Blockchain Honor Degree Sem VII

HBCC701 : Blockchain Development

Module - 3 : Smart Contract (10 Hours)

Instructor : Mrs. Lifna C S

Topics to be covered

- **Smart Contracts**
  - **Use cases of smart contract,**
  - **Smart Contracts: Opportunities, Risks**
- **Solidity programming,**
  - **Smart Contract programming using solidity,**
  - **mapper function,**
- **ERC Standards**
  - **ERC20 Token**
  - **ERC721 Token,**
  - **Comparison between ERC20 & ERC721,**
- **ICO Initial Coin Offerings,**
- **Metamask (Ethereum Wallet),**
- **Setting up development environment,**

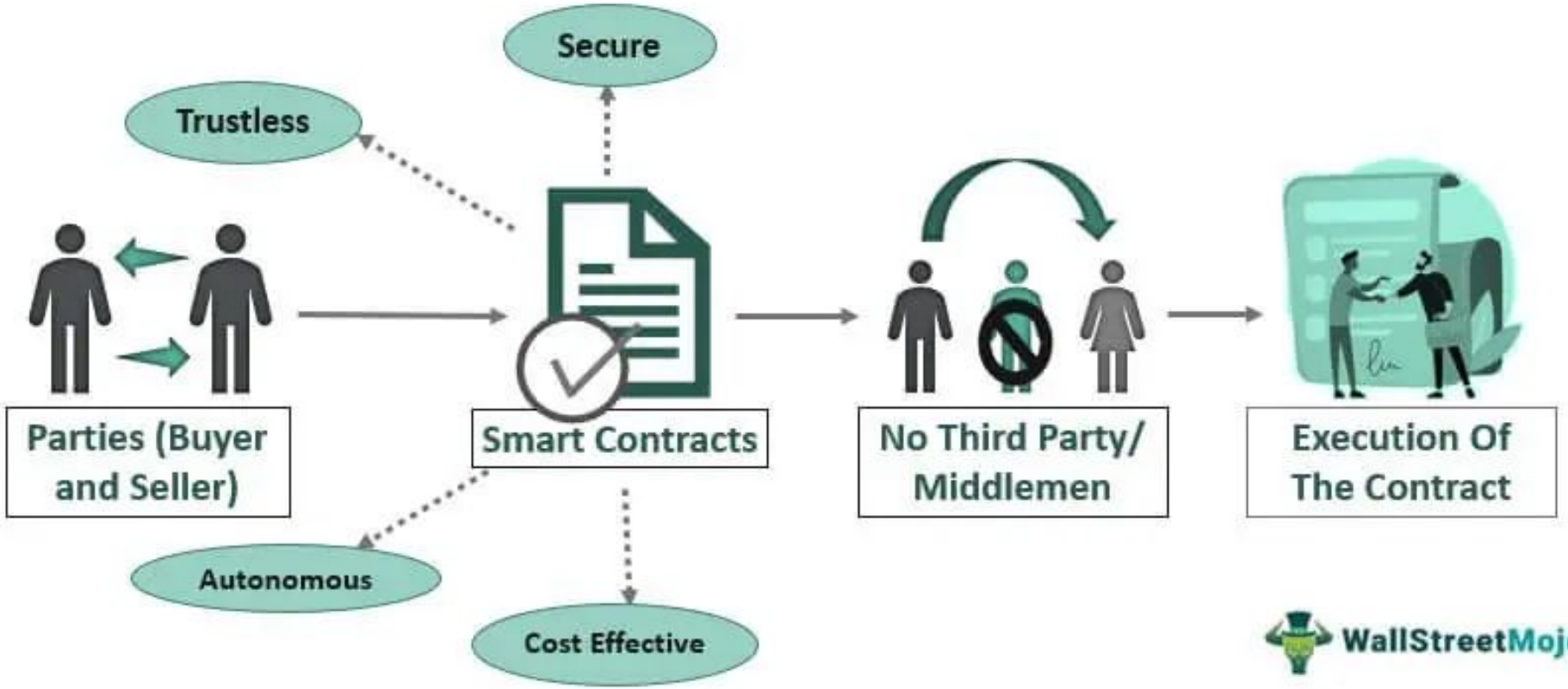
Self-learning Topics: Cryptocurrencies and their security issues, Consensus mechanisms, Digital Signatures

What is a Smart Contract ?

- A **self-executing program** that automates the actions required in an agreement or contract. Once completed, the transactions are trackable and irreversible.
- Smart contracts **permit trusted transactions and agreements to be carried out among disparate, anonymous parties** without the need for a central authority, legal system, or external enforcement mechanism.
- **do not contain legal language**, terms, or agreements
- **Only code that executes actions when specified conditions** are met.
- Nick Szabo, an American computer scientist
  - invented a virtual currency called "**Bit Gold**" in 1998,
  - defined smart contracts as computerized transaction protocols that execute the terms of a contract

Courtesy : Investopedia

What is a Smart Contract ?



