MOBIUS

Tomorrow, Delivered.

The Platform Academy Blockchain Workshop

Fueling Growth with Training Solutions of Tomorrow.

GAIAN

Who am I?

Jose Anitt

Platform Architect, Tech Enthusiast

IIT Bombay, 2014 Grad
https://www.linkedin.com/in/gjanitt/

Building Mobius

What is Web3?

Definition: Web3 is a decentralized version of the internet where users have ownership and control over their data, powered by blockchain technology.

Evolution: Web1 (static content) → Web2 (dynamic, user-generated content) → Web3 (decentralized and trustless)

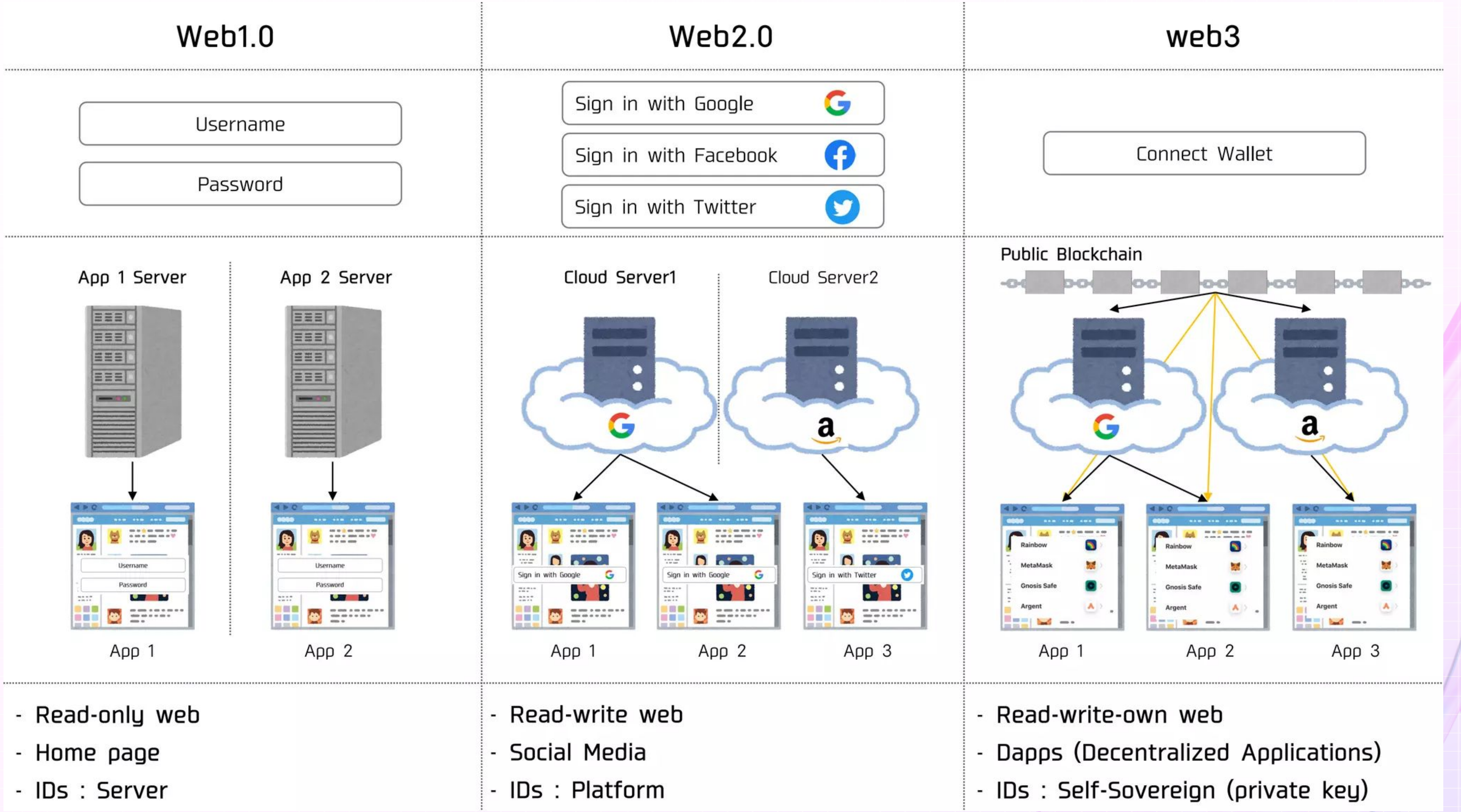
Key Differences:

- *Web2*: Centralized, platform-dependent, data owned by platforms.
- *Web3*: Decentralized, peer-to-peer, data owned by users.

Importance: Empowers users, enhances privacy, and reduces dependency on centralized entities.

The Platform Academy

Web3



The Platform Academy

Web 2.0 - How we are operating currently

Centralized data

Owned and controlled by organisations

Bigger organizations make profits

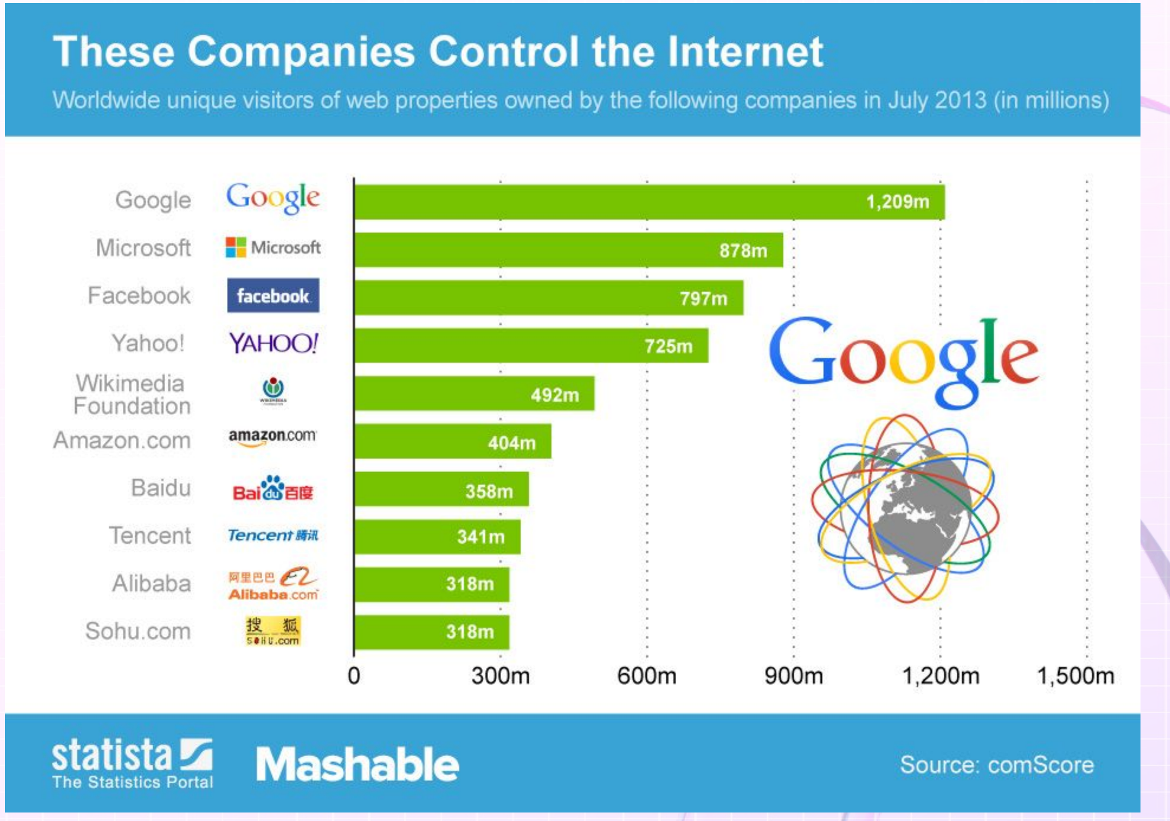
Monopoly

Owning user behavior data

Insecure

Inequality between millions of people and few in power

Web2.0 provided a fertile ground for corporations to monopolize control and profits



Reference: Blockchain Concepts - Murughan Palaniachari

The Platform Academy

Web 3

Secure and private

Decentralized

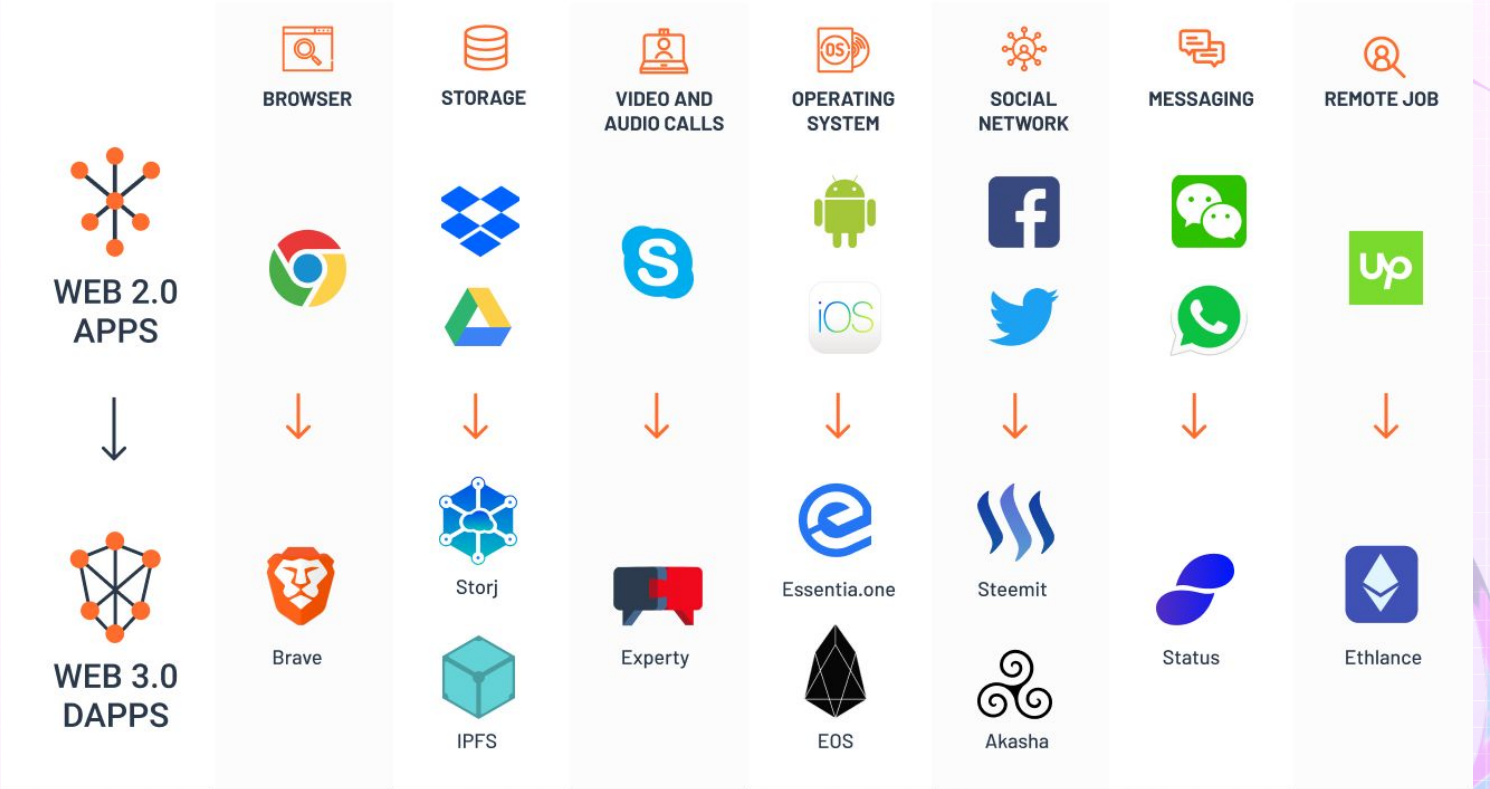
Individual Ownership

No central point control

Blockchain and Cryptocurrency

Permission-less

Uninterrupted service



Reference: Blockchain Concepts - Murughan Palaniachari

The Platform Academy

Decentralization

No Central Authority: Power and control are distributed across multiple participants, rather than centralized in a single entity.

