** the sequence of **events witnessed**, but proof that it came from the largest pool of the **CPU** or the **power** of a machine. As long as the majority of CPU power is controlled by **nodes** that are not cooperating to attack the network, they will generate the **longest chain** and outpace attackers. The network itself requires minimal structure. **Messages** are **broadcast** on a best effort basis, and **nodes** can **leave** and **rejoin** the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

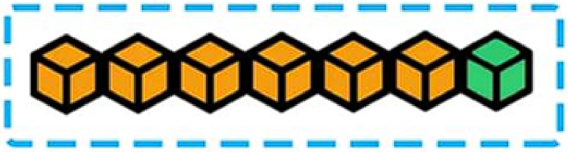
2. **"Digital Gold: Bitcoin and the Inside Story of the Misfits and Millionaires Trying to Reinvent Money"** by **Nathaniel Popper**
   The author introduces us to the **central characters** who played pivotal roles in **Bitcoin's development**. From an **Argentinian millionaire** to a **Chinese entrepreneur**, and even the elusive creator of Bitcoin, Satoshi Nakamoto. **"Digital Gold"** explores the notion of a new currency maintained by users' computers worldwide which is a concept that has both fascinated. Bitcoin has grown into a technology worth billions of dollars, with followers who view it as a groundbreaking idea to the creation of the Internet. **Believers** from **Beijing to Buenos Aires** envision a **financial system free** from **traditional banks** and **government control**, and **Bitcoin** has the potential to **decentralize** some of society's fundamental institutions.

3. **"The Basics of Bitcoins and Blockchains"** by **Antony Lewis**
   Bitcoin, Ethereum, and other cryptocurrencies. Gain an understanding of a broad spectrum of Bitcoin topics including the **history of Bitcoin**, the Bitcoin blockchain, and Bitcoin buying, selling, and mining. Learn how payments are made, and how to put a **value on cryptocurrencies** and **digital tokens**. **Blockchain technology**. Learn about notable **blockchain platforms**, **smart contracts**, and other important facets of blockchains and their function in the changing **cyber-economy**. Things to know before buying cryptocurrencies. Find trustworthy and balanced insights into Bitcoin investing and investing in other cryptocurrency. Discover the **risks** and **mitigations**, learn how to **identify scams**, and **understand cryptocurrency exchanges**, digital wallets, and regulations.

Blockchain Process

1. Someone requests a **transactions.**
2. The **requested transaction** is **broadcast** to a **Peer-to-Peer** network consisting of computers known as **nodes.**

3. The P2P network of nodes **validated** the transaction and the user's status using known **algorithms**.
4. Once **verified**, the transaction is **combined** with other transactions to create a **new** block of data for the **ledger.**



5. The **new block** is then **added** to the **existing blockchain** in a way that is **permanent** and **unalterable.**

   Blocks hold **batches** of **valid transactions** that are **hashed** and encoded into a **Merkle tree**. Each block includes the cryptographic hash of the prior block in the blockchain, linking the two. The **linked blocks** form a **chain**. To assure the integrity of a block and the data contained in it, the block is usually digitally signed.

   Block time is the **average time** it **takes** for the network to **generate** one **extra block** in the blockchain. During the **block completion**, the included data become **verifiable**. In cryptocurrency, when the transaction takes place, a **shorter block time** means **faster transaction**. Block time for Ethereum is set to between 14 and 15 seconds.
6. The transaction is completed.

Blockchain Characteristics

1. Immutability
   Ensures that once data (**transactions**) is **recorded** on the blockchain, it becomes **tamper-resistant** and virtually **impossible to alter**. The cryptographic nature of blockchain and the **interlinking** of blocks **through hashes** create a secure and unchangeable history of transactions.

2. Decentralization
   By storing data across its p2p network. There is **no central authority** or i**ntermediary controlling** the blockchain network. Instead, it operates on a **distributed network of nodes** where **each node** has its own **copy** of the entire blockchain. This **eliminates** the **need for intermediaries**, like banks or government control entities, **fostering a peer-to-peer environment.** Allowing **double-spending** (fundamental flaw in a digital cask protocol in which the same single digital token can be spent more than once).

   A **public key** (a long, random-looking string of numbers) is an address on the blockchain; **value tokens** sent across the network are recorded as belonging to the address. A **private key** is like a password that gives its owner **access** to their digital assets.

Every **node** in a decentralized system has a **copy** of the blockchain. **Data quality** is maintained by massive **database replication**. **Transactions** are **broadcast** to the **network** using the software. Blockchains use various **timestamping schemes**, such as **proof-of-work**, to **serialize** changes; later **consensus methods** include **proof of stake**. The **growth** of decentralized blockchain is accompanied by the **risk of centralization** as the **CPU** resources required to process larger amounts of data are also **consuming** resources.

3. Transparency
   Ensures transparency by making all **transactions visible** to every participant in the network. Once a **transaction is added** to the blockchain, it becomes a **permanent** and **auditable record** that anyone on the network can **inspect/withdraw**.

4. Trust
   The **immutable recording** of transactions in a **distributed ledger** ensures that the data is reliable and **cannot be manipulated**. Verification and validation processes, often achieved through **consensus mechanisms** like Proof of Work or Proof of Stake, further enhance trust by **requiring agreement** among **network participants** before a **transaction is** considered **valid**.

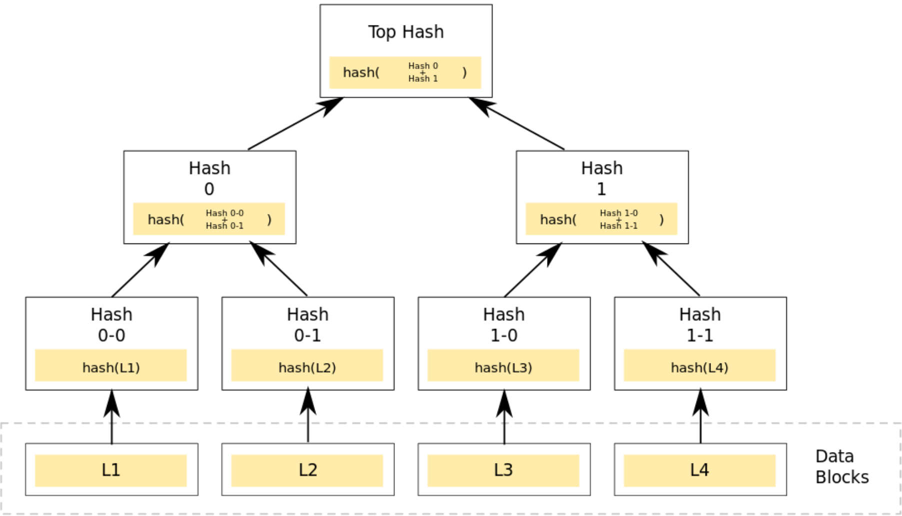
   In summary, **data immutability** ensures the **security** and **permanence (remaining/unchanged/lasting)** of recorded **transactions**, **decentralization** eliminates the need for **intermediaries**, **transparency** provides **visibility** into the transaction history, and **trust** is established through a combination of immutable recording, **verification processes**, and **consensus mechanisms**.

History and Evolution of Blockchain

1. Cryptographer **David Chaum** first proposed a blockchain-like protocol in 1982.

2. Early foundations (1990s)
   Researchers Stuart Haber and W. Scott Stornetta **proposed** a cryptographically secure chain of blocks to **timestamp digital documents**, preventing **backdating** or **tampering**. In 1992, they incorporated Merkle trees/hash tree into the design, which improved its efficiency by allowing several document certificates to be collected into one block.

   Merkle tree is a tree in which every leaf/node is labelled with the cryptographic hash of data block, and every node that is not a leaf (called a branch) is labelled with cryptographic hash of the labels of its child nodes.



3. The Bitcoin Revolution (2008-2009)
   True birth of blockchain, **Satoshi Nakamoto**, marked the true genesis of blockchain. Born in the aftermath of the global financial crisis, Nakamoto aimed to establish a **decentralized monetary** system free from **central control**. Nakamoto improved the design using a Hashcash method to timestamp blocks without requiring them to be signed by a trusted party and introducing a difficulty parameter to stabilize the rate at which blocks are added to the chain. **Bitcoin**, the first use of blockchain, introduced a revolutionary method for **securing** and **validating transactions** through a **distributed ledger**. The operational launch of the Bitcoin (currency exchange only) blockchain in January 2009, with the mining of the "Genesis Block," represented the practical realization of blockchain technology's potential.