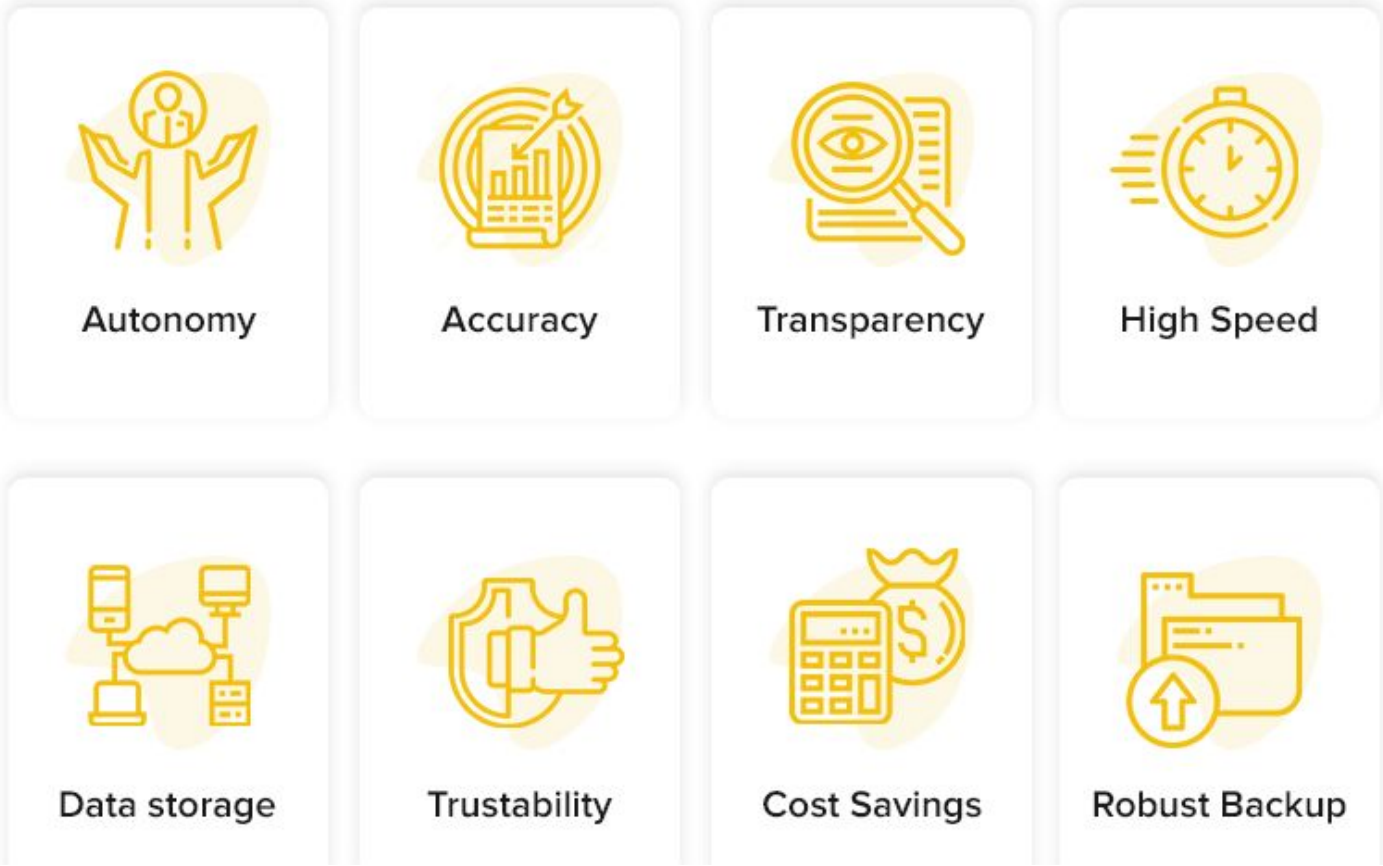
Courtesy : WallStreetMojo

Features of Smart Contracts

1. **Distributed:** Everyone on the network is guaranteed to have a copy of all the conditions of the smart contract and they cannot be changed by one of the parties. A smart contract is replicated and distributed by all the nodes connected to the network.
2. **Deterministic**: Smart contracts can only perform functions for which they are designed only when the required conditions are met. The final outcome will not vary, no matter who executes the smart contract.
3. **Immutable:** Once deployed smart contract cannot be changed, it can only be removed as long as the functionality is implemented previously.
4. **Autonomy:** There is no third party involved. The contract is made by you and shared between the parties. No intermediaries are involved which minimizes bullying and grants full authority to the dealing parties. Also, the smart contract is maintained and executed by all the nodes on the network, thus removing all the controlling power from any one party's hand.
5. **Customizable**: Smart contracts have the ability for modification or we can say customization before being launched to do what the user wants it to do.
6. **Transparent:** Smart contracts are always stored on a public distributed ledger called blockchain due to which the code is visible to everyone, whether or not they are participants in the smart contract.
7. **Trustless:** These are not required by third parties to verify the integrity of the process or to check whether the required conditions are met.
8. **Self-verifying:** These are self-verifying due to automated possibilities.
9. **Self-enforcing:** These are self-enforcing when the conditions and rules are met at all stages.