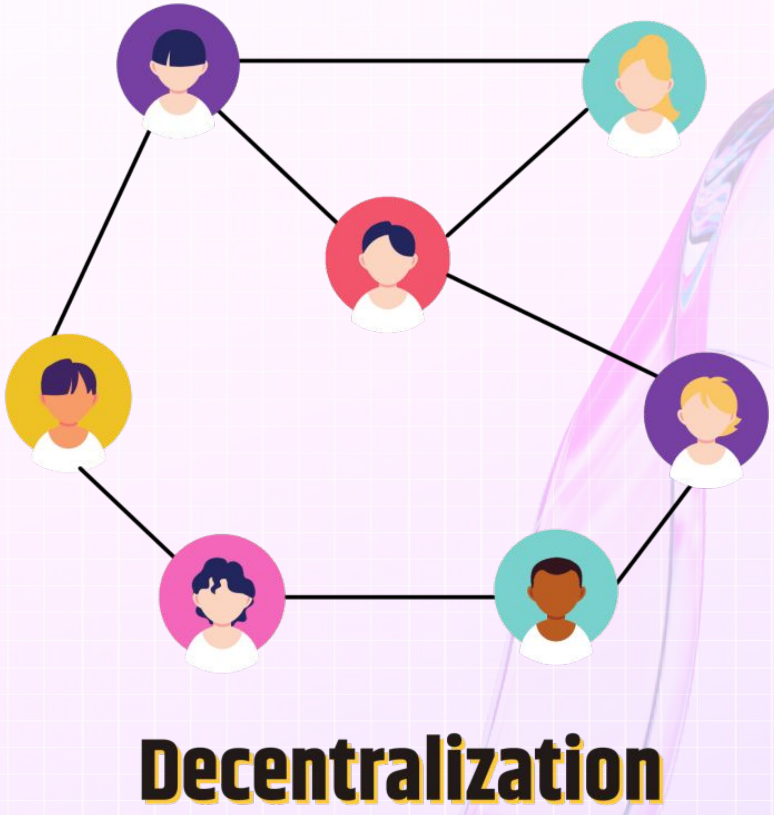
Increased Security: Reduces the risk of single points of failure and makes systems more resilient to attacks.

Enhanced Transparency: All participants have equal access to data, promoting openness.

Empowerment: Users have more control over their data and decisions, reducing reliance on intermediaries.

Examples:

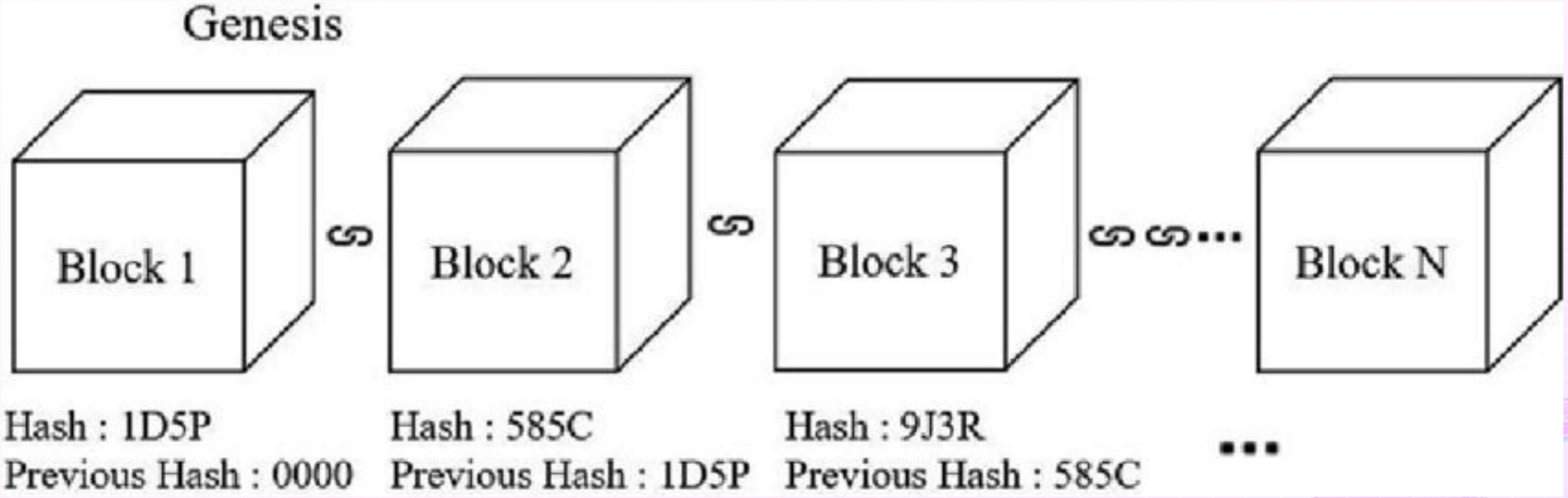
- **Blockchain**: Transactions are validated by a network of nodes, not a central authority.
- **Peer-to-Peer Networks**: Resources and data are shared directly between users.



Blockchain

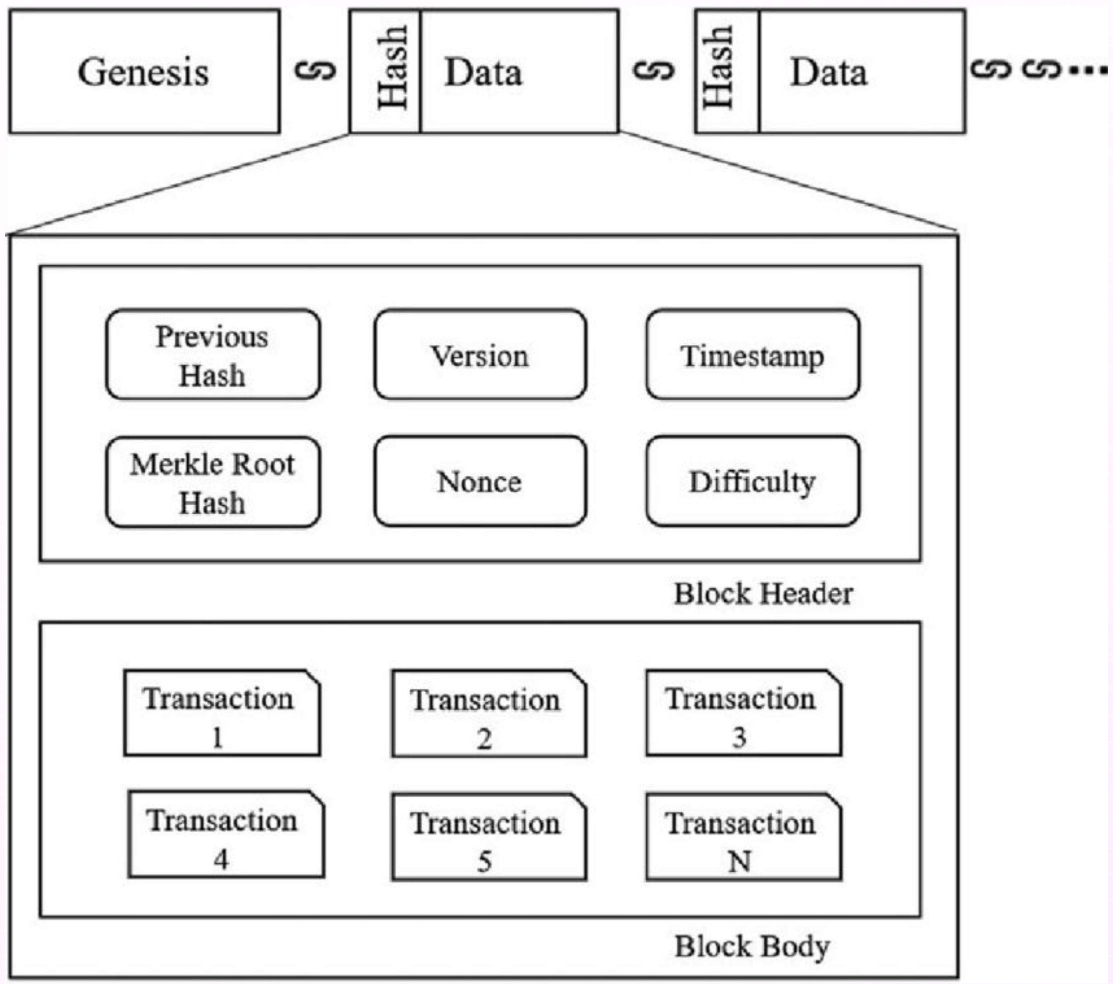
A blockchain is a growing list of records, called blocks, that are securely linked together using cryptographic hashes.

Each block contains a cryptographic hash of the previous block, a timestamp and transaction data.



The Platform Academy

Block Structure



Block Header:

- **Version**: Block version number.
- **Previous Block Hash**: Links to the previous block, maintaining the chain.
- **Merkle Root**: Represents all transactions in the block.
- **Timestamp**: Time the block was created.
- **Difficulty Target**: Sets mining difficulty.
- **Nonce**: Value used in mining to find a valid hash.

Transactions:

- List of all transactions included in the block.

Block Hash:

- Unique identifier for the block, generated by hashing the block header.

The Platform Academy

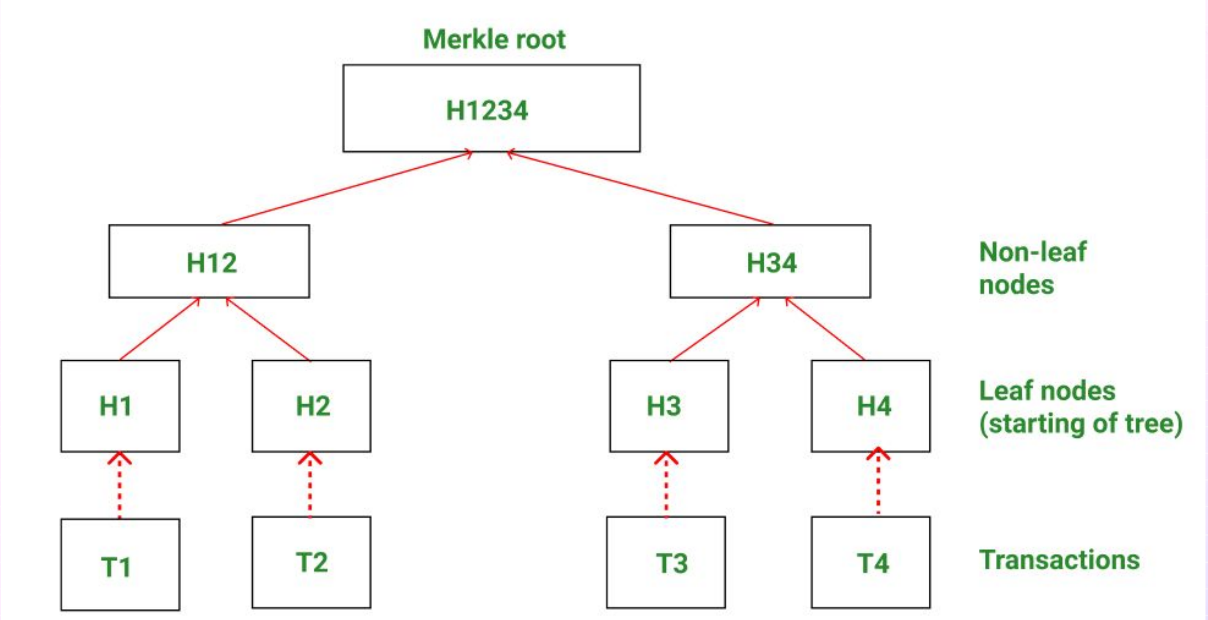
Merkle Tree

A cryptographic tree structure where each leaf node is a hash of a transaction, and the root node (Merkle Root) represents the entire block.