   Native currency: Bitcoin

4. Beyond Bitcoin (2010-2017)

   The potential of blockchain began to be recognized beyond just **powering cryptocurrencies**. Developers and entrepreneurs started exploring other uses of blockchain for creating **decentralized applications (DApps)** beyond currency transactions.

5. **Ethereum** (2015) [**20k to 100k transactions/sec**]

   One of the significant developments was the introduction of **Ethereum** in 2015 by Vitalik Buterin, Gavin Wood and Jeffrey Wilcke. Ethereum expanded upon Bitcoin's capabilities, introducing currency exchange and the amalgamation of **smart contracts** with the terms of the **agreement** directly **written into code**, enabling **more complex** and **automated interactions** on the **blockchain**. Ethereum marked the shift from blockchain as a financial tool to a multi-functional platform and can be used for all kinds of blockchain applications.

   Smart Contracts (Short Definition)

   Provide the capability of "**code execution**" for embedding business logic on Blockchain. **Self-enforcing agreements** (Only enforced by the parties to it; no external party interference) embedded in computer code. Allow the performance of credible transactions without third parties. Broader applications for blockchain technology.

   Native Currency: Ether

6. Hyperledger Fabric (2017) **[20k translation/sec]**

   Modular blockchain framework that serves as a foundation for developing blockchain-based products, solutions, and applications within private enterprises such as banks, financial institutions, and supply chain networks, can utilize Hyperledger Fabric to build secure and efficient blockchain solutions.

   From the perspective of **Architecture**, Hyperledger Fabric provides a modular architecture that allows **interchangeable** components. These components include **consensus mechanisms** and **membership services**. It enables a **plug-and-play** environment, making it **flexible** for various use cases. **Fabric** is designed to meet **diverse industry needs**, suits for a wide range of applications. To ensure privacy, **Fabric** offers a unique approach to **consensus** that balances **scalable performance** with privacy. It ensures robustness, customization, and scalability for **distributed ledger applications**.

7. Maturation and Diversification (2018-Present)

   Various new blockchain platforms have **emerged**, each with specific focuses like improved scalability, security, or interoperability. Major industries, including finance, healthcare, supply chain, and even governments, have begun adopting blockchain for its **transparency, security**, and **efficiency**. The technology's evolution has also **sparked**

debates and **research** in **areas** like **scalability** (e.g., **Bitcoin's Lightning Network**), **sustainability** (concerns over **energy consumption** in **Proof of Work** models), and **regulatory acceptance**.

References

“Blockchain Revolution” by **Don Tapscott** and **Alex Tapscott** explores the transformative potential of blockchain technology across various industries. The authors emphasize the **decentralized** nature of blockchain as a **disruptive force** that eliminates the centralized systems. Enables **trust** among participants without the need for intermediaries. The concept of **smart contracts** is explored, showcasing how **self-executing contracts** embedded in code can **automate** and **streamline** various processes, reducing the need for intermediaries.

“Banking on Bitcoin”, an overview of the **history of Bitcoin**. Released in 2016, directed by **Christopher Cannucciari**, the documentary explores the rise and development of Bitcoin. It explores the motivations behind the development of this **decentralized digital currency** and its potential to **challenge traditional financial systems**. "Banking on Bitcoin" explains the underlying technology behind Bitcoin – blockchain. It introduces the **concept** of a **decentralized ledger**, highlighting how blockchain ensures **transparency**, **security,** and **trust** in the Bitcoin network. The documentary features interviews with **early adopters** of Bitcoin, including developers, entrepreneurs, and enthusiasts who played crucial roles in its initial growth. It explores their perspectives on the **disruptive potential** of this new form of **currency**.

The need for web3 and how it is different from web2

Web 1.0 (frontend)

First iteration of the internet was primarily a collection of **static websites** or the **“Read-only Web”**, serving as a **digital information repository** and served with **limited interactivity**. During the phase, users only passively consume the **information displayed** on websites, **lacking interaction** of the dynamic and interactive features. Typically, web1 were **informational** and presented in a **one-way communication format**, and the overall web experience was largely **static and read only** with **minimal user engagement**.

Web 2.0 (Centralized)

The internet known as **social web** or **“Read-Write Web”**, is characterized by **interactive experiences**, social media, and **user-generated content (UGC)** or the **end user’s experience**. Web 2.0 was responsible for creating communities, collaborations, dialogue and social media. The internet transformed into a **dynamic platform** where users could **actively contribute**, share and collaborate from blogging. Besides, Web2 supports key components such as **modern frontend backend technologies** (responds to the user’s input), cloud computing, Developed Application Programming Interfaces (API) and databases. **Emphasize engagement** and **interconnected online experiences**. Mobile Internet access has contributed to Web2.0's growth. Mobile devices such as Android-powered devices and Iphones are also responsible for it. In addition, Web 2.0’s growth includes mobile applications such as Tik Tok, Twitter and Youtube to expand and dominate the online platform. However, Web2 **centralized the control of data** and platforms in the hands of a few major companies leading to **concerns over privacy**, **data ownership** and **monopolistic practices**.

Web 3.0 (read, write and execute)

Web3 focus shifts to **decentralization**, increased **privacy**, a more **secure** and **transparent** internet infrastructure, openness and **greater user empowermen**t. Web3 is built upon the backbone of blockchain technology, which allows for decentralized and **p2p interactions without** the need for **intermediaries**. Web3 will be anywhere, **accessible** from any device and can be accessed **anytime by anyone**. The goal is to reduce reliance on centralized authorities, enhance **user control** over personal data. Where individuals have **greater ownership** and **control over** their **digital identities** and **assets**. In Web3, blockchain networks become the replacement for traditional and centrally managed databases and applications that **gate user’s access** to content, and store and manage their data. With blockchain, users **no longer** create a **username and password** on a centralized server that a central authority could lock them out of, shut down, or limit access to. Instead, users **connect to sites** and **applications** that have some or all of their components **hosted** on **blockchain networks**, making them partially or fully decentralized. These decentralized apps and sites on Web3 are often called “DApps”. The blockchain’s encrypted nature ensures individuals and businesses **benefit** from secure

transactions when interacting online. This **immutable**, secure technology makes it ideal for storing **sensitive data** and powering **decentralized** applications.

What’s cryptography and how is it used in Web3?

The field of **cryptography** deals with how to **encrypt** and **decrypt** information as to keep information confidential and as to prevent unauthorized persons from accessing the information in transit. Generally, a crypto wallet is to keep the cryptocurrencies safe and secure. Every crypto wallet consists of a unique pair of **public and private keys**. The asymmetric nature of cryptography, where a pair of keys, namely a **private** key and a **public** key, are generated. The **public** key can be **derived** from the **private** key, but it's **impossible** to **reverse** this process to obtain the private key from the public key. This **one-way relationship** ensures that sensitive information encrypted with the **public key** can only be **decrypted** with the **corresponding private key,** providing a secure means of communication. The public key can be freely shared with others, allowing them to **encrypt** messages or verify **digital signatures** created using the corresponding **private key**. This **asymmetric encryption** method, utilizing **key pairs**, is fundamental to modern cryptography and is widely used in **securing data transmission** and **authentication processes**.

Private key: A private key is a secret code that is used to **access** a cryptography **wallet** and authorized transactions. Private key used to be a long string of alphanumeric characters that is mathematically related to the public key. A private key is **64** hexadecimal characters.
Note: If someone gains access to your private key, they can steal your cryptocurrency funds.

Public key: The public key is the code, and can be given to others to send you tokens. The public key of a crypto wallet is **derived** from the corresponding **private key** using mathematical Elliptic Curve Cryptography (ECC). It is a cryptography code used to **encrypt messages** and **verify digital signatures**. In hexadecimal, 2 digits represent a byte, meaning addresses contain 40 hexadecimal digits. Example: **0x**b794f5ea0ba39494ce839613fffba74279579268. Total: **42**
Note: The public key is not the same as a wallet address. A public key is part of the wallet address is used to identify a destination for cryptocurrency transactions.