Courtesy : GeeksforGeeks

Benefits of Smart Contracts



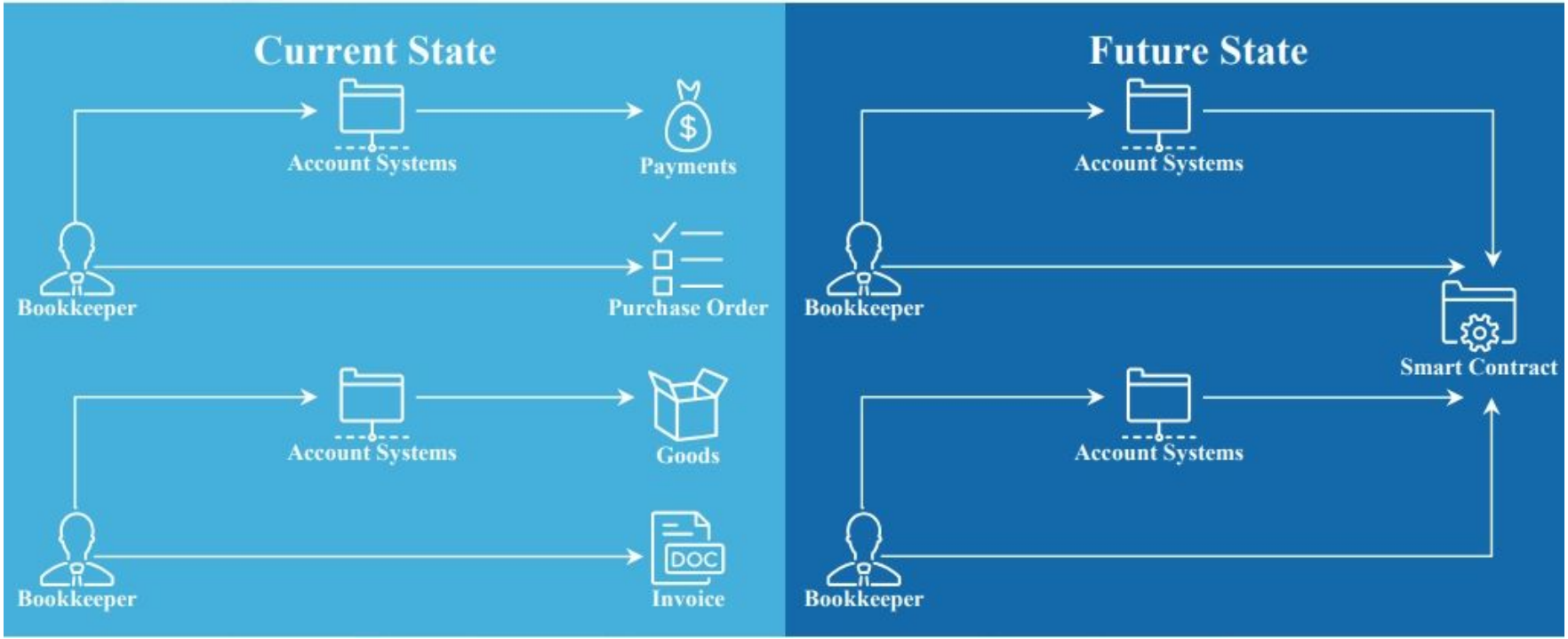
Courtesy : Appinventiv

Types of Smart Contract

1. **Smart Legal Contracts**
   - Legally enforceable
   - Require the parties to fulfill their contractual obligations.
   - Failure to do so may result in strict legal actions against them.
2. **Decentralized Autonomous Organizations (DAO)**
   - These are blockchain communities that are bound to specific rules coded into blockchain contracts combined with governance mechanisms.
   - Any action taken by the community members gets replaced by a self-enforcing code.
3. **Application Logic Contracts (ALC)**
   - contain an application-based code that remains in sync with other blockchain contracts.
   - It enables communication across different devices, such as the merger of the Internet of Things with blockchain technology.