Purpose: Ensures data integrity and efficient verification of transactions in a block.

How It Works:

- Transactions are hashed.
- Hashes are combined to form parent nodes until a single Merkle Root is obtained.



Importance:

- Detects tampering with transaction data.
- Enables lightweight clients to verify transactions efficiently.

The Platform Academy

Mining Difficulty

hash power / hash rate
computational power of a specific cryptocurrency network
how many hashes are being generated by bitcoin miners - hashes per second

cryptocurrency difficulty = measure of how difficult it is to mine a block in a proof of work blockchain
bitcoin uses it to keep the average block production time steady as the network's hash power changes.

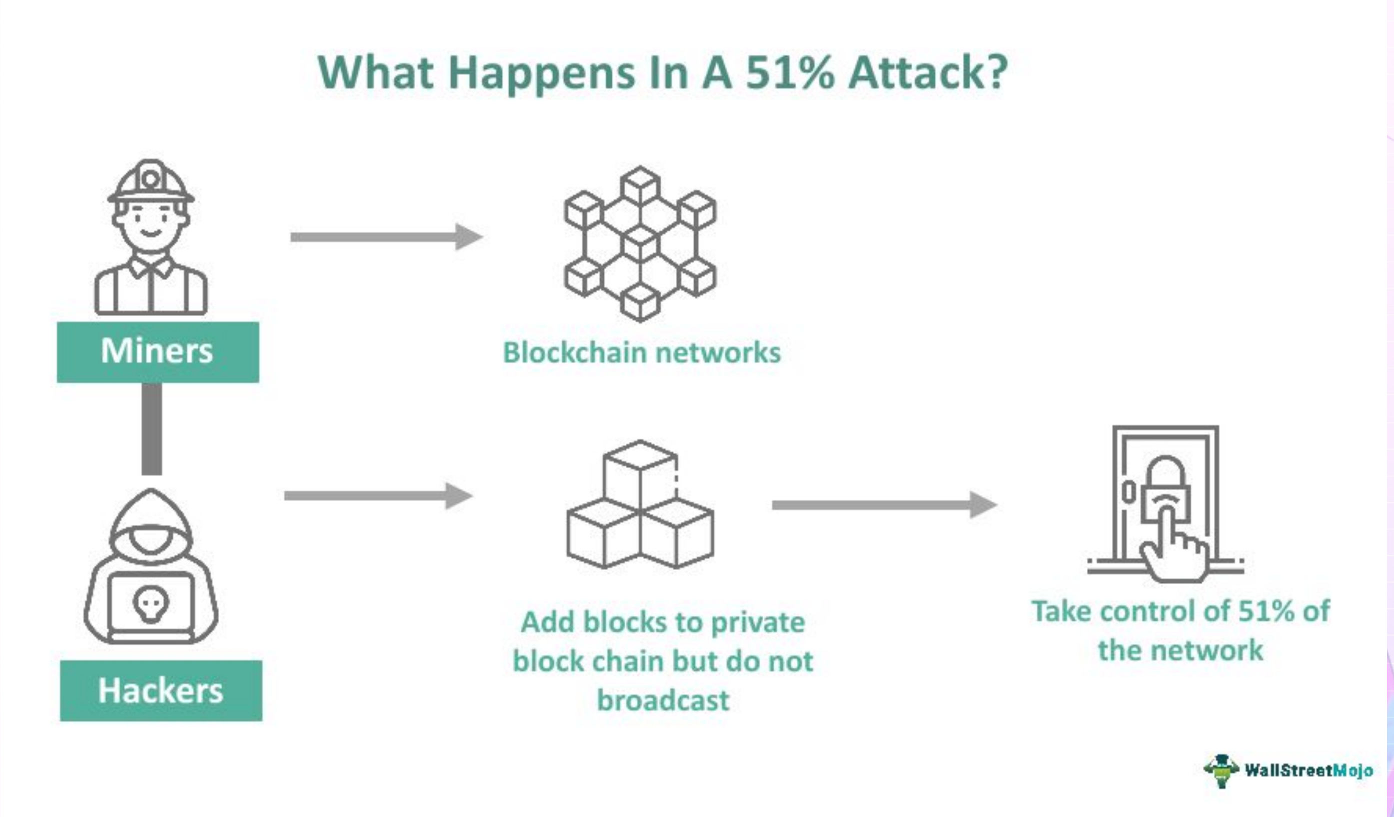
Why to keep the average block production time steady ?

- predictable supply of cryptocurrency
- regular transaction confirmation and backlog management
- network synchronization
- economic and market stability/predictability

The Platform Academy

51% Attack

A 51% attack occurs when a single entity or group controls more than 50% of the network's computational power or mining power.



The Platform Academy

51% Attack

Implications:

- **Double Spending**: The attacker can spend the same cryptocurrency more than once.
- **Transaction Reversal**: The attacker can reverse transactions, causing inconsistencies in the blockchain.
- **Network Control**: The attacker can block new transactions and prevent the confirmation of blocks.

Prevention:

- **Decentralization**: Increase the number of participants and spread mining power.
- **Consensus Mechanisms**: Use algorithms resistant to such attacks (e.g., Proof of Stake).

The Platform Academy

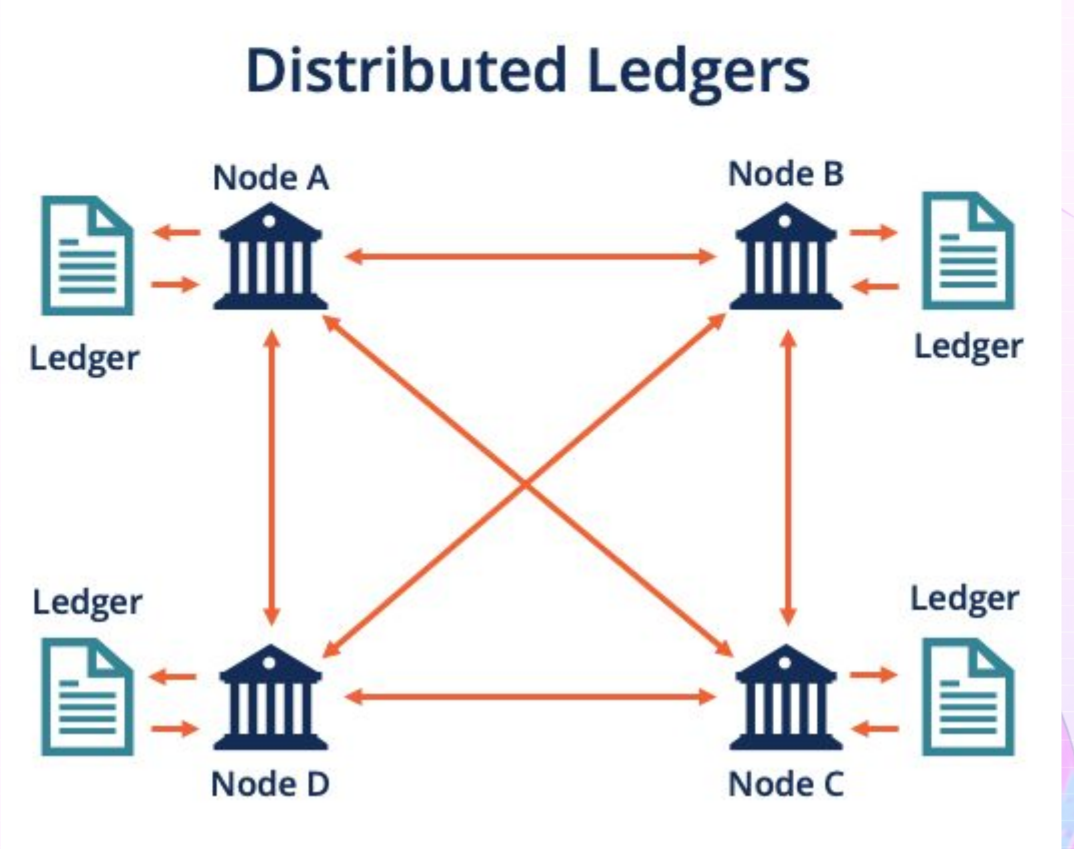
Distributed Ledger

Decentralized: No central authority; maintained by multiple participants (nodes).

Transparent: All participants share the same ledger, ensuring transaction transparency.

Immutable: Transactions cannot be altered once recorded, ensuring data integrity.