Cryptocurrencies are digital assets that are linked to **particular blockchain networks**. These are the **cryptocurrencies** that make up the **“block rewards”** given out to **nodes** for **adding/verifying** or **validating** new blocks on the chain. With these native cryptocurrencies that represent **digital value**, **blockchain networks** are able to use consensus mechanisms to facilitate network operations like **transferring assets** or **adding** and **validating new blocks**. For example, **sending bitcoin** from one person to another will **incur a transaction fee** (sometimes called a “**gas fee**”) for using **network resources** like **electricity and computing power**. Those **transactions**, batched into **blocks**, are added to the **shared ledger** by **network nodes**.

Crypto wallets: store assets and connect to Web3

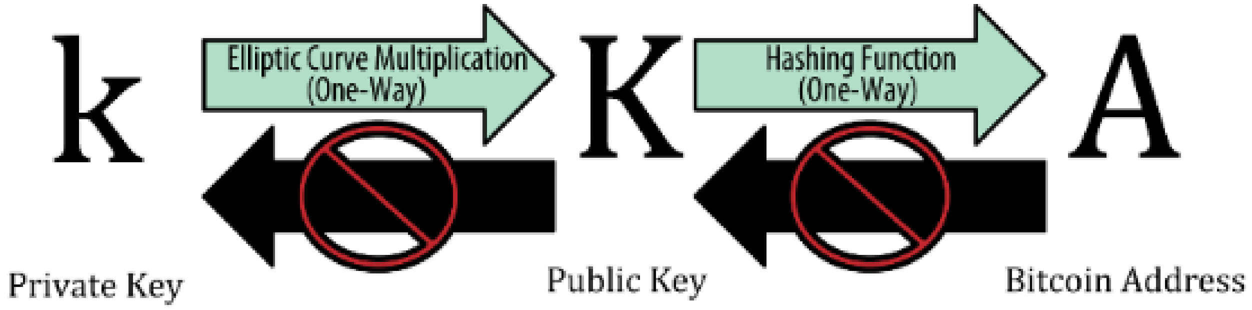
With all this crypto being **exchanged** to facilitate the **operation** of decentralized networks, people need a way to store their assets. **Crypto wallets** are designed to do just that. A crypto wallet is a way for **Web3 users** to **store crypto**, **transfer** it to others, **pay** transaction fees, and more. **Web3** relies on **blockchain networks**, **blockchains rely** on **cryptocurrency** to **facilitate operations**, and cryptocurrency needs **crypto wallets** to be **stored in**, **sent from**, and transacted with. Crypto wallets are like your **passport** to Web3. The only difference is that, unlike a travel passport, **crypto wallets don’t** have any **central authority**, like a government managing them.

How do crypto wallets enable you to sign on to an app or website on web3

Crypto wallets use **private keys** to **access public “addresses”** that can replace traditional login credentials. A wallet address is a unique identifier that is used to receive or send cryptocurrencies and can be linked to a bank account. Besides, the address can be shared with another person and is used to receive transfers of digital assets. The wallet address is mathematically derived from the wallet’s public key through a one-way function called “Hashing”. For example, a typical **blockchain public address** might look something like this:

0x634790328Ab021cA1E9Cf80457E8f8eFc5E8bA79

That address is a **unique wallet identifier**. Think of it like a **username**. Now, **when visiting** an app or website **on Web3**, you’ll be asked to **connect your wallet**. To do so, both **wallet address** and the **private key** will be needed (like a password) **to authorize** the connection.


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

Role of each of these **3 concepts** for a wallet user

- Create a **private key** when creating a crypto wallet. **Never do anything** when it is consciously, but it is used to **sign the transaction** when **sending crypto assets**.
- Create a **secret phase** and store it safely on a piece of paper. Use it if ever you have to **restore** the crypto wallet funds after **losing** the **private key**.
- **Public key** is used to **verify** and prove the wonder of a **wallet address** and that as receiving **crypto assets**. Personally, a public key is **not** to be used when **making** or receiving a **transaction**.

- If **receiving** any transaction or used for sending money from another wallet of the same cryptography, we need to tell the **wallet address** to the **sender**. Likewise, the **wallet address** of a **recipient** is needed if **senders** want to **send** crypto assets to the **recipients**.

The Need for Web3

True digital ownership, and the need to **reduce** the power of **centralized internet authorities**. By leveraging blockchain technology, Web3 promises a more democratic, **secure**, and **transparent internet** where users have **greater control** and **autonomy**.

Web2	Web3
Dominated by **centralized services** like Google, FB, Amazon, where user data is **controlled** by these **entities**	**Decentralized networks**, offering a platform where data is **distributed access** a network, **mitigating** the **risks of centralized control** and **data breaches** from **failures**.
Data ownership and **privacy**. Surrender their data to the service provider.	Emphasize **user sovereignty** (allows users to **identify** themselves on the Web using **credentials** stored in a **digital wallet** on their **smartphone**) over **personal data**. Ensure that individuals **maintain ownership** and **control over** their **own data**.
Many web2 applications	Characterized by **open protocols** (not owned by anyone/public) and **standards**, enabling **different services** and applications to **interact** more. Foster a more **interconnected** and **interoperable** web ecosystem.
Not possible for **tokenization**	Incorporates **concepts** like cryptocurrencies and **non-fungible token (NFTs)** allowing for the creation, **ownership** and **trading of digital assets**. The **tokenization** introduces **new economic models** and **opportunities** for users to directly **benefit** from the **value** they **create online.**
Usernames and **passwords** grant access to one app	**Wallet address** and **private key** grant access to any app or website on Web3. Wallet addresses may help to keep pseudonymous (fictitious names/false names) to those supporting services.

Extra

How to Get a Private Key from a Wallet Address?

To get a **private key** from a **wallet address**, the **owner** will need to **access** the wallet software or service that was used to create the address. The specific steps may vary depending on the service or software used.

Is a Public Key the Same as a Wallet Address?

No. A public key and a wallet address are not the same, but they have in common. Both are related to cryptocurrency transactions. Furthermore, unlike private key and secret phrase, **public key** and **wallet address** are **shareable**.

The main **difference** between them is that a **public key** is used to **encrypt** and **verify** transactions, and a **wallet address** is used to **identify** the **destination** or **source** of a **transaction**, and it can also be shared with anyone. In short, the **wallet address** is a **door** and the **public key** is the **label** on it, which shows who is the owner of the house.

Is a Private Key Same as a Wallet Address?

A private key is not the same as a wallet address. A private key is used to **sign transactions** and prove ownership of a particular wallet address. Owner can share the wallet address with others but should never even reveal the private key to anyone.

What is an Etherum Address?

An **Ethereum address** is a unique identifier used to **send** and **receive** Ether and other Ethereum-based tokens. It is a **string** of **42** characters starting with “0x”.

0xb794f5ea0ba39494ce839613fffba74279579268

Is an Etheruem Address the same as a Public or Private Key?

An Ethereum address is not the same as a public or private key, but is derived from a public key using a specific algorithm. The public key is, in turn, derived from the private key.

Evolution of the web (Web1 to Web3)

Web1, the **first iteration** of the **internet**, was primarily a collection of **static websites**, serving as a **digital information repository**. Then came **Web2**, the internet as we largely know it today, characterized by **interactive experiences**, **social media**, and **user-generated content (UGC)**. However, **Web2 centralized** the **control of data** and **platforms** in the hands of a **few major companies**, leading to **concerns over privacy**, **data ownership**, and **monopolistic practices**. Lastly, Web3 allows high engagement of **user control**, **decentralization** and **ownership.**

Web3 vs Metaverse

Web3 is a concept for a **decentralize**d version of the **WWW**
The **metaverse** refers to virtual worlds that enable online social interaction using **digital avatars**.

The term "metaverse" refers to virtual worlds facilitating **online social interaction** through digital avatars, often employing virtual or augmented reality technology. Eg. Second Life, Minecraft, and Roblox, with Pokemon Go showcasing AR in a metaverse context. While projects like Decentraland and Sandbox leverage blockchain and **non-fungible tokens,** many metaverse initiatives still rely on web2 technology.