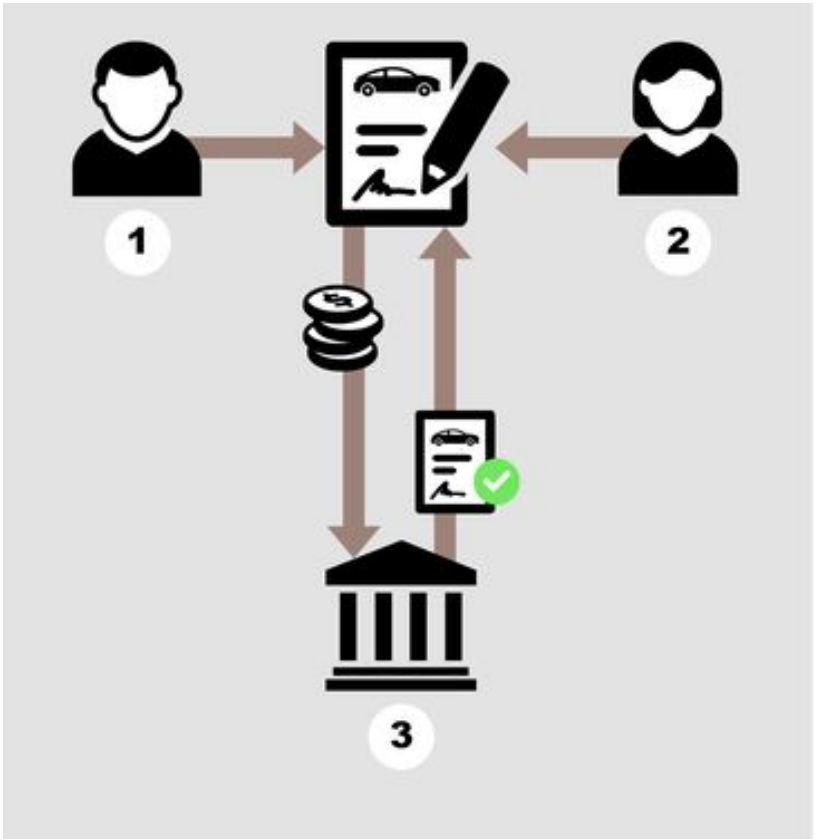
Courtesy : WallStreetMojo

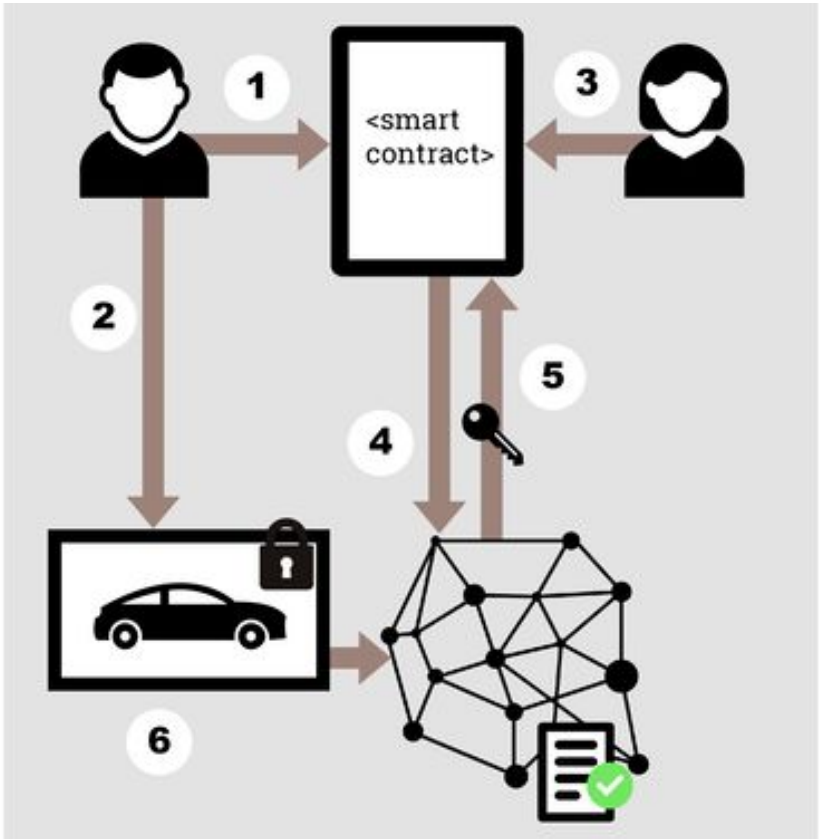
Future of Financial Reporting



Courtesy : AI Multiple

Traditional Contracts Vs Smart Contracts





Courtesy : LinkedIn - Marvin Fedler

Traditional Contracts Vs Smart Contracts - Explained

Traditional Contracts

1. Bob would like to sell his car.
2. Alice would like to buy a car.
3. A third party (intermediary) enables the trust that is needed in order to transfer the ownership of the car. Mostly different intermediaries are needed: motor vehicle registration authority, notary, insurance company. All middlemen take fees.

Courtesy : LinkedIn - Marvin Fedler

Traditional Contracts Vs Smart Contracts - Explained

Smart Contracts

1. Bob would like to sell his car. He defines in a smart contract the conditions by which he will sell the car and signs the contract with his private key.
2. Bob leaves his car locked with a smart lock in his garage. The car has its own blockchain address and the smart lock is controlled by a smart contract.
3. Alice would like to buy a car. She finds Bobs car on an internet platform and signs the smart Bob's contract with her private key. She adds the agreed amount from her blockchain address to Bob's blockchain address.
4. As soon as the smart contract is executed the whole blockchain network will check if Bob is the real owner of the car and if Alice has enough money to buy the car.
5. If all peers in the blockchain network agree on the same state, it means that all conditions in the smart contract are met. The access code for the smart lock will be transferred to Alice and the blockchain address of the car will be registered to Alice. Bob will get the defined amount of money in his blockchain address.
6. Alice will be able to open the smart lock with her private key.