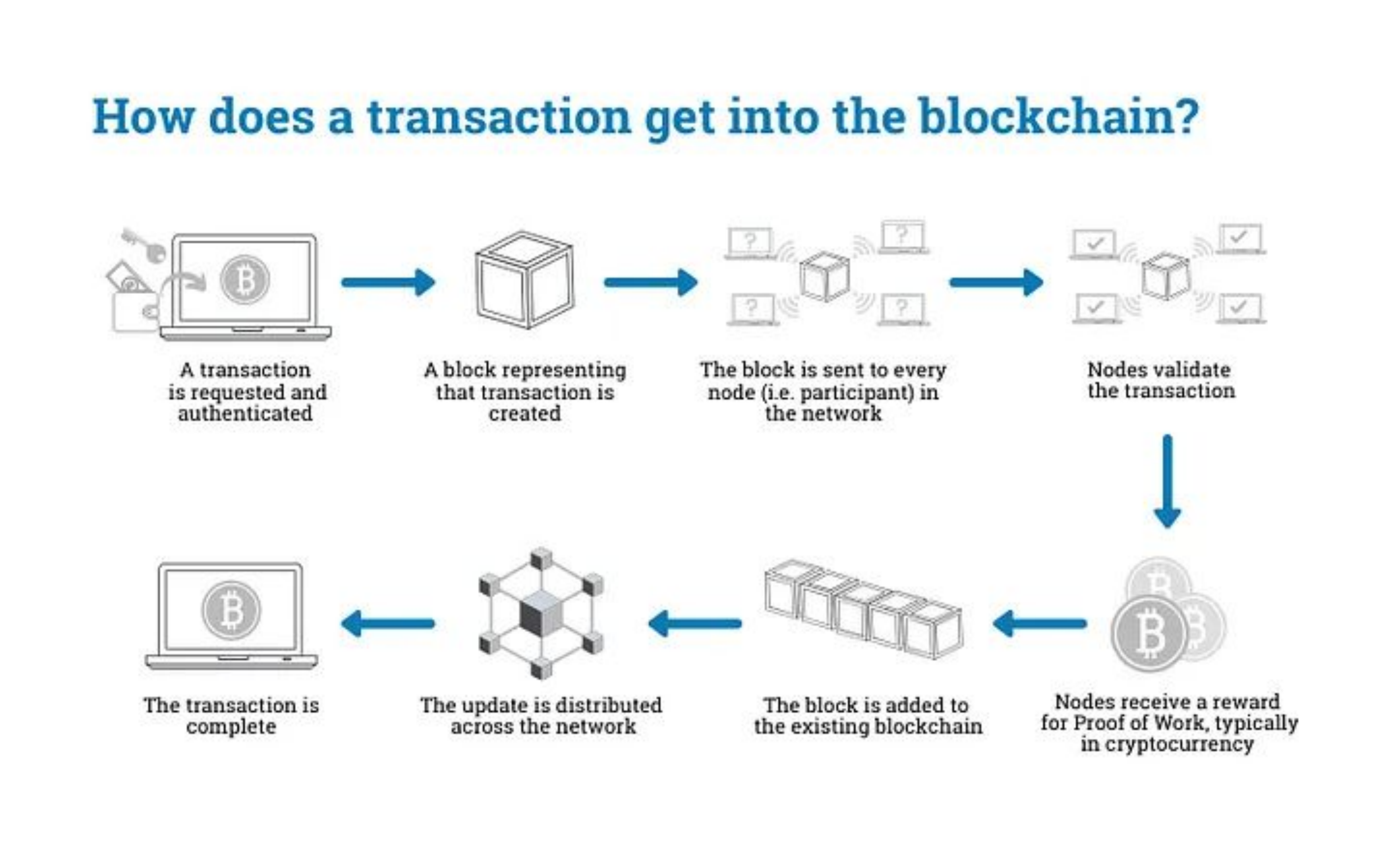
Consensus Mechanism: Transactions are validated through network-wide agreement (e.g., Proof of Work).



Reference: https://corporatefinanceinstitute.com/

The Platform Academy

Blockchain Transaction

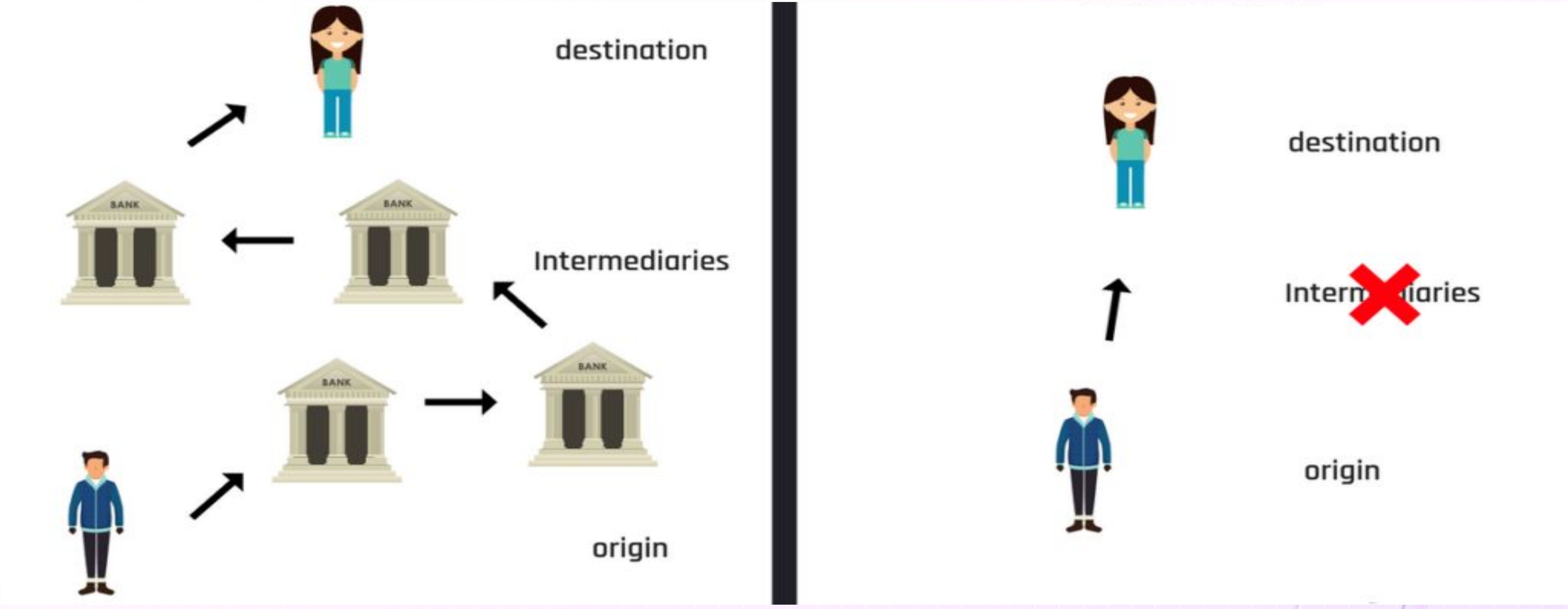


Key Features:

- **Immutability**: Transactions cannot be altered.
- **Transparency**: Visible to network participants.
- **Security**: Secured through cryptographic techniques.

The Platform Academy

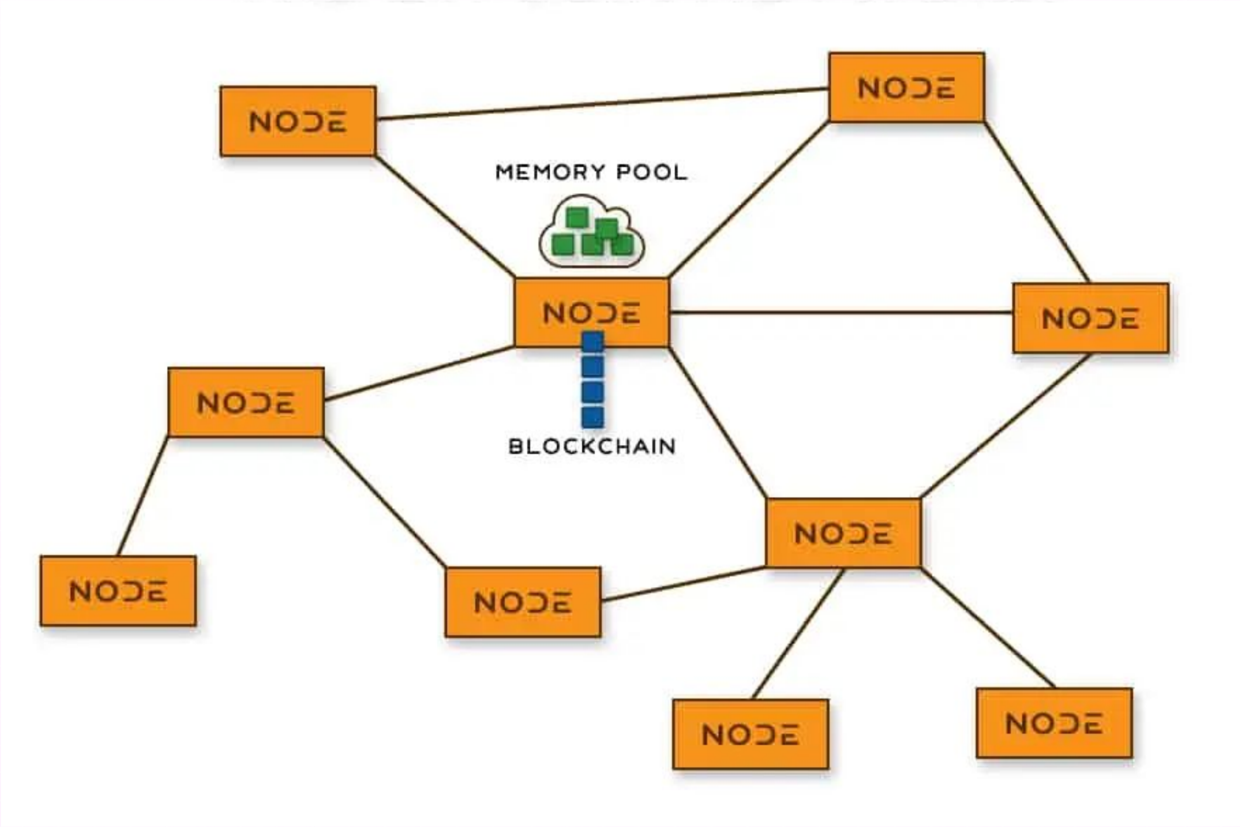
Traditional Transaction vs Blockchain Transaction



Intermediaries - Payment gateways, Banks, Payment Processing Networks like Visa and Mastercard, Clearing Houses etc..

Blockchain Mempool

Definition: Mempool (Memory Pool) is a collection of unconfirmed transactions waiting to be included in the blockchain.



The Platform Academy

Blockchain Mempool

Function:

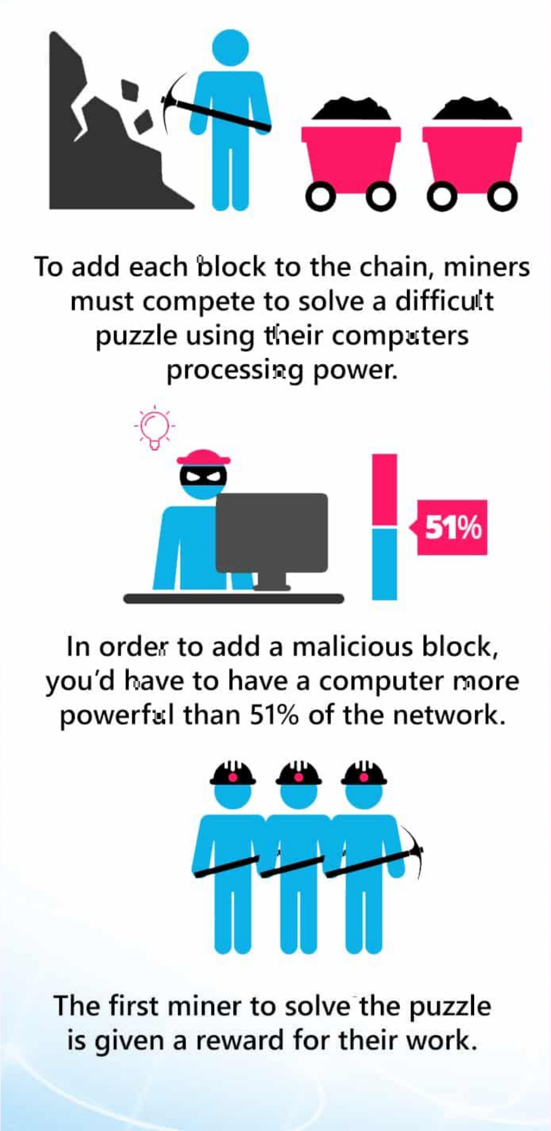
- **Transaction Holding**: Stores transactions that have been broadcast to the network but have not yet been added to a block.
- **Priority Handling**: Miners select transactions from the mempool based on transaction fees and other factors to include in the next block.