	Web3	**Metaverse**
Application	● Encompassess **decentralized finance** (**DeFi**) ● Enable **seamless globa**l **payments** and **enhanced financial security** ● Extends **beyond finance** to **supply chain visibility** services ● Rely on **blockchain**	● Primarily used for gaming and **social Interaction** ● Not inherently focused on financial applications ● Mainly on gaming and social media, offers a new way for virtual co-workers to collaborate ● May incorporate with **VR or AR**
Scalability	Blockchain **trilemma (3 pillars)** 1. **Security** refers to the **robustness** prevent any malicious entities from taking over 2. **Scalability** refers to the **demands** that blockchain accommodate a **large number of transactions** 3. **Decentralization** where network is equally **distributed** among all participants instead of being concentrated in a single entity	● Encounter **scalability** due to the **technology limits** ● Required **advancements** in **network infrastructure**

User Experience	● Inspire **trust**, **transparency**, **partial ownership** and enhance **security** ● Diverse range of **applications** and **services** ● Core elements for an **ideal web3** user experience	● Incorporate **mixed VR** ● Focuses on **immersive digital world experiences** ● Reduce **mad actors** and enhance **overall user experience** ● VR tech

References

"The Infinite Machine" by **Camila Russo** is a book that chronicles the development and journey of Ethereum, a groundbreaking platform in the **Web3** space. Published in 2020, the book provides an **in-depth exploration** of the creation, **challenges**, and **impact of Ethereum**. **"The Infinite Machine"** delves into the founding **vision of Ethereum** and its creator, Vitalik Buterin. It explores Buterin's motivations and the **conceptualization of Ethereum** as a platform for **decentralized applications (dApps)** and **smart contracts.** However, one of the significant events covered is "The **DAO"** (**Decentralized Autonomous Organization**) incident in 2016, where a **vulnerability** in a **smart contract** led to a **major hack**. The **aftermath** of this event led to a controversial **hard fork**, resulting in the creation of **Ethereum** and Ethereum Classic.

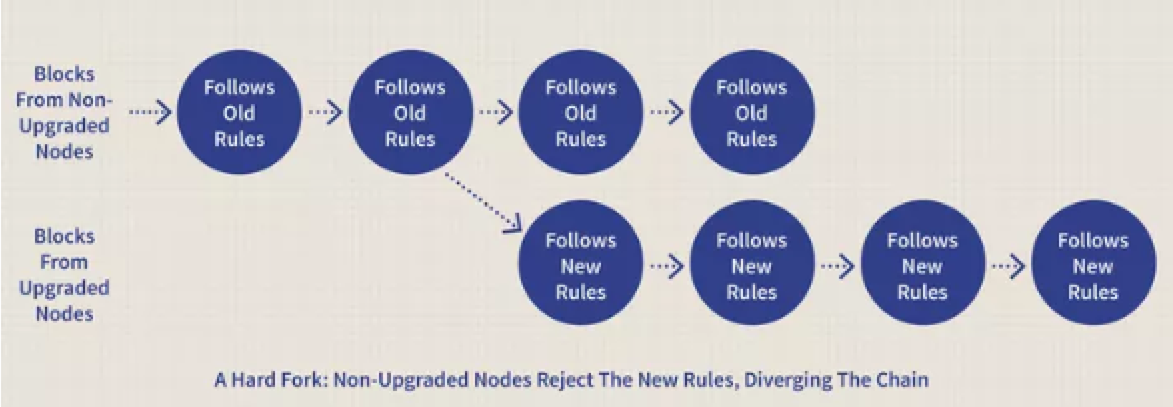
The book provides **insights** into how **Ethereum** has influenced the broader **blockchain space** and contributed to the development of the **decentralized internet**, often referred to as **Web3**. It explores Ethereum's role in enabling new possibilities **beyond cryptocurrencies**, including **decentralized finance (DeFi)** and **non-fungible tokens (NFTs)**. Therefore, the book discusses **Ethereum's global impact** and the **challenges** it faces, such as **scalability issues** and the **transition** to **Ethereum 2.0**. Overall, **"The Infinite Machine"** explores the broader implications of this groundbreaking blockchain platform in the **evolution** of the **decentralized internet** from the **historical** account of **Ethereum's development**.

Hard Fork

A hard fork is a significant change to a blockchain's **rules,** making previously **invalid transactions** valid or creating two **separate chains**. It happens when nodes of the latest blockchain version **no longer** accept the **older version**, causing a **permanent split**. For a hard fork to work, all users must **upgrade** to the **new protocol**. This results in **two chains**, one following the **old rules** and one with the **new changes**. **Token holders** on the **original chain** get **tokens** on the **new chain**, but **miners** must decide which chain to support. Therefore, adding **new rules** to the codes to **create a fork** in the blockchain: one path **follows** the **new** upgraded blockchain, and the other **continues** along the **old path**. After some time, those on the **old chain** will realize their version is **outdated** or irrelevant and **quickly upgrade** to the latest version.

How Forks Work

A blockchain fork, indicating a **change in protocol** or rules, is not exclusive to Bitcoin but can occur on any crypto platform. The blockchain operates similarly across platforms, with blocks acting as cryptographics **keys moving** memory. **Miners,** who set the **rules** governing (to control/rule) the **networ**k, must collectively **agree** on new **rules** and **valid blocks**, prompting a "fork" to signify a **protocol change**. This process led to the **creation** of **various digital currencies** such as **bitcoin cash** and **bitcoin gold**.



Developers may implement a hard fork for reasons such as addressing **security risks**, adding functionality, or, as in the case of Ethereum, **reversing transactions** like the **hack** on the DAO. Etheruem underwent a hard fork, backed by a near-unanimous (near **two people agreement**) community **vote**, to **roll back** transactions and return funds to DAO token holders. The hard fork did not reverse the network's transaction history but relocated DAO funds to a new smart contract, enabling original owners to withdraw their Ether at a rate of approximately 1 ETH to 100 DAO. The remaining balance was distributed for "failsafe protection" by DAO curators (ppl for collecting artwork, achieving smtg).

Hard forks and **soft forks** share the concept of **altering** a cryptocurrency platform's code while **keeping** the **old version intact**. However, the key distinction lies in the impact on the blockchain.

In a **soft fork**, only **one blockchain** remains **valid** as users adopt the **update**. Conversely, a **hard fork** results in both the **old and new** blockchains **coexisting**, requiring software **updates** to align with the **new rules**. While **both forks** create a **split**, a **hard fork** leads to **two blockchains**, while a **soft fork** aims for a **unified blockchain**.

Considering security differences, users and developers often **prefer hard forks**, even when a soft fork could suffice. Despite the **computational power** needed for blockchain **overhauls**, the privacy **benefits** of a **hard fork** often **outweigh** those of a soft fork.

Digital assets NFTs

Many blockchain qualities of non-fungible tokens (NFTs) make them valuable and compatible with Web3. As unique blockchain tokens, NFTs enable you to provide **seamless ownership evidence** for digital art, music, data, in-game assets, personal records, and more.

Benefits:

- Creators can generate **unique digital products** that can be **sold** as **collectibles**, allowing them to get **direct revenue** from their **work**.
- Provide **digital assets** with a new dimension of **ownership** and **management**. With Web3, people have **complete ownership** over their **data and assets**, and NFTs make it **feasible** for individuals to **own** and **control** their **digital assets.**
- Facilitate the **development** of new **revenue models** (**revenue earning plan**) for digital content creators. Eg. they can generate additional **revenue streams**, such as royalties on secondary **sales** of their work.
- New degree of **security** and **transparency** to the digital world. While non-fungible tokens are maintained on a blockchain, they are **irreversible** and cannot be **altered**, preserving the asset’s **legitimacy (belief that a rule)** and **possession (holding on one’s own)**.
- Certain social media networks now offer NFT verification systems that enable you to utilize a **crypto wallet** to **certify** NFT **ownership** and **display** it as your profile picture (PFP).
- Provide users with control over their **digital identity**, **membership**, and **voting privileges**.