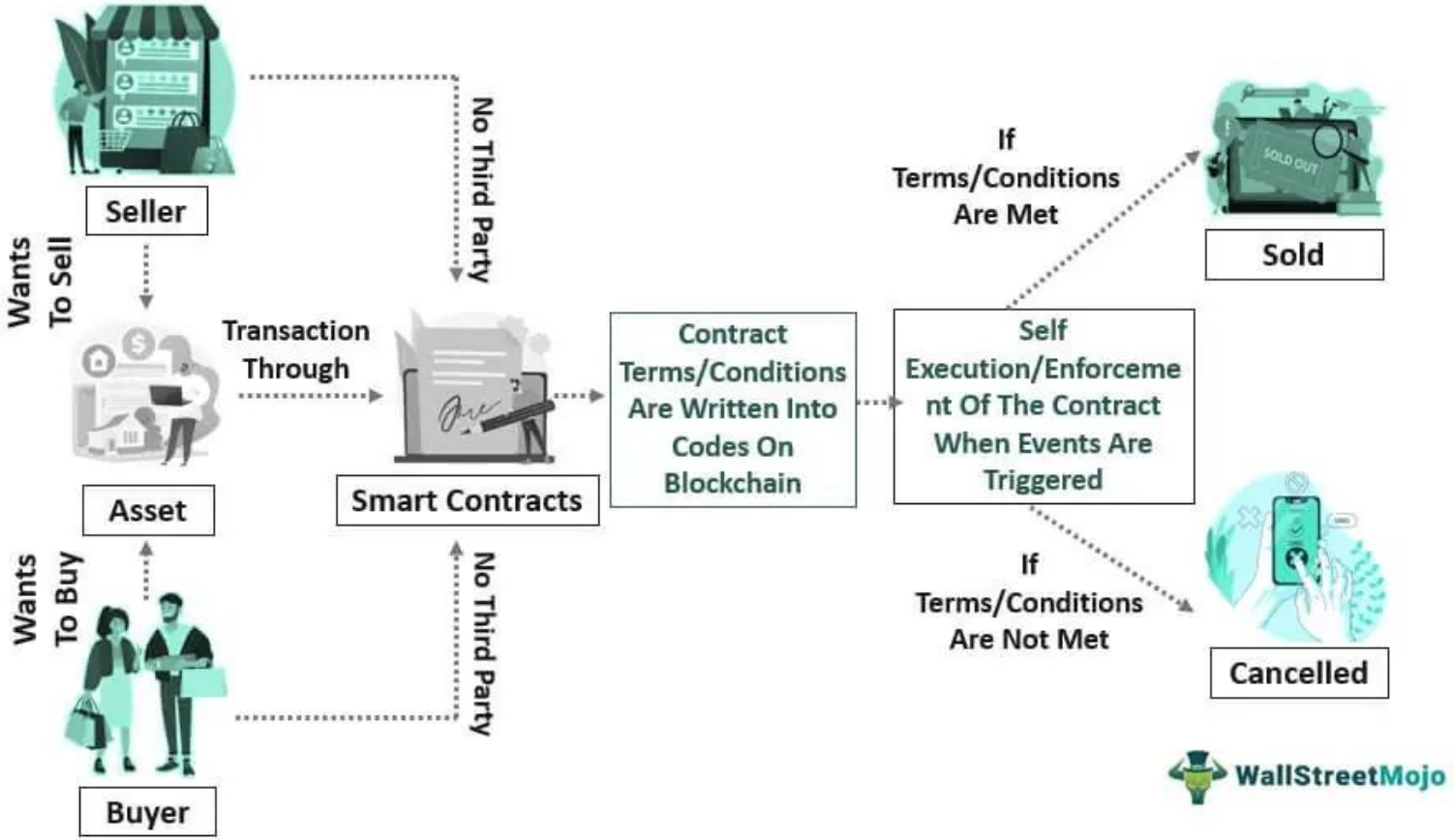
Courtesy : LinkedIn - Marvin Fedler

Future of Financial Reporting

	Written contract	Smart contract
Language/Code	Human language	Machine computer code
Automation	All parts of the agreement	Only transactions to be automated
Recording	Conditions can be written on a paper by the concerned parties	Embedded into blockchain or another ledger
Modification of an internal state	Subject to interpretation	Generally immutable

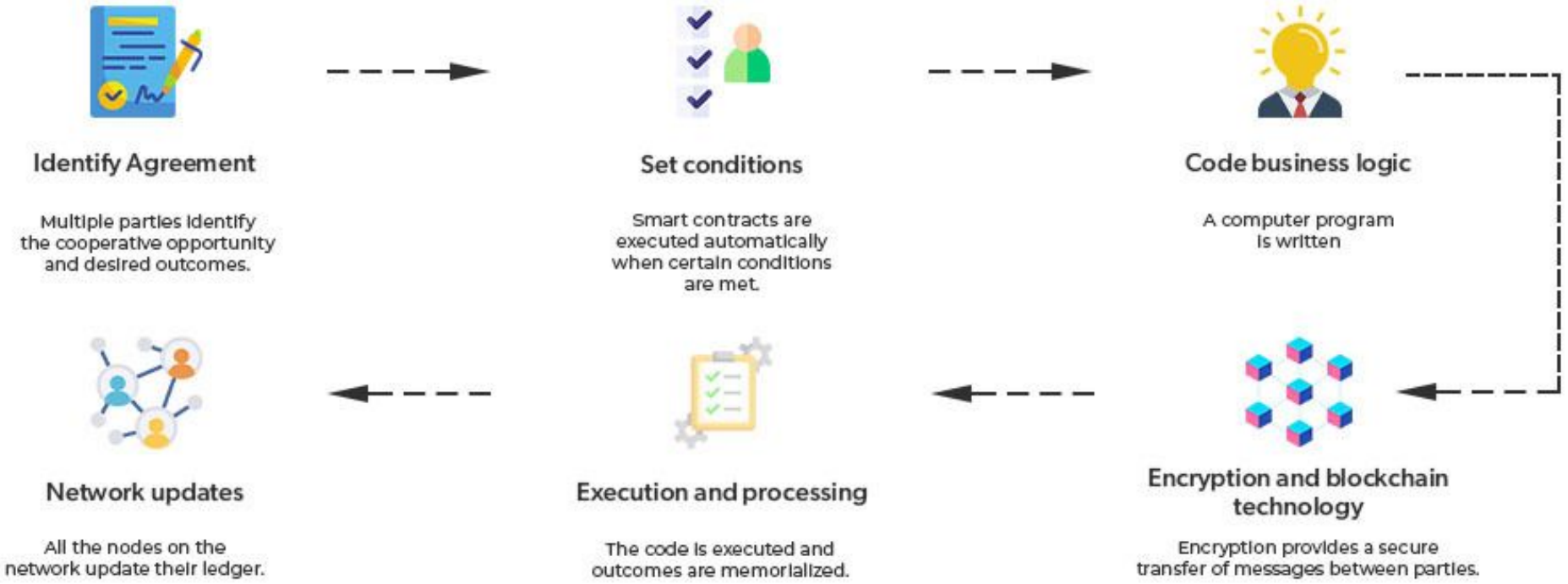
Courtesy : Cointelegraph

Smart Contract Functioning



Courtesy : WallStreetMojo

How do smart contracts work ?