Key Features:

- **Temporary Storage**: Transactions stay in the mempool until confirmed or removed.
- **Fee Optimization**: Transactions with higher fees are prioritized by miners.
- **Network Traffic**: Acts as a buffer to manage network congestion and prevent overload.

The Platform Academy

Proof of Work



A consensus mechanism where participants (miners) solve complex mathematical puzzles to validate transactions and create new blocks.

How It Works:

- **Puzzle Solving**: Miners compete to solve a cryptographic puzzle. The first to solve it gets to add a new block to the blockchain.
- **Verification**: The solution is verified by other nodes in the network.
- **Reward**: The successful miner is rewarded with cryptocurrency.

Features:

- **Security**: Difficult puzzles prevent tampering and ensure network security.
- **Decentralization**: No central authority; control is distributed among miners.
- **Energy-Intensive**: Requires significant computational power and energy.

Examples:

- **Bitcoin**: Uses PoW to secure the network and issue new bitcoins.

SHA256 Hash

Definition: SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic hash function that produces a 256-bit (32-byte) hash value from input data.
Characteristics:

- **Fixed Size**: Regardless of input size, the output is always 256 bits.
- **Deterministic**: The same input always results in the same hash.
- **Fast Computation**: Quickly generates hash values for input data.
- **Pre-image Resistance**: Difficult to reverse-engineer the original input from the hash.
- **Collision Resistance**: Extremely hard to find two different inputs that produce the same hash.
- **Avalanche Effect**: A small change in input results in a significantly different hash.

Uses:

- **Cryptocurrencies**: Secures blockchain transactions (e.g., Bitcoin).
- **Data Integrity**: Verifies file integrity by comparing hashes.

Test it out here: https://andersbrownworth.com/blockchain/hash

Visualizing Blockchain Transactions

Blockchain Explorer - an online tool that enables you to search for real-time and historical information about a blockchain, including data related to blocks, transactions, addresses, and more.

- Etherscan (Ethereum Explorer) : https://etherscan.io/
- Bitcoin Blockchain : https://btcscan.org/
- Polygon Blockchain : https://polygonscan.com/
- Cardano Blockchain : https://explorer.cardano.org/en

The Platform Academy

Gas Limit and Gas Fee

Gas fees are the transaction fees paid by users to perform operations on a blockchain network.

Purpose of Gas Fees:

Compensation for Miners/Validators

Preventing Spam and Abuse

Resource Management

The Platform Academy

Blockchain Sharding

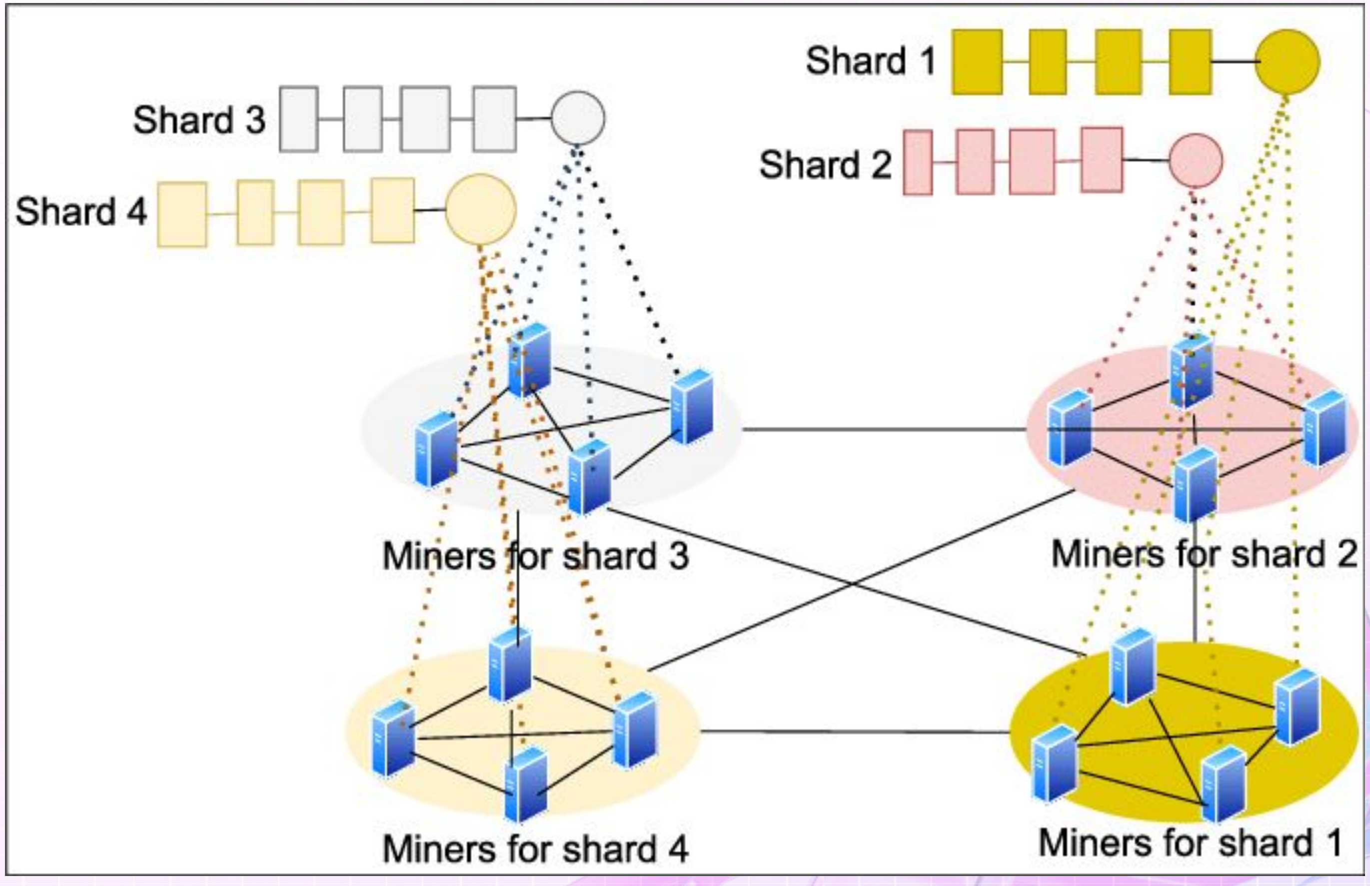
Sharding divides a blockchain into smaller, parallel shards, each handling its own subset of transactions, increasing scalability and efficiency.

How It Works:

- **Shards**: Independent sections that process transactions simultaneously.
- **Cross-Shard Communication**: Ensures consistency across the network.

Benefits:

- **Increased Scalability**: More transactions processed in parallel.
- **Reduced Latency**: Faster transaction confirmations.
- **Cost Efficiency**: Lower computational costs for nodes.



Layer 2 Solutions

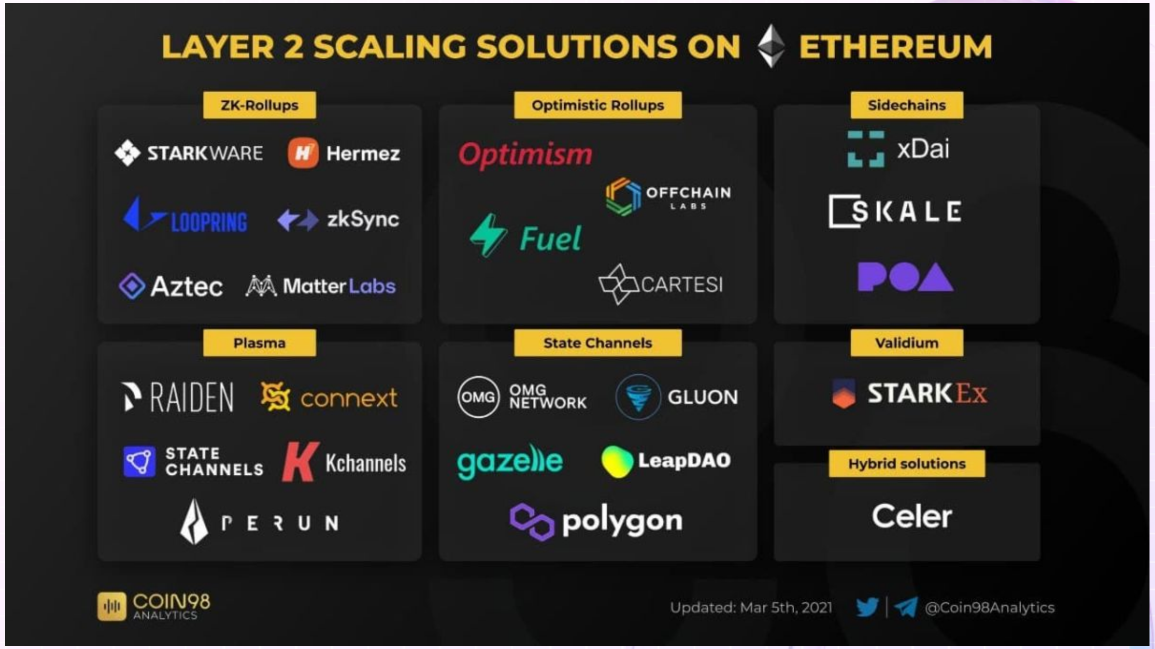
Layer 2 solutions are secondary frameworks or protocols built on top of a blockchain (Layer 1) to improve scalability, speed, and reduce transaction costs.

How It Works:

- **Off-Chain Processing**: Transactions are processed off the main blockchain and later consolidated.
- **Types**:
  - **State Channels**: Allow parties to transact off-chain and settle on-chain.
  - **Rollups**: Bundle multiple transactions off-chain and post them to the main chain as a single transaction.
  - **Sidechains**: Independent blockchains that run in parallel to the main chain, handling specific transactions.

Benefits:

- **Scalability**: Handles more transactions without congesting the main chain.
- **Lower Fees**: Reduces transaction costs by processing them off-chain.
- **Faster Transactions**: Speeds up transaction processing time.



The Platform Academy

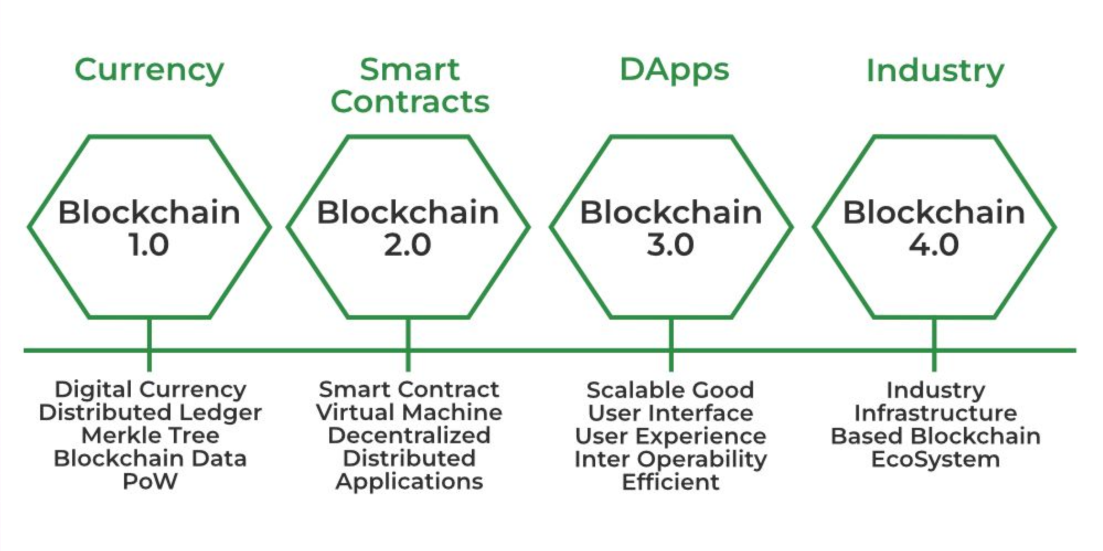
Blockchain 1.0 to 4.0

Blockchain 1.0 (Cryptocurrency)

- **Concept**: Introduced by Bitcoin in 2008.
- **Purpose**: Digital currency and decentralized payments.
- **Key Feature**: First use of blockchain to maintain a secure, transparent ledger for financial transactions.