The future of NFTs in Web3

As an increasing number of creators and users adopt Web3, the **demand** for NFTs will continue to increase. NFTs will emerge as the primary method for **monetizing digital assets**, and new business models/revenue models will develop around them. Moreover, NFTs will permit new types of **digital ownership** and **governance**. With NFTs, users can possess and control their **digital assets** in a manner not previously imaginable. Hence, new forms of **online collaboration** and **ownership structures** will **emerge**. As NFTs become more prevalent, there will be an increase in innovation surrounding them. New use cases, like NFT-based **gaming products**, **virtual real estate**, and even identity verification, will arise.

Conclusion

Web3 and NFTs alter everyday interactions with the internet and **digital assets**. NFTs offer a new method for **monetizing digital assets**, enabling new forms of **ownership** and **control**, and providing **transparency** and **security** in the **digital world**. As more creators and consumers adopt Web3, the demand for NFTs will increase, and new business models and use cases will arise to support them.

Decentralized Autonomous Organization (DAO)

Decentralize management processes by using **smart contracts** and giving each member of an organization a **vote** in how it's run. All votes are **recorded** on the blockchain, ensuring **transparency**. DAO replaces authority with **smart contracts** on the blockchain. Decision-making shifts from managers to members who vote on proposed smart contracts. Any DAO **member** can propose and write **smart contracts**. The contracts may involve rule changes or suggestions on fund usage. Proposed smart contracts are **visible** to all members on the blockchain. Therefore, a **transparent** voting process where members decide on the contract's execution. DAO executes the smart contract only if there is consensus among members. DAO mechanics involve advanced cryptography and computer coding.

Components of a DAO

There are **three** basic components that are critical to DAO when its based on a blockchain
Smart contacts: self-executing contracts encoded on blockchain. SC are similar to standard contractual agreements that you sign on paper, but when encoded on blockchain, they are **impossible** to **break** without the consensus of the parties involved. Smart contracts **automate** and streamline decision making from **eliminating middlemen**. In a DAO, the smart contracts contain the **rules** and describe operating conditions such as **decision-making processes**, **fund allocation**, and **member interactions.**

Tokens: digital assets used to represent **ownership** and members in DAO, similar to owning **shares** of a company. In the context of a DAO, **tokens** are in the form of **cryptocurrency** like Ethereum. Tokens can be **traded** on cryptocurrency **exchanges** and held in **Crypto Wallets**. By **owning tokens** of a particular DAO, **members** can **participate** in the management process and **receive** a **share** of the **DAO's profits**. Owning **one** is enough to **participate** in the DAO, but **owning more** will increase the **decision-making power**. Depending on the terms of the smart contract, a member with 200 tokens may have twice the voting power as a member with 100 tokens.

Decentralization (No centralized legal entity): No human organization involved in the management of DAO. DAO can operate based on the contents of the smart contracts, which all **members** vote on. That makes a DAO more **democratic** and less likely to be corrupted than a centralized organization. After the **smart contract** is **complete** and **funding secured**, the DAO is **deployed** on the blockchain. Stakeholders then can participate in DAO, **vote** on proposals, and **make decisions.**

Advantages and Disadvantages

a. **Advantages**

   1. Decentralization: All members of DAO vote on all decisions, and the smart contract executes the decisions automatically. As a DAO is transparent, all members likely share a common interest in its success.

   2. Participation: Anyone whos owns a token in the DAO can participate in decision making power. This motivates each member to engage in the community. The more members who participate, the more successful the DAO is, especially if the DAO's aim is community growth. The decentralized structure could allow token holders to propose and vote on new features.

   3. Transparency: Every member can see what everyone else is doing, the system's integrity remains intact. All actions in the DAO including voting are considered public. This ensures that all members are held accountable. Bad actors are unlikely to crop up because their actions are instantly visible to the rest of the members. Transparency builds trust in the DAO and ensures that the organization continues to operate in the best interest of its participants.

   4. Community: DAO brings together people who share common interests and goal. This builds a sense of community, a social network, A shared connection and a guarantee of equitable results encourage members to engage in DAO and increase its chance of success.

b. Disadvantages

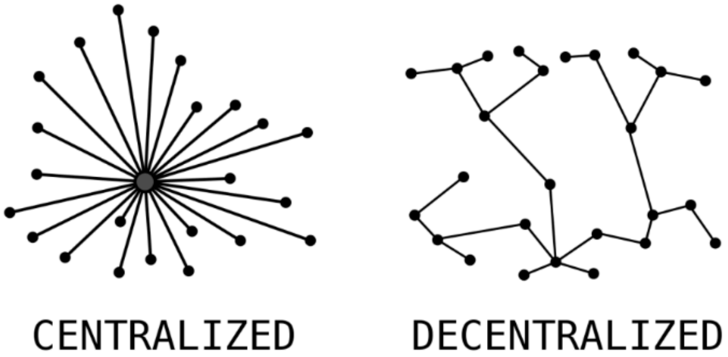
   1. Speed: DAOs require input from all individuals in the organization, which can cause longer voting periods and slower decision-making. A DAO may take longer to implement new feature or change due to the need for extensive voting and consensus-building among token holders. Sometimes, decision making could be slower without a strong authority.

   2. Education:DAO allow all individuals to say, yet somes not have proper background knowledge on a topic. With varying degrees of education and

experience,, a DAO may suffer from poor decision-making due to a lack of expertise. The organization could split if the community disagrees on issues due to varying education levels.

Key concepts: blocks, transactions, decentralized networks

Decentralization
Unlike traditional systems where a central authority controls the data, the **ledger is distributed** across a network of nodes. The process of decentralization ensures that **no single entity** has absolute authority/**control**, reducing risks of **censorship**, **fraud** and **downtime**. Decentralization democratized data management, allowing a strong user control and contributing to a more sure and resilient system. Decentralization allows for trustless transactions and **eliminates** the need for **third parties** to **verify** transactions. Businesses that may not have the resources to process transactions on their own gain the most benefits. In short, decentralization allows for **greater freedom** and **transparency** in transactions, as well as improved security and efficiency.



Blockchain ledgers are decentralized, with **nodes hosting** the blockchain data distributed **across** multiple **locations**. This **data distribution** across many nodes means that a compromised ledger on one node cannot compromise the entire network. (1 bad node cannot destroy all nodes). Decentralization is a significant advantage of blockchain over centralized that could result in the loss of all data stored in a single location.

Immutability
Once data is recorded in a block, it cannot be altered retroactively without altering all subsequent blocks through the process of cryptographic hashing, this means that all data and transactions on the blockchain are **permanent** and **unchangeable**. Every **block** contains the **hash** of the **previous block**, creating an unbreakable chain. **Altering** any information would need **immense computational power** to modify all **subsequent** blocks, which is practically impossible/unfeasible in a large network. This **immutability** contributes to a **trustworthy** and **tamper-proof**/secure record, which is vital for applications such as financial transactions, legal contracts and identity verification as well as reduced costs and risks.

With **blockchain ledgers**, modifying stored data is nearly impossible, as all the copies of the data on all computers (nodes) across the **globe** must be modified, Wherever data is stored in blockchain ledger, all nodes are **updated** in real time, and these pieces of data are stored in such a way that cannot be modified including **backdated**, deleted or erased. For instance if the **new action** is to **reverse** a **previous transaction**, a **new block** of data will be **created** that

references the **previous** record. The new record will initiate a new transaction, and all nodes must **vote** on its **validity** before it can be **added** to the ledger. Otherwise, other nodes disagree on the transaction's validity, it will be rejected, and the record will not longer be added to the ledger.

Transparency
Blockchain networks, especially public one, offer unparalleled transparency. Every transaction on the blockchain is visible to **anyone** who **accesses** it. For example, with blockchain, we can upload the product's data onto the network and let the customers see this information directly. This transparency ensures all network participants can **verify** and **audit transactions** independently. Transparency is able to build trust and accountability, as every acton is **traceable** and **irreversible**.

Security
Enhanced security by providing **transparency** and **traceability** for all transactions. With every transaction visible on a **public ledger**, parties involved gain **assurance** in the **trustworthiness** of their actions. Each transaction creates a new block on the chain, accessible to every node in the network, and is **verified** by **miners** to ensure it meets **specific conditions**. Once **verified**, the transaction is **added** to the public record, visible to all parties to access to the blockchain. This **verification** process, conducted by **authorized miners**, safeguards transactions from tampering or falsification, ensuring the network's security and integrity.