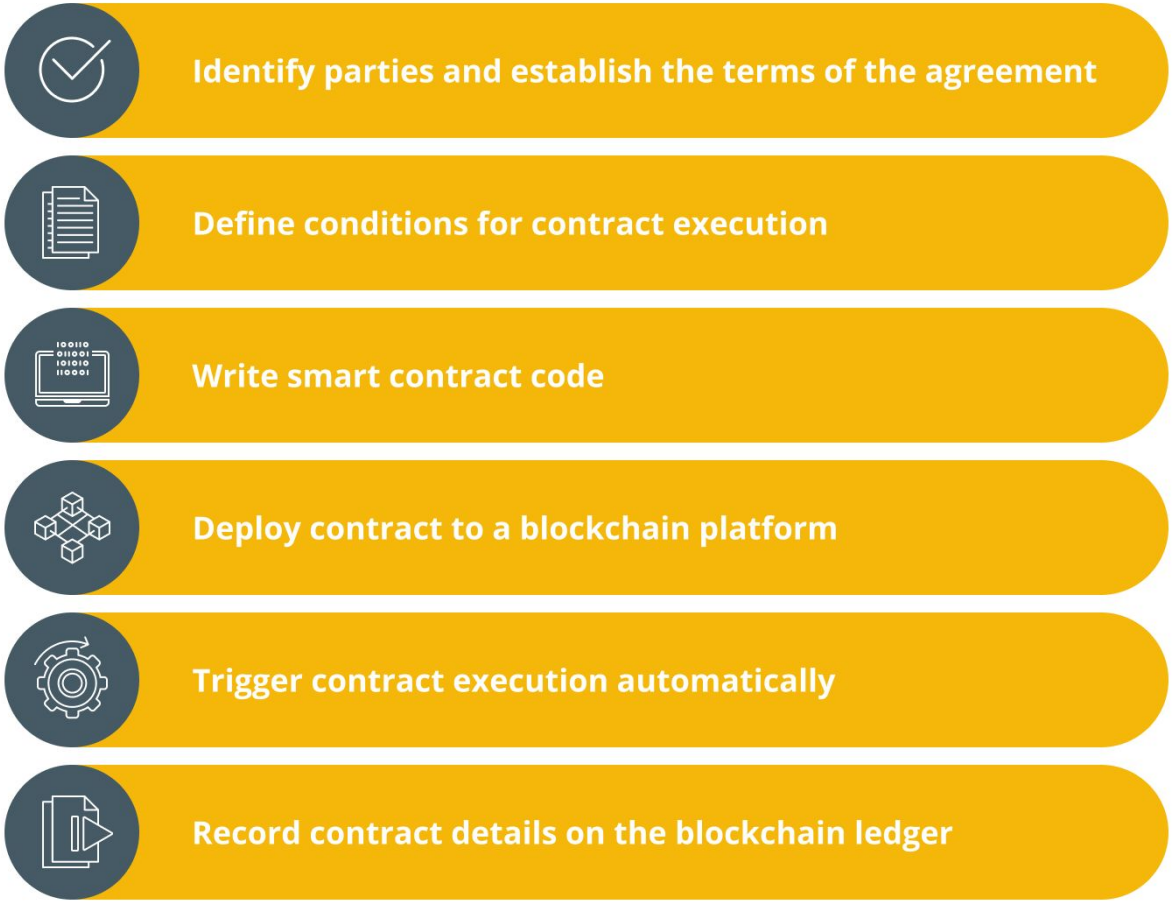
Courtesy : GeeksforGeeks

How do smart contracts work ?

1. **Identify Agreement:** Multiple parties identify the cooperative opportunity and desired outcomes and agreements could include business processes, asset swaps, etc.
2. **Set conditions:** Smart contracts could be initiated by parties themselves or when certain conditions are met like financial market indices, events like GPS locations, etc.
3. **Code business logic:** A computer program is written that will be executed automatically when the conditional parameters are met.
4. **Encryption and blockchain technology:** Encryption provides secure authentication and transfer of messages between parties relating to smart contracts.
5. **Execution and processing:** In blockchain iteration, whenever consensus is reached between the parties regarding authentication and verification then the code is executed and the outcomes are memorialized for compliance and verification.
6. **Network updates:** After smart contracts are executed, all the nodes on the network update their ledger to reflect the new state. Once the record is posted and verified on the blockchain network, it cannot be modified, it is in append mode only.