Blockchain 2.0 (Smart Contracts)

- **Concept**: Expanded by Ethereum in 2015.
- **Purpose**: Beyond currency, enabling programmable transactions and decentralized applications (dApps).
- **Key Feature**: Smart contracts automate and execute contract terms without intermediaries.

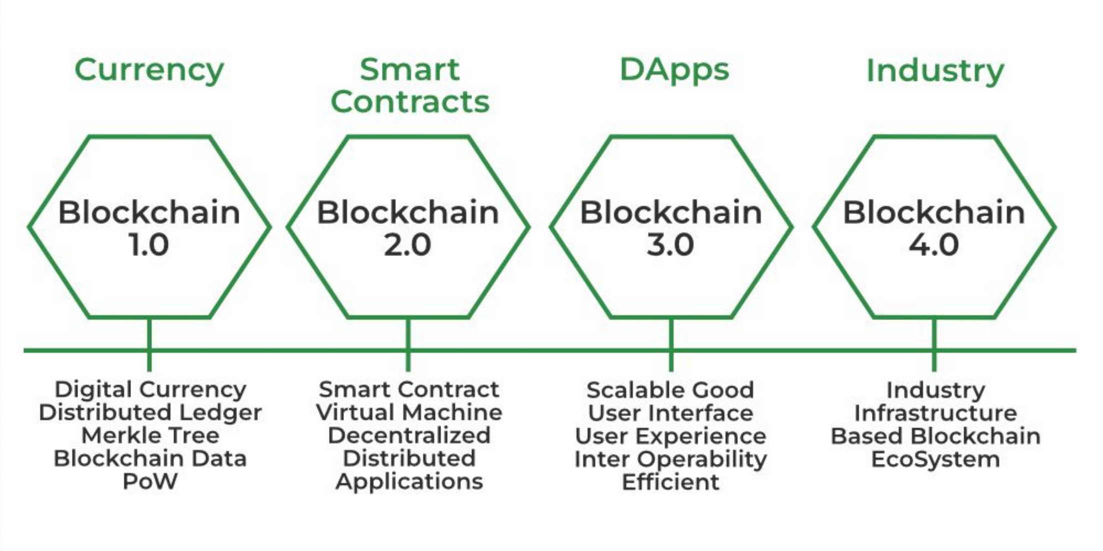


The Platform Academy

Blockchain 1.0 to 4.0

Blockchain 3.0 (Scalability and Interoperability)

- **Concept**: Focus on improving scalability, interoperability, and usability.
- **Purpose**: Enhance blockchain technology for broader applications in various sectors like finance, supply chain, and healthcare.
- **Key Features**:
  - **Scalability**: Solutions like sharding and Layer 2 technologies.
  - **Interoperability**: Cross-chain communication and integration with other blockchains.
  - **Governance**: Decentralized governance models for better decision-making.

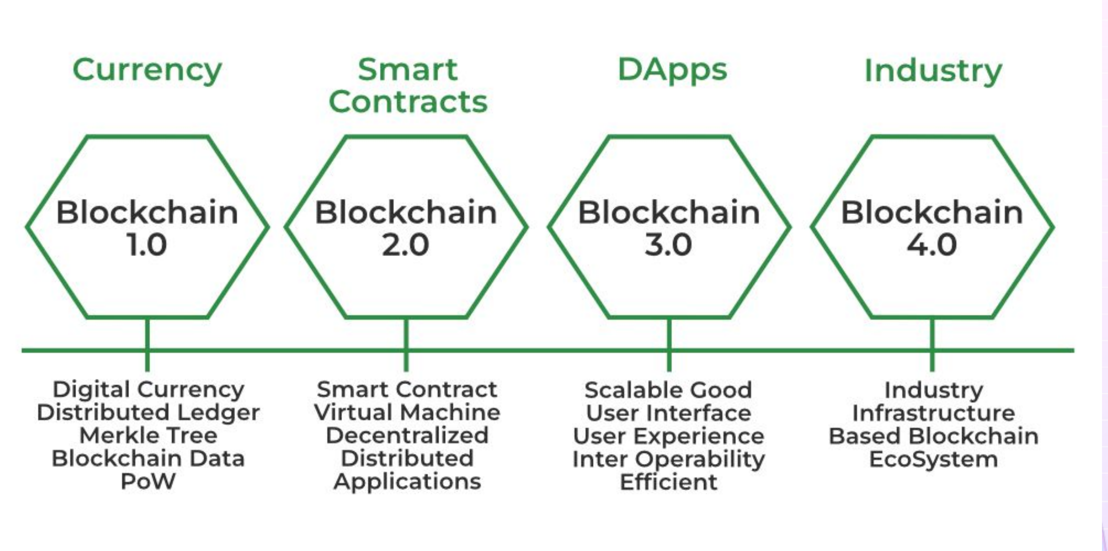


The Platform Academy

Blockchain 1.0 to 4.0

Blockchain 4.0 (Enterprise and Industry Adoption)

- **Concept**: Adoption in enterprise and industrial applications.
- **Purpose**: Drive innovation in industries by integrating blockchain for transparency, security, and efficiency.
- **Key Features**:
  - **Private and Consortium Blockchains**: Tailored for specific industries or groups.
  - **Integration with IoT and AI**: Enhanced capabilities for smart contracts and data management.



The Platform Academy

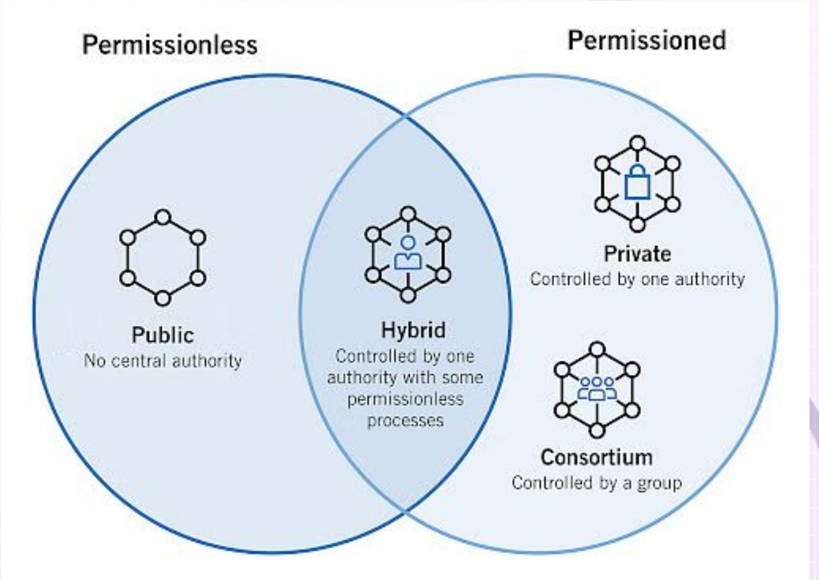
Types of Blockchain

Public Blockchain

- **Definition**: Open and decentralized; anyone can join, participate, and access the data.
- **Examples**: Bitcoin, Ethereum.
- **Features**: Transparent, secure through consensus mechanisms like Proof of Work, and decentralized control.

Private Blockchain

- **Definition**: Restricted access; controlled by a single organization or a group.
- **Examples**: Hyperledger Fabric, R3 Corda.
- **Features**: Limited access, faster transactions, and centralized control.



The Platform Academy

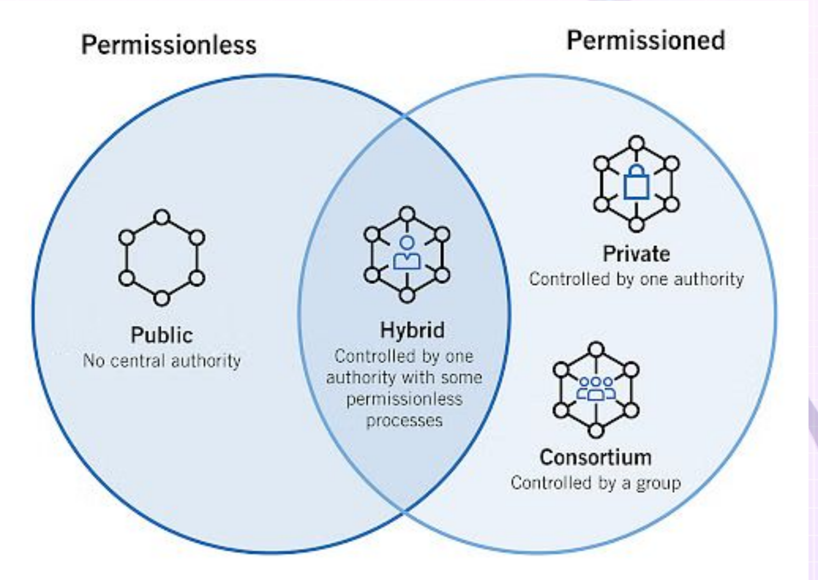
Types of Blockchain

Consortium Blockchain

- **Definition**: Semi-decentralized; governed by a group of organizations.
- **Examples**: Quorum, Hyperledger Sawtooth.
- **Features**: Collaborative governance, efficient, and restricted access to consortium members.

Hybrid Blockchain

- **Definition**: Combines elements of public and private blockchains.
- **Examples**: Dragonchain, IBM Blockchain.
- **Features**: Balances transparency with privacy, flexible access controls, and can interact with both public and private blockchains.



The Platform Academy

What is Bitcoin?

Definition: Bitcoin is a decentralized digital currency created by an unknown person or group using the pseudonym Satoshi Nakamoto in 2008.

Key Features:

- **Decentralization**: Operates on a peer-to-peer network without a central authority or government.
- **Blockchain Technology**: Transactions are recorded on a public ledger called the blockchain, ensuring transparency and security.
- **Limited Supply**: The total supply is capped at 21 million bitcoins to prevent inflation.
- **Mining**: New bitcoins are created through a process called mining, where miners solve complex mathematical puzzles to validate transactions and add them to the blockchain.

Uses:

- **Digital Currency**: Used for online transactions and as a store of value.
- **Investment**: Often bought and held as an investment asset.

Advantages:

- **Security**: Transactions are secured by cryptography and decentralized verification.
- **Transparency**: All transactions are visible on the blockchain.

The Platform Academy

What is Ethereum?

Definition: Ethereum is a decentralized, open-source blockchain platform that enables the creation and execution of smart contracts and decentralized applications (dApps). It was proposed by Vitalik Buterin in late 2013 and went live in 2015.