Scalability
Scalability is the capacity of a blockchain system to accommodate an **increasing demand** volume of transactions without **compromising**. Three main factors that affect the factors are **execution, storage and consensus.**

a. Execution: refers to the **speed** at which **transactions** are **processed** on the blockchain network. It is determined by factors such as **no. of nodes** involved in the **network**, **processing power** and **bandwidth** available.

b. Storage: Blockchain storage encompasses **two** main components; **historical data** and **global state**. Historical data includes all **raw transactions** and **block information**, while global state represents a **snapshot** of **data accessible** to **smart contracts**. Full nodes require access to both historical data and global state for synchronization and validation purposes. However, as the ledger and storage requirements grow, computation of state becomes slower and more resource-intensive. **State bloat** may occur when **storage** requirements **increase**, making it **challenging** for full **nodes** to stay **synchronized** and for **new** nodes to **join** the network. To address this issue, **scalability** solutions aim to enable blockchains to **process** and **validate** more data **without** significantly **increasing** storage requirements, thereby ensuring the integrity and efficiency of the network.

c. Consensus: Blockchain consensus is the **process** by which **nodes** in a decentralized network **agree** on the **current state** of the blockchain. It **aims** to **achieve** an honest majority and ensure finality, meaning **transactions** are processed **accurately** and unlikely to be reversed. Consensus designs focus on **minimizing communication** overhead to enhance decentralization and Byzantine fault tolerance, while also **reducing** settlement time. Scaling the consensus layer requires solutions to achieve faster, cheaper, and more trust-minimized finality in a predictable and stable manner.

Privacy

Privacy verifies the **transparency** of data and transactions on the network, since all the **information** is **accessible** to everyone; therefore it cannot be misrepresented. It also allows for greater accountability since everyone involved in a blockchain project is publicly visible. However, note that privacy is not guaranteed forever. One of the options is to use **strong cryptography** and ensure all the network nodes are reliable and honest. However, a **private blockchain** network may be more **beneficial** than a **public** one.

Flexibility

Transactions can be carried out swiftly and precisely, which is especially useful when planning to expand the system in future. Besides, by remaining streamlined and manageable, the overall complexity of the **network** remains at a **minimum level**, allowing for its growth without making any additional demands on users or developers.

Components of Blockchain

1. **Blocks**: A blockchain is a series of individual blocks linked in a chronological chain. Each block contains a collection of transactions. Every block has a unique identifier called a hash and contains the hash of the previous block, creating a linked chain.
2. **Transactions**: Transactions are the fundamental units of blockchain. They represent the data being exchanged in the blockchain network, such as **cryptocurrency transfers**, **contract executions**, or **information exchanges**.
3. **Nodes**: Nodes are individual computers that collectively maintain and update the blockchain. Each node has a copy of the entire blockchain, contributing to the network's resilience and security by processing transactions and validating new blocks. These nodes are unique identifiers. The more nodes a network has, the mode decentralization and secure it becomes.
    a. Crypto Nodes distribute the signed transaction:
    When signing a transaction, the details are sent to a set of nodes. The first set of nodes passed it onto the other nodes, who then passed it onto the next level of nodes. They do this until the transaction is included in a block or discarded.
    b. Nodes then verify transactions in the **Mempool** (a cryptocurrency node's mechanism for storing information on unconfirmed transactions):
    As the transaction is **distributed**, it enters a mempool in each node. Initially, it has a **queued status**. But from there, the nodes must **validate** the transaction. Once the majority of nodes **validate** that the transaction is **valid**, it moves to **"Pending"** status. This indicates the **transaction** is **ready** to be added to the chain. If the **majority** of nodes decide a transaction is **not valid**, it will be **discarded**.
    c. Nodes Add Transactions to Blocks and Broadcast them to the network:
    Once the **transaction** moves to a **pending** status, **miner** or **validator** nodes are able to **add** the block to the **network**. At the point a miner or validator **wins** the block and adds to the chain, the transaction becomes **immutable**. To explain, making any **alteration** to the transaction would require **majority** of the nodes' **approvals**, which could be **thousands** of **nodes** in more popular blockchains.

Nodes have Incentives (and **Deterrents** (discourage)) to ensure good behaviour
At this point, it's important to note that some nodes are **responsible** for **adding** blocks to the **network**, which will usually **earn** them **cryptocurrency rewards**.

On a **proof-of-work** blockchain like Bitcoin, this requires a **large** amount of **computational power** in order to **solve** a **complex** cryptography problem. Means that **miners** have an **incentive** to **add** valid **blocks** to the chain, but they also have **discouraged** from behaving dishonorably, by simply the **energy cost** of **mining** wouldn't be **profitable** without the **block rewards**.

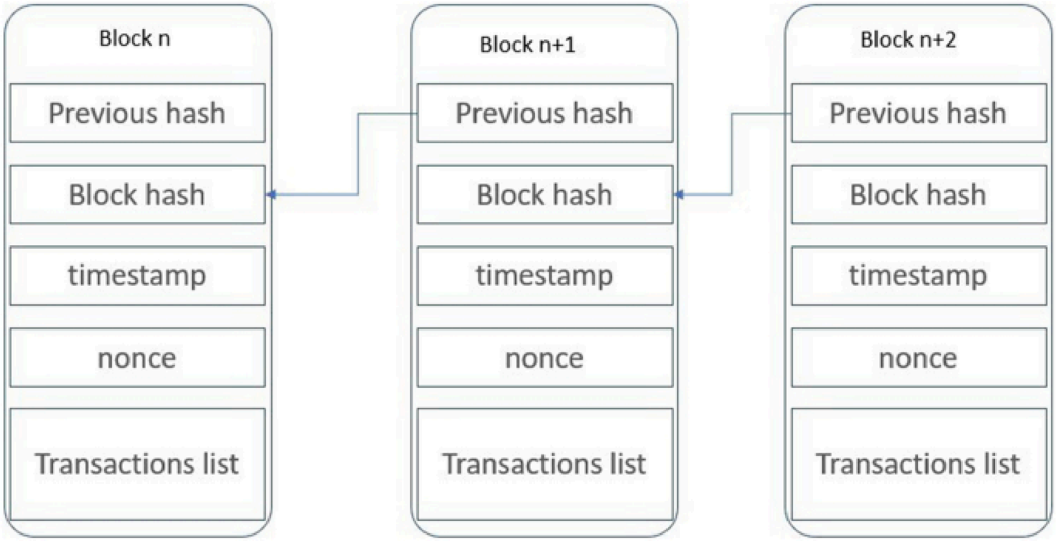
On a **proof-of-stak**e blockchain, participating **nodes** also receive **block rewards**, but as a **deterrent** for **bad behaviour**, these chains have a different method. To explain, **validators** must **lock** up a **significant** amount of their **funds** as **collateral** (an item of value pledged to secure a loan 抵押品), also aka a **stake**. This **mechanism** ensures that nodes **behave** as they should, even **without** an **ongoing energy cost** and **expensive equipment.**

4. **Nonce**: A nonce is a **unique number miners** solve for to **add blocks** to the blockchain, **receiving rewards** upon **success**. It functions like a **one-time password** in a proof of work system, requiring high computational power. Despite its simplicity, nonce is a vital security feature in blockchain.
5. **Hashing**: Hashing employs a **hash function** to convert **texts or numbers** into fixed-length values, safeguarding message content during transmission, particularly in blockchain transactions. Cryptocurrency transactions use hashing algorithms like SHA-256 in Bitcoin, ensuring a consistent 256-bit output regardless of input length. Hashing detects even minor file alterations, enhancing security in data transfer.
6. **Cryptography**: Cryptography employs **algorithms** to **protect data**, ensuring only **intended recipients** can **access** it by rendering messages unreadable. Despite operating on public blockchains and peer-to-peer networks, blockchains remain secure due to cryptography and other technologies, making them one of the 21st century's most robust inventions.