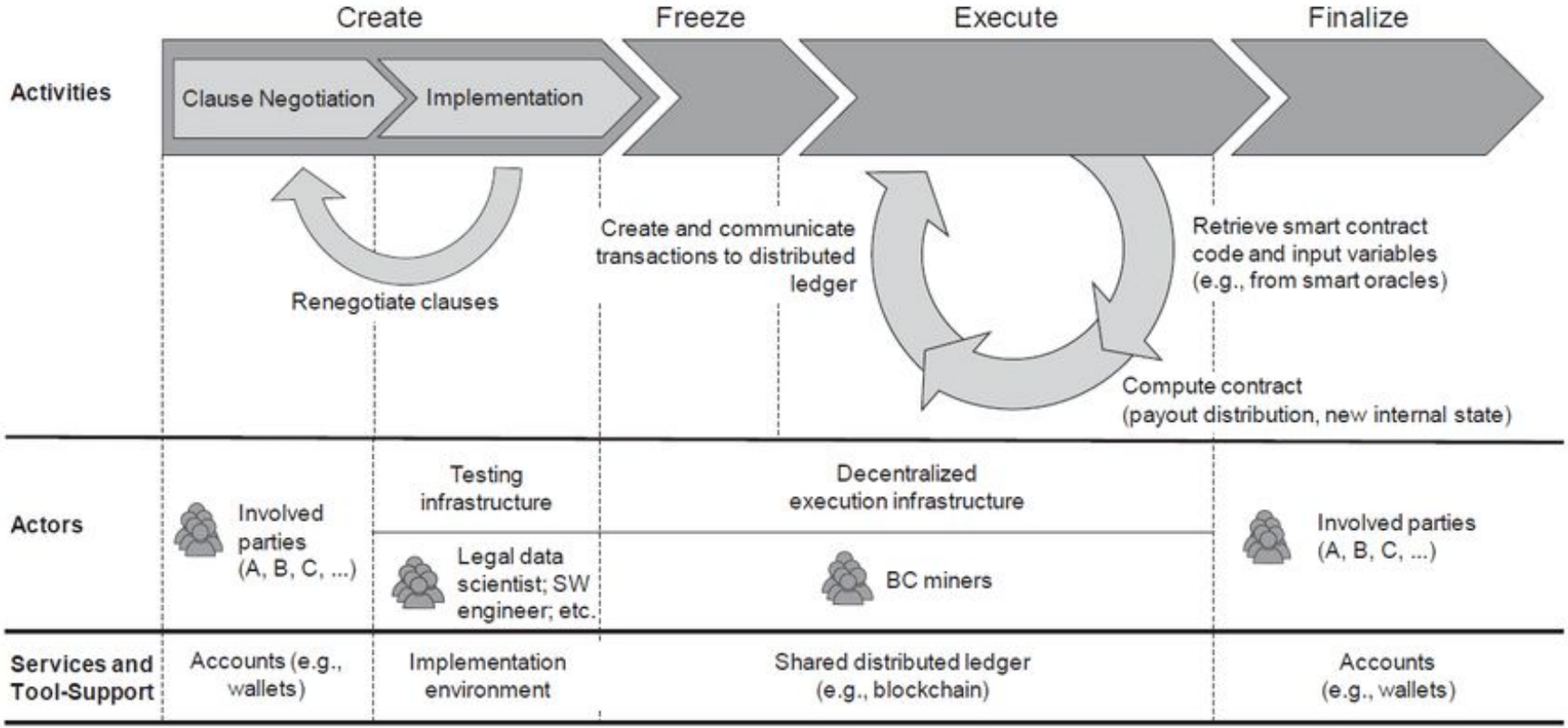
Courtesy : GeeksforGeeks

Steps involved in the Smart Contracts



Courtesy : Cointelegraph

Life Cycle of a Smart Contract



Courtesy : ResearchGate Paper

Life Cycle of a Smart Contract

1. **Creation Phase:**

- consists of iterative contract negotiation and an implementation phase.
- First, the parties must agree on the broad content and goals of the contract.
- Similar to typical contract negotiations and can be conducted online or in person contracts
- During this phase, the following tasks are completed:
    1. Multiple-party bargaining.
    2. Design, implementation, and validation of smart

2. **Freeze:**

- The validation of transactions on a blockchain is performed by a network of computers known as nodes. The blockchain miners are these nodes.
- To prevent the ecosystem from being swamped with smart contracts, miners must be paid a tiny fee in exchange for this service.

Courtesy : Medium

Life Cycle of a Smart Contract

3. **Execution:**

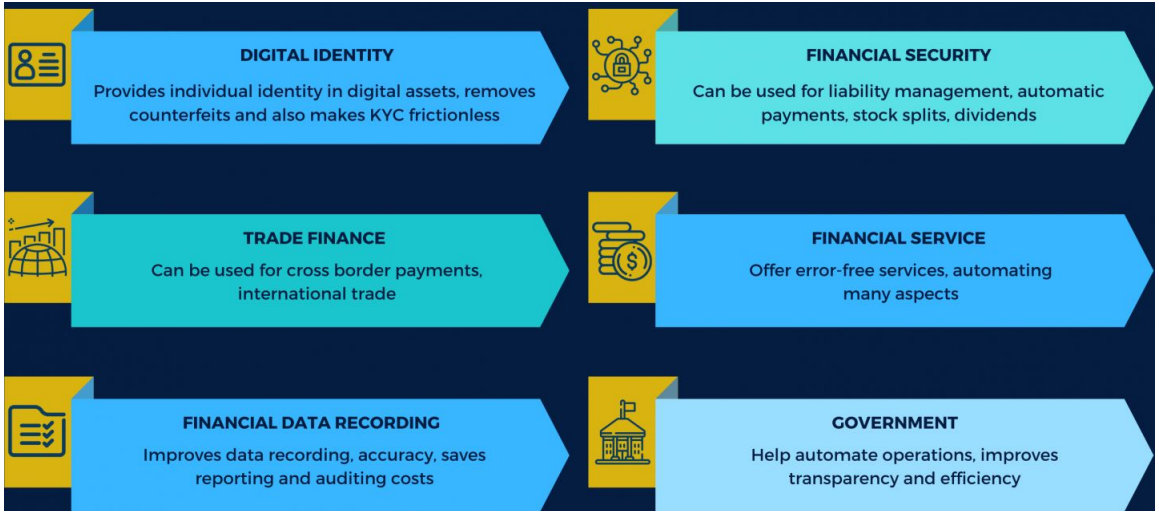
- Contracts placed on the distributed ledger are read by participating nodes.
- The authentication nodes validate the integrity of a smart contract,
- the code is performed by the smart contract interference engine (or by the compiler).
- When one party's inputs for execution are received in the form of coins (commitment to goods via coins), the interference engine **generates a transaction triggered by the met criterion**.
- The execution of the smart contract **results in a new set of transactions and a new state for the smart contract.**
- The **discoveries and new state information** are entered into the distributed ledger and validated using the consensus procedure.