Key Features:

- **Smart Contracts**: Self-executing contracts with the terms of the agreement directly written into code, enabling automated and trustless transactions.
- **Decentralized Applications (dApps)**: Applications that run on the Ethereum blockchain, leveraging its smart contract capabilities.
- **Ether (ETH)**: The native cryptocurrency of the Ethereum platform, used to pay for transaction fees and computational services on the network.
- **Ethereum Virtual Machine (EVM)**: The runtime environment for executing smart contracts and dApps on Ethereum, providing a consistent execution environment across nodes.

Proof of Stake

A consensus mechanism used in blockchain networks where participants (validators) are selected to create and validate new blocks based on the number of coins they hold and are willing to "stake" or lock up as collateral.

How It Works:

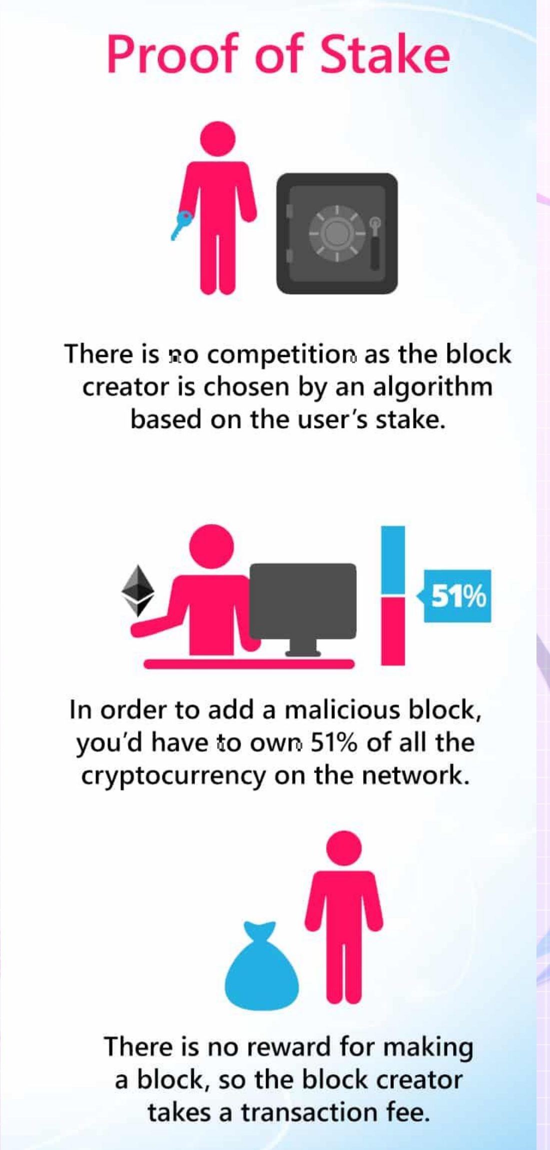
- **Staking**: Validators lock up a certain amount of cryptocurrency as a stake.
- **Selection**: Validators are chosen to create and validate new blocks based on their stake, with higher stakes increasing the chance of being selected.
- **Validation**: The chosen validator processes transactions and adds the new block to the blockchain.
- **Rewards**: Validators earn transaction fees or new coins as rewards for their participation and correct validation.

Advantages:

- **Energy Efficiency**: Less energy-intensive compared to Proof of Work (PoW) as it doesn't require solving complex puzzles.
- **Security**: Economic incentives discourage malicious behavior since validators risk losing their staked assets.
- **Scalability**: Can handle more transactions per second and scale better compared to PoW.

Examples:

- **Ethereum 2.0**: Transitioning from PoW to PoS to improve scalability and efficiency.
- **Cardano**: Utilizes PoS to validate transactions and secure the network.



Proof of Work vs Proof of Stake

Proof of Work (PoW)

A consensus mechanism where participants (miners) solve complex mathematical puzzles to validate transactions and add new blocks to the blockchain.

- **How It Works**:
  - **Puzzle Solving**: Miners compete to solve cryptographic puzzles.
  - **Block Creation**: The first to solve the puzzle gets to add a block to the blockchain.
  - **Rewards**: Miners receive cryptocurrency as a reward for their work.
- **Pros**:
  - **Security**: High security due to computational difficulty and network size.
  - **Decentralization**: Widely distributed control across many miners.
- **Cons**:
  - **Energy Consumption**: Requires significant computational power and energy.
  - **Scalability**: Limited transaction processing speed.
- **Examples**: Bitcoin, Litecoin.

Proof of Stake (PoS)

A consensus mechanism where participants (validators) are selected to create and validate new blocks based on the amount of cryptocurrency they hold and are willing to stake.

- **How It Works**:
  - **Staking**: Validators lock up a certain amount of cryptocurrency.
  - **Selection**: Validators are chosen to create new blocks based on their stake and other factors.
  - **Rewards**: Validators earn transaction fees or new coins as rewards.
- **Pros**:
  - **Energy Efficiency**: Uses less energy compared to PoW.
  - **Scalability**: Can handle more transactions per second and scale better.
- **Cons**:
  - **Centralization Risk**: Higher stakes might lead to more control by wealthy participants.
  - **Initial Distribution**: May favor those who already hold a significant amount of the cryptocurrency.
- **Examples**: Ethereum 2.0, Cardano.

Crypto Wallet

Definition: A digital tool that allows users to store, manage, and transact with cryptocurrencies.
Types:

1. **Hot Wallets**:
    - **Definition**: Online wallets connected to the internet.
    - **Examples**: Software wallets, mobile apps, web wallets.
    - **Pros**: Convenient and easy to access for frequent transactions.
    - **Cons**: Higher risk of hacks and security breaches.
2. **Cold Wallets**:
    - **Definition**: Offline wallets not connected to the internet.
    - **Examples**: Hardware wallets, paper wallets.
    - **Pros**: Enhanced security and protection against online attacks.
    - **Cons**: Less convenient for frequent transactions.

Key Components:

- **Public Key**: An address that others use to send cryptocurrency to the wallet.
- **Private Key**: A secret key used to access and manage the cryptocurrency; must be kept secure.

Functions:

- **Storage**: Safely stores cryptocurrency and private keys.
- **Transactions**: Allows sending and receiving of cryptocurrencies.
- **Balance Management**: Displays account balance and transaction history.

The Platform Academy

Metamask

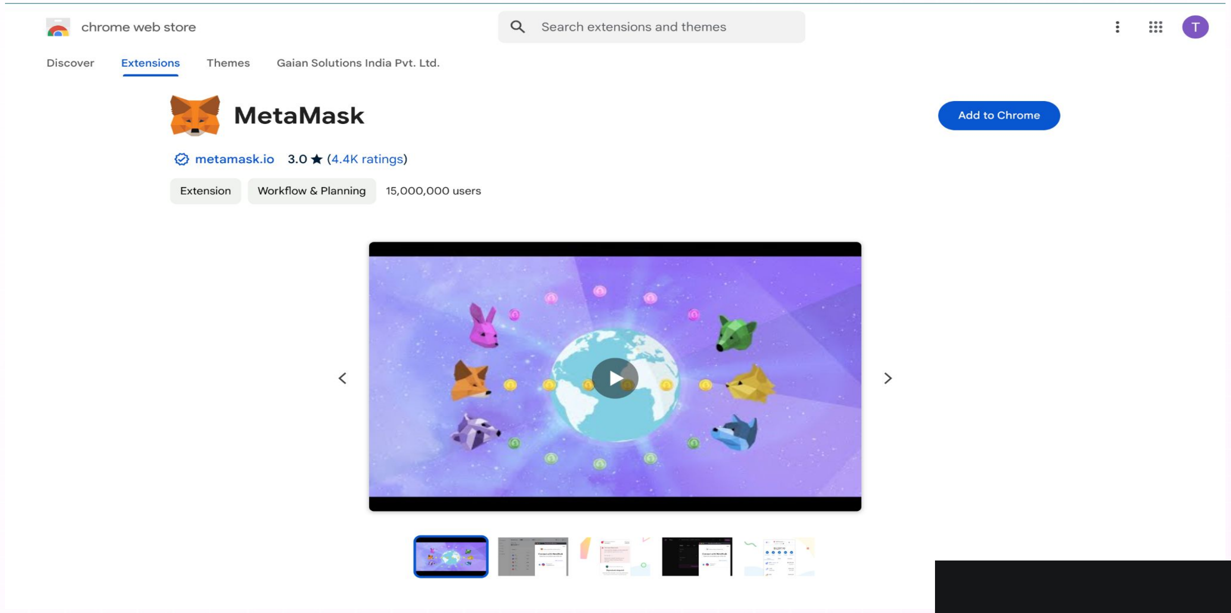
MetaMask is a popular browser extension and mobile app that acts as a cryptocurrency wallet and gateway to the Ethereum blockchain and other decentralized applications (dApps).

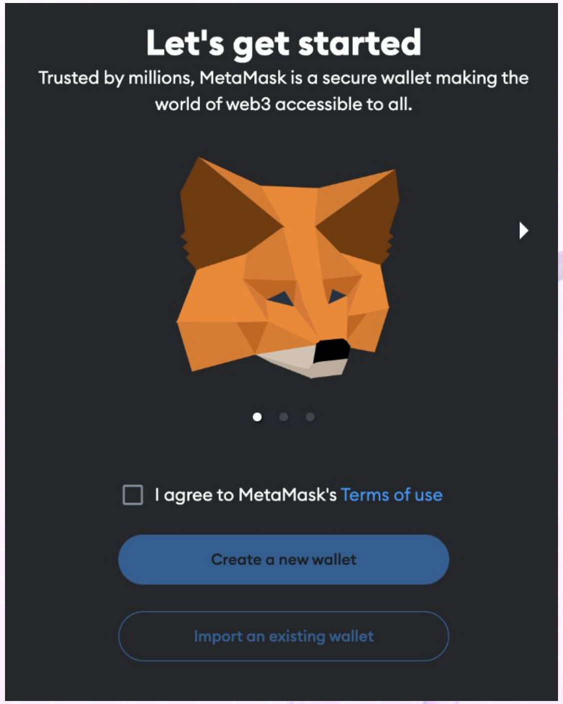
Provides secure key storage through encrypted local storage. Users manage their private keys with a password and recovery phrase.

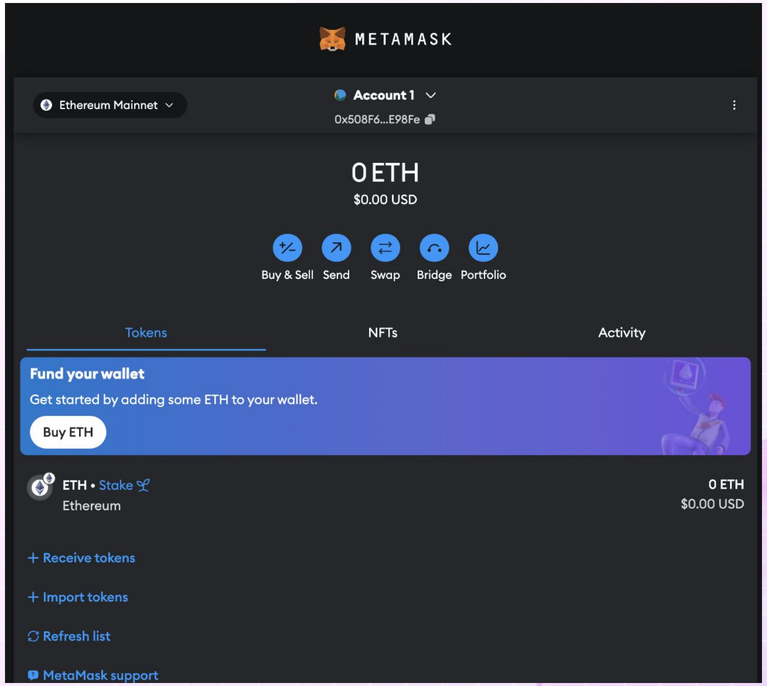
Allows users to interact with decentralized applications directly from their browser or mobile device.