Blockchain Concept

1. Transaction are **broadcast** to the network
   P2P network enables participants to interact with each other and conduct transactions without intermediaries or centralized authorities. **Structure networks** have organized data structure making **data access** more **efficient**, but in terms of **complicated setup process**. In contrast, **unstructured networks** are more **flexible**, allowing participants to **join and leave** the network as **desired**, but not as efficient as structured networks. **Hybrid networks** combine **P2P** and **traditional client-server models** with a **central server** locating **nodes**.
2. Transaction are **collected** in blocks
   The **header** contains **metadata** such as **timestamp** which has a random **number** used in the **mining process** and the **previous block's hash.** The **data section** contains the main/actual information like transactions and smart contracts which are stored in the block. The **hash** is a unique **cryptographic value** that works as a **representative** of the **entire** block which is used for **verification purposes.**

   a. Block Time: time taken to generate a new block in a blockchain. Different blockchains have different block times, which can be vary. Shorter block times provide faster transaction confirmations but the result has higher chances of conflicts but the longer the block times may increase the timing for transaction confirmations but reduce the chances of conflicts.



3. Blocks are chained together
4. All nodes maintain the **same ledger**
   A **node** is a **single computer** that can interact with and is part of, a blockchain network. Each node is a single and separate computer which **stores** all the **information** on the blockchain, also known as **distributed ledger**. Since every node has a copy of the **same information**, they can **verify each other.** This allow the nodes to **verify** and

record new transactions and **broadcast** them to the network without the help of a central entity.

5. Distributed ledger replaces databases
   One of blockchain's fundamental concepts is replacing centralized databases with a distributed ledger system. Unlike traditional databases, where data is stored in a single location controlled by a central authority, blockchain's distributed ledger replicates data across multiple nodes within a network. Each node maintains its own copy of the ledger, enhancing transparency, security, and resilience. This decentralized approach eliminates the need for intermediaries, as transactions are verified and recorded through a consensus mechanism agreed upon by network participants. The distributed ledger's fault tolerance ensures operational continuity even if some nodes fail, contributing to blockchain's decentralized and trustless nature.

Importance of Crypto Nodes

Crypto nodes play a vital role in ensuring the security and decentralization of blockchain systems. They enable fair transaction validation without the need for centralized entities, making collusion difficult. The increasing number of participants operating nodes enhances network security and decentralization. Whether running a full node, miner, or validator, nodes are essential for blockchain infrastructure, making understanding them crucial for network participants.

1. Full node: A full node stores a **full copy** of the network's digital ledger. These types of nodes build the foundations for most blockchain; storing the **history** of the chain and communicating with other full nodes. Full nodes are the **backbone** of a network.

2. Miner Nodes: Miner nodes are responsible for **verifying** transactions and **adding** them to the **blockchain** on **proof-of-work** blockchain. Mining requires a lot of **computational powe**r to solve the **complex** computation puzzles. However, they also **receive cryptography rewards** in **return** of their work.

3. Validator Nodes: Validator nodes are similar to miner nodes, but on a **proof-of-stake network** instead. They also **validate transactions** and **create blocks**. However, they **don't** have to **solve** complex computational problems, they are **chosen** in relation to the **amount of funds** they **lock** in the system. Like **miners**, they also receive **rewards** for **creating blocks**.
4. Light Nodes: **SPV clients**, also aka **light nodes**. Firstly, these types of nodes do **not store** the entire blockchain like their **heavyweight counterparts**. Instead , they just **download 'block headers'**. Means they **don't** require as much **storage capacity**. Instead, these crypto nodes' only task is to **verify transaction**s in the blockchain using **simplified payment verification (SPV)**. SPV is a form of **lightweight** client that **confirms** blockchain transactions. In short, a light client is a type of **software** that **interacts** with the **blockchain** without having to run constantly, nor do they read and write mass amounts of data to the blockchain.

   Miners: In many blockchains, miners are special nodes that **validate** new transactions and **add** them to the blockchain. They use **computational power** to solve complex cryptographic puzzles, a process known as **minin**g, especially in **Proof of Work (PoW)** systems.

   ### Miners vs Validators

   A **miner** is a dedicated computer system that can **add** new blocks of transaction to the blockchain. To **mine new coins** or **validate transactions**, this miner must **solve** the **complex mathematical** computations, which requires a considerable amount of energy. This crypto mining mechanic is what keeps a **proof-of-work** blockchain secure. And also, **bitcoin mining** is too **costly** as well to consider as trying to cheat the system. Each **miner** is a **node**, **but not every** node is a miner. For example, anyone can run a

crypto node to help the Bitcoin **consensus** run **without mining** a single coin. Plus, beyond those distinctions, there are also differences across different type of networks too.

On **proof-of-stake** networks, nodes are operated by **Validators** instead of miners. But even on proof-of-stake networks, own node can be **set up** without validating the transactions. Thus, nodes and validators are the same either.

Blockchain vs traditional databases

Blockchain technology and traditional databases, while both used for storing and managing data, have fundamental differences in their architecture, operation, and overall purpose. Understanding these distinctions is crucial for recognizing the unique advantages and limitations of each system. Meanwhile, traditional db, such as relational databases have been the backbone of data storage for decades. They rely on centralized architecture, where data is stored in tables with predefined relationships.

1. Structure and Data Organization
    a. Blockchain: It is **structured** as a **chain** of blocks, with each block containing a **set** of transitions or **records**. These blocks are **linked** using **cryptographic** principles. Creating a **chronological** and **immutable sequence**.
    b. Traditional Databases: They are typically **structured** in **tables and rows**, following a model like **relational databases**. Data is **organized** based on specific schemas and can be **easily modified** or deleted by database administrators. This structure ensures **data integrity** and **consistency** and **reliability** due to their **structured nature** and central, but it can be **challenging** to **accommodate changes** or new data formats.
2. Decentralization vs. Centralization
    a. Blockchain: Inherently decentralized, blockchains distribute copies of ledger across multiple nodes (computers) in a network. This means **no single entity** has **complete** access/**contro**l over the entire database, intermediary such as bank or government entity that validates transactions. Hence, **eliminates** the need for **intermediaries**, reducing costs, and increasing the speed of transactions.
    b. Traditional Databases: they are **centralized**, **hosted on** specific **servers**, and controlled by a **central authority** or **administrator**. Meaning the data is stored and **managed** by a **single entity** or server. This central authority controls access, security and data integrity. This centralization can be a point of **vulnerability**, **exposing** the **databas**e to **risks** of **single points of failure** (one fails the entire system down) and **centralized control**.
3. Immutability
    a. Blockchain: Once data is recorded in a blockchain, it becomes almost **impossible** to **alter**. This immutability is a core feature, ensuring data integrity and trust in the records.
    b. Traditional Databases: Data **can** be modified, updated, or deleted by users with appropriate access rights. This flexibility is necessary for many business applications but can lead to **concerns** about **data tampering** or loss.
4. Transparency and Anonymity
    a. Blockchain: Public blockchains are transparent, allowing anyone to view the transaction history. However, the identities of the parties involved are often pseudonymous.

    b. Traditional databases: They usually offer **privacy** and **access controls**, restricting data visibility to **authorized users only**. User identities are typically known to the database administrators.