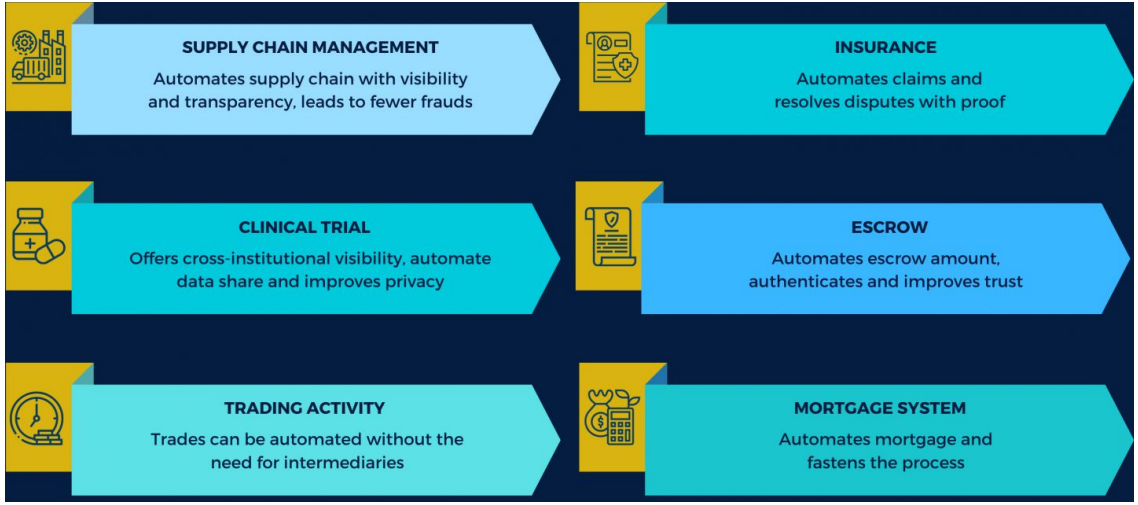
4. **Finalize:**

- the resulting transactions and updated state information are recorded in the distributed ledger and confirmed via the consensus process.
- The previously pledged digital assets are transferred (assets are unfrozen), and the contract is signed to confirm all transactions.

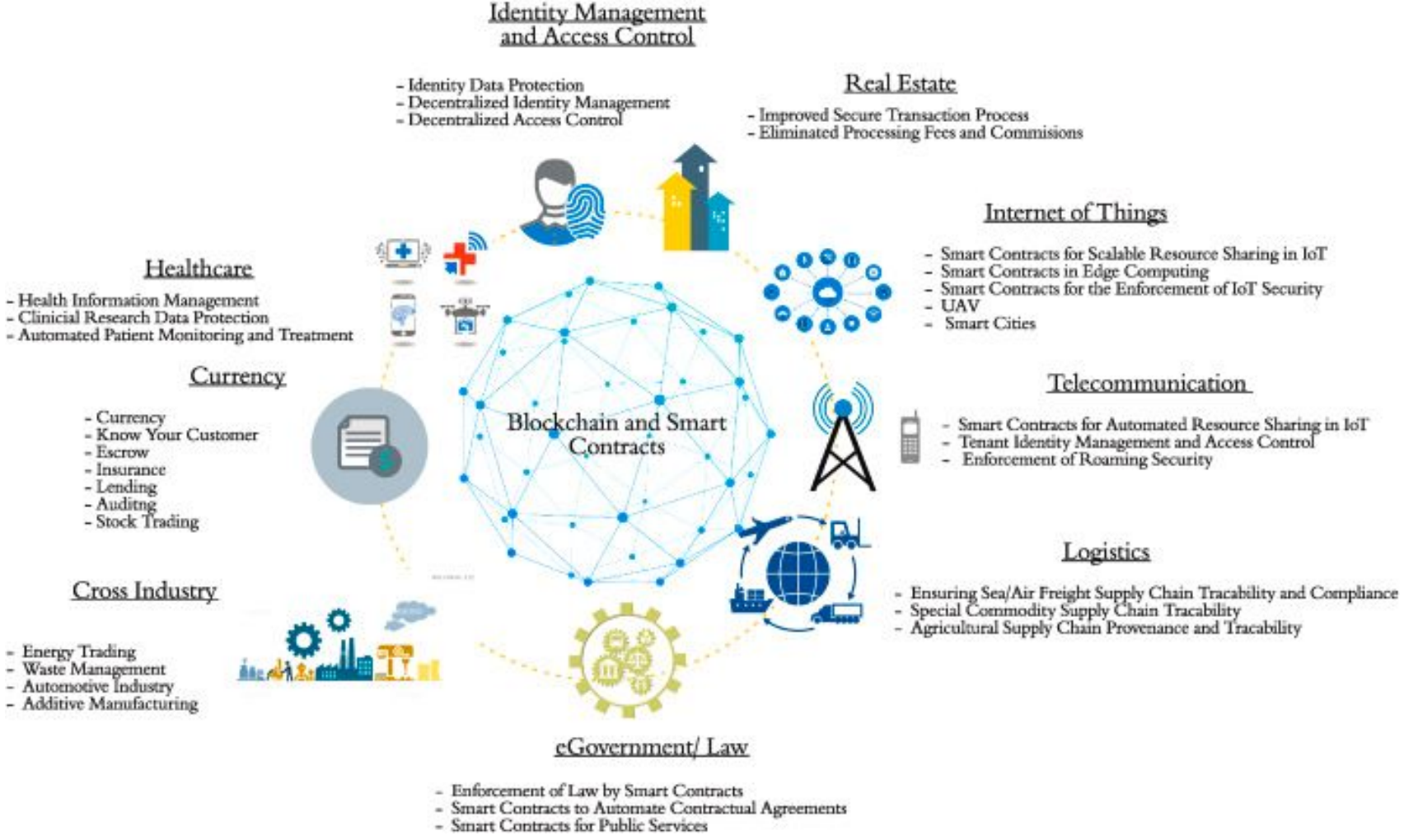
Courtesy : Medium

Use cases of Smart Contracts