Connects to Ethereum Mainnet, test networks, and custom networks.

Metamask - Setup - Chrome Extension







The Platform Academy

Metamask - Sepolia Test Net

Connect to Sepolia test net

Get test tokens from faucets

https://cloud.google.com/application/web3/faucet
https://faucets.chain.link/sepolia

The Platform Academy

Dev Environment Setup

Installing node.js and npm
https://nodejs.org/en

Remix IDE
https://remix.ethereum.org/

Connect Remix to MetaMask
During deployment of a smart contract, use “Injected Provider - MetaMask” as the Environment

Smart Contracts

Self-executing contracts with the terms directly written into code on a blockchain. They automatically execute and enforce the contract when predefined conditions are met.

Automation: Executes transactions and enforces rules automatically, without the need for intermediaries.

Transparency: Contract details and transactions are recorded on the blockchain, making them visible and verifiable.

Immutability: Once deployed, the contract code cannot be altered, ensuring the contract terms are consistently enforced.

Security: Protected by blockchain's cryptographic security, reducing risks of fraud and tampering.

Deployment: Written in code and deployed on a blockchain platform (e.g., Ethereum).

Trigger: Executes actions (e.g., transferring funds) when specific conditions are met.

Execution: Automatically performs contractual obligations without human intervention.

Solidity Programming

Solidity is an object-oriented programming language created specifically by the Ethereum Network team for constructing and designing smart contracts on Blockchain platforms.

It's used to create smart contracts that implement business logic and generate a chain of transaction records in the blockchain system.

It acts as a tool for creating machine-level code and compiling it on the Ethereum Virtual Machine (EVM).

Solidity Programming - Syntax and Structure

Contract Structure: Start with the contract keyword followed by the contract name.

State Variables: Variables that hold the state of the contract. Example: `uint256 balance;`

Functions: Define the behavior of the contract. Example: `function setBalance(uint256 newBalance) public { balance = newBalance; }`.

Modifiers: Used to change the behavior of functions. Example: `modifier onlyOwner { require(msg.sender == owner); _; }`.

Events: Allow logging to the blockchain. Example: `event BalanceSet(uint256 newBalance);`.

Callback Function: A special function executed when the contract receives Ether without any data.

The Platform Academy

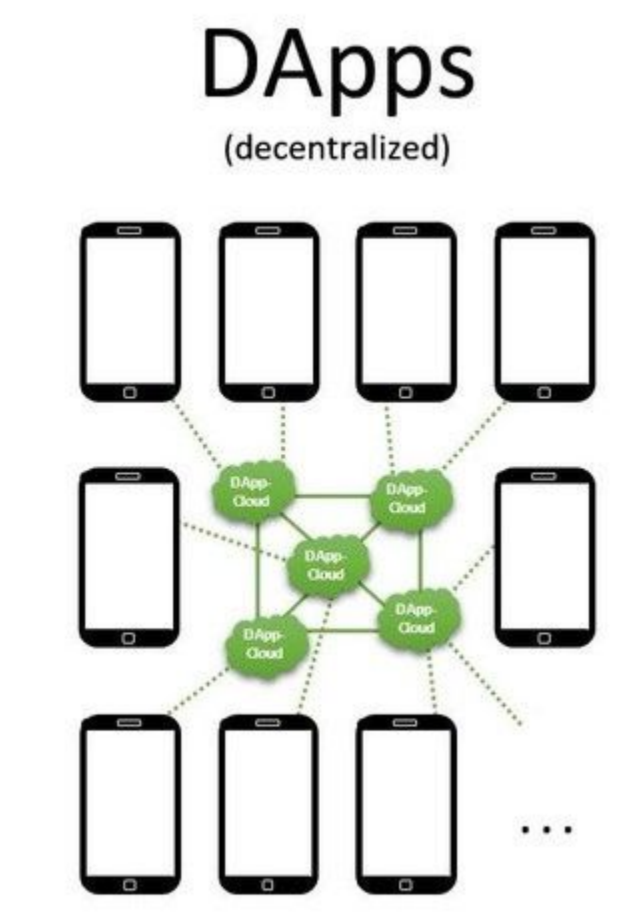
DApps

Decentralized Applications that run on a blockchain network.
They leverage smart contracts to operate autonomously.

Smart Contracts: dApps use smart contracts to define and enforce the rules of the application.

Blockchain Integration: Interact with blockchain networks to store data and execute transactions.

User Interface: Front-end interfaces connect users to the dApp, while the back-end operations are handled by smart contracts on the blockchain.



The Platform Academy

HelloWorld Smart Contract

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract HelloWorld {
    string public message = "Hello, Web3!";

    function setMessage(string memory newMessage) public {    infinite gas
        message = newMessage;
    }

    function getMessage() public view returns (string memory) {    infinite gas
        return message;
    }
}
```

Voting Smart Contract

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract Voting {
    struct Candidate {
        uint id;
        string name;
        uint voteCount;
    }

    mapping(address => bool) public voters;
    mapping(uint => Candidate) public candidates;
    uint public candidatesCount;

    address public owner;

    modifier onlyOwner() {
        require(msg.sender == owner, "Only owner can call this function.");
        _;
    }

    constructor() {    825704 gas 800600 gas
        owner = msg.sender;
    }

    function addCandidate(string memory name) public onlyOwner {    infinite gas
        candidatesCount++;
        candidates[candidatesCount] = Candidate(candidatesCount, name, 0);
    }

    function vote(uint candidateId) public {    infinite gas
        require(!voters[msg.sender], "You have already voted.");
        require(candidateId > 0 && candidateId <= candidatesCount, "Invalid candidate ID.");

        voters[msg.sender] = true;
        candidates[candidateId].voteCount++;
    }

    function getCandidate(uint candidateId) public view returns (string memory name, uint voteCount) {    infinite gas
        require(candidateId > 0 && candidateId <= candidatesCount, "Invalid candidate ID.");
        Candidate memory candidate = candidates[candidateId];
        return (candidate.name, candidate.voteCount);
    }

    function isOwner() public view returns (bool) {    2540 gas
        return msg.sender == owner;
    }
}
```

Voting Smart Contract

Only owner can add candidates

The Platform Academy

Use Cases

1. Financial Services

- **Decentralized Finance (DeFi)**: Enables lending, borrowing, and trading without intermediaries.
- **Automated Payments**: Executes payments automatically based on predefined conditions.

2. Supply Chain Management

- **Tracking Goods**: Records and verifies the movement of goods in real-time.
- **Authenticity Verification**: Ensures the authenticity of products through immutable records.

3. Real Estate

- **Property Transactions**: Automates buying, selling, and transferring ownership of properties.
- **Smart Leases**: Manages rental agreements and payments automatically.

4. Healthcare

- **Medical Records**: Secures and manages patient data with privacy and accessibility controls.
- **Insurance Claims**: Automates insurance claims and payouts based on predefined conditions.

5. Voting Systems

- **Secure Voting**: Ensures transparency and integrity of voting processes by recording votes on the blockchain.
- **Election Auditing**: Provides immutable records for auditing election results.

6. Intellectual Property

- **Digital Rights Management**: Manages and enforces intellectual property rights for digital content.
- **Royalty Payments**: Automates the distribution of royalties to creators based on usage data.

Advanced Blockchain Topics

Layer 2 Solutions and Scalability

Scalability Challenges:

- **High Gas Fees:** As the number of users on a blockchain like Ethereum increases, so do the fees, making it less feasible for smaller transactions.
- **Slow Transaction Speeds:** Ethereum, for example, can process around 15 transactions per second (TPS), which is slow compared to traditional payment systems.

Layer 2 Solutions:

Optimistic Rollups

- **How it Works:** Optimistic Rollups execute transactions off-chain and then post the transaction data on-chain. They assume that transactions are valid, and if a dispute arises, the transaction is checked on-chain.

Advanced Blockchain Topics

Layer 2 Solutions:

ZK-Rollups:

- **How it Works:** ZK-Rollups bundle hundreds of transfers off-chain and create a cryptographic proof, called a SNARK (Succinct Non-Interactive Argument of Knowledge), which is posted on-chain.
- **Example:** zkSync.

State Channels:

- **How it Works:** Participants lock up some assets in a multi-signature wallet on the main blockchain, then they can transact off-chain. Only the final state is recorded on-chain, reducing the load on the network.
- **Example:** Lightning Network for Bitcoin.

The Platform Academy

Advanced Blockchain Topics

Interoperability and Cross-Chain Communication

The Need for Interoperability:

- **Different Blockchains, Different Protocols:** As different blockchains like Bitcoin, Ethereum, and Polkadot evolve, they need a way to communicate and transfer assets without intermediaries.
- **Cross-Chain Bridges:**
  - **How it Works:** Bridges connect two blockchains, allowing tokens or data to move between them. This is crucial for decentralized finance (DeFi) applications that want to offer services across multiple platforms.
  - **Example:** Polkadot Relay Chain allows different blockchains (parachains) to interoperate.

Atomic Swaps

- **How it Works:** Atomic swaps enable the exchange of one cryptocurrency for another without the need for a centralized exchange. The process is trustless, meaning the swap is either completed or fails entirely.

The Platform Academy

Advanced Blockchain Topics

Key Projects:

Polkadot:

- **Relay Chain:** The main chain of Polkadot that provides shared security and enables cross-chain communication.
- **Parachains:** Independent blockchains that connect to the Relay Chain.

Cosmos:

- **IBC (Inter-Blockchain Communication):** A protocol that allows different blockchains within the Cosmos network to communicate with each other.

The Platform Academy

Advanced Blockchain Topics

Emerging Technologies :

Polkadot:

- **Shared Security:** All parachains on Polkadot share the security of the Relay Chain, making it easier for developers to launch new blockchains.
- **Interoperability:** Polkadot architecture allows different blockchains to exchange messages and transactions in a trustless way.

Cardano:

- **Proof of Stake (Ouroboros):** Cardano uses a unique PoS protocol that is more energy-efficient than traditional Proof of Work (PoW).
- **Layered Architecture:** Cardano separates the network into two layers—the Cardano Settlement Layer (CSL) and the Cardano Computation Layer (CCL), allowing for flexibility and scalability.

Solana:

- **Proof of History (PoH):** Solana's unique consensus mechanism that timestamps transactions, allowing the network to process thousands of transactions per second.
- **High Throughput:** Solana can handle around 50,000 TPS, making it ideal for high-frequency applications like decentralized exchanges (DEXs).

Island of Knowledge

As our island of knowledge grows, so does the shore of our ignorance.

shore of ignorance - the increasing amount of things we don't know

island of knowledge - all that we DO know that we didn't know before.

while acting on the knowledge of what we do know, we can continue to grow by exploring the shores of what we don't.

Remember to have fun along the way

The Platform Academy

Contact

Jose Anitt Goli

gjanitt@gmail.com

8767981729

The Platform Academy

Conclusion

Great career opportunities at Mobius..!!

Do your life's best work with us..

Always on the lookout for right talent

Come join us..! or Train with us!

Thank you.

The Platform Academy