5. Consensus Mechanisms
    a. Blockchain: It relies on consensus mechanisms like Proof of Work or Proof of Stake to **validate transactions**. This process ensures that **all participants agree** on the data's validity **without** needing a **central authority**.
    b. Traditional databases: They rely on standard **CRUD** operations managed by **database administrators** or automated scripts without the need for network-wide consensus.
6. Security and Encryption
    a. Blockchain: Employs **advanced cryptographic** techniques to **secure data**. Each **block** in the chains contains a **unique identifier hash** of the previous block, creating a tamper-proof system. Data on the blockchain is encrypted, enhancing data security.
    b. Traditional database: Implement security measures like **access controls** and encryption, they are susceptible to **data breaches** and **hacking attempts**.
7. Smart Contracts
    a. Blockchain: Smart contracts are **self-executing agreements** with **predefined rules** written in code. They automatically **execute** when specific conditions are met. Smart contracts eliminate the need for intermediaries, reducing transaction costs and increasing efficiency.
8. Use Cases and Efficiency
    a. Blockchain: Best suited for scenarios where decentralized control, data immutability, and transparency are crucial, such as in cryptocurrency transactions, supply chain tracking and voting systems.
        i. **Cryptocurrencies and Finance**
        Advent of cryptocurrencies like Bitcoin. It revolutionized the financial industry by enabling secure and transparent transactions.
        ii. **Supply Chain management**
        Enhances supply chain management by providing real-time visibility, traceability and authenticity verification of products.
        iii. **Voting Systems**
        Ensure the integrity of voting processes and prevent fraud, ensuring democratic practices
    b. Traditional Databases: More efficient for high-speed transactions and data processing tasks in business environments where centralized control and data mutability are necessary. Scenarios such as Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Human Resources Management (HRM).
        i. Enterprise Resource Planning (**ERP)**
        Manage various aspects of a company such as inventory, finance, and human resources.

ii. Customer Relationship Management (**CRM)**
To store and manage customer-related data, helping business improve customer interactions

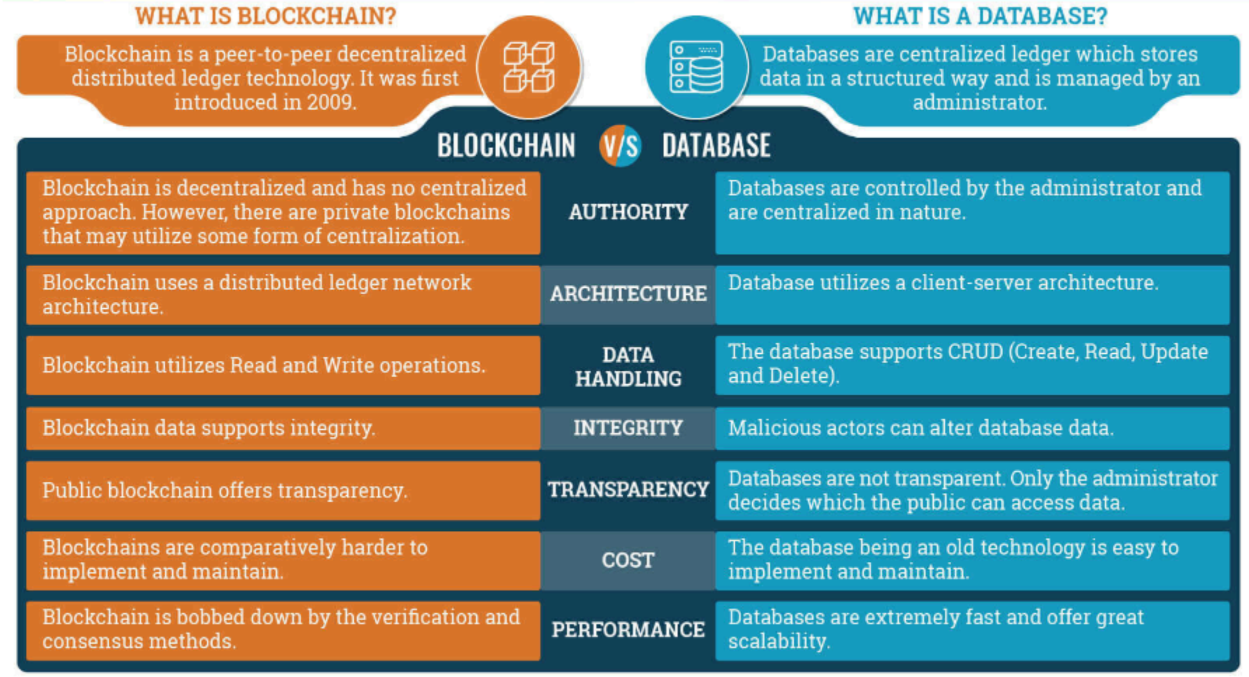
iii. Human Resources Management (**HRM)**
Facilitate human resource management by storing employee records, payroll data and performance evaluation

Limitation

Blockchain	Traditional DB
face scalability issues, resulting slower transaction processing times as the network gas.	Centralized are vulnerable to data breaches, with hackers targeting single points of failure
Energy consumption, due to Proof-of-Work (Pow) consensus mechanisms used by some blockchain demand substantial energy consumption, raising environmental concerns.	A failure in a centralized database system can disrupt entire operations, leading to downtime and data loss
Aspects include decentralized and pseudonymous nature of blockchain has raised concerns regarding compliance with regulations and laws	In multi-server environments, traditional db may encounter data synchronization challenges, leading to inconsistencies.

Conclusion:

Blockchain offers decentralization, immutability, and reduced intermediaries, while traditional databases provide consistency, reliability and better scalability for large-scale applications.



WHAT IS BLOCKCHAIN?

Blockchain is a peer-to-peer decentralized distributed ledger technology. It was first introduced in 2009.

WHAT IS A DATABASE?

Databases are centralized ledger which stores data in a structured way and is managed by an administrator.

BLOCKCHAIN V/S DATABASE

BLOCKCHAIN		DATABASE
Blockchain is decentralized and has no centralized approach. However, there are private blockchains that may utilize some form of centralization.	AUTHORITY	Databases are controlled by the administrator and are centralized in nature.
Blockchain uses a distributed ledger network architecture.	ARCHITECTURE	Database utilizes a client-server architecture.
Blockchain utilizes Read and Write operations.	DATA HANDLING	The database supports CRUD (Create, Read, Update and Delete).
Blockchain data supports integrity.	INTEGRITY	Malicious actors can alter database data.
Public blockchain offers transparency.	TRANSPARENCY	Databases are not transparent. Only the administrator decides which the public can access data.
Blockchains are comparatively harder to implement and maintain.	COST	The database being an old technology is easy to implement and maintain.
Blockchain is bobbed down by the verification and consensus methods.	PERFORMANCE	Databases are extremely fast and offer great scalability.

Blockchain	Databases
Decentralized Data Storage	Centralized Data Storage
Each Participant Holds a Secure Copy	Central Administrator Controls Data
Transparent Data Provenance	Limited Transparency
Immediate Identification of Inconsistencies	Manual Correction Required for Inconsistencies
Swift Correction of Unreliable Information	Potential for Delayed Correction
Removal of Manipulation Risks	Risk of Manipulation By Single Administrator
Decentralized Approach Enhances Trust	Centralized Control may destroy trust among Participants
Secure and Transparent Data Exchange	Limited Trust in Data Exchange, between competitors

To give an illustration, your friends; watch would immediately identify and correct any unreliable information. Your friend’s watch would immediately self-correct for daylight saving time, if a third person maliciously changed the time so they would be late, the time would immediately be verified against all participants and corrected.

More Difference between Blockchain and Traditional database

Blockchain	Traditional database
Enable **several parties** to share data **without** requiring a **central system** or administrator. As a result, the data is kept safe and secure.	The admin has **authority over** it from a central location. Any change in the data can cause the information to change all over the place. Anyone with access to the centralized databases can **corrupt** the **whole database**. It has resulted in hacking cases.
Faces **scalability issues** because of its **dependency** on all **decentralized nodes**.	Permissions are centralized and the powers to alter data are fews, traditional db can handle **enormous volumes** of transactions per second. The client-server design lowers reliance on nodes, which are replaced with isolated server centers.
Has the potential to **boost data transparency** and trust, resulting in increased creativity, productivity and quality	Traditional db **fail** to provide the level of transparency offered by Blockchain.
Cryptography is required to hide information on Blockchain. There is **no information confidentiality**.	When it comes to traditional databases, **information** is only **accessible to members.**
As blockchain is **public distributed ledger**, it can not be customized by an individual's choice	A traditional database offers numerous **customization** choices. **Permissions, privileges**, and **set-up requirements** can all be optimized as centralized management.
The user can **contribute** more data in the form of **additional blocks** in the Blockchain. The **old data** will **not** be **destroyed**; it will stay in the system and be **accessible** to the **public.** (Add and Read)	The client performs four roles in a typical database; **CRUD**.
The information that is **current** and the information that **was previously available**. It generates a database with its histories in it. This property of Blockchain makes it handy for **tracing records** and determining a **product's validity**	The information in a **centralized** or traditional database is current at **any given time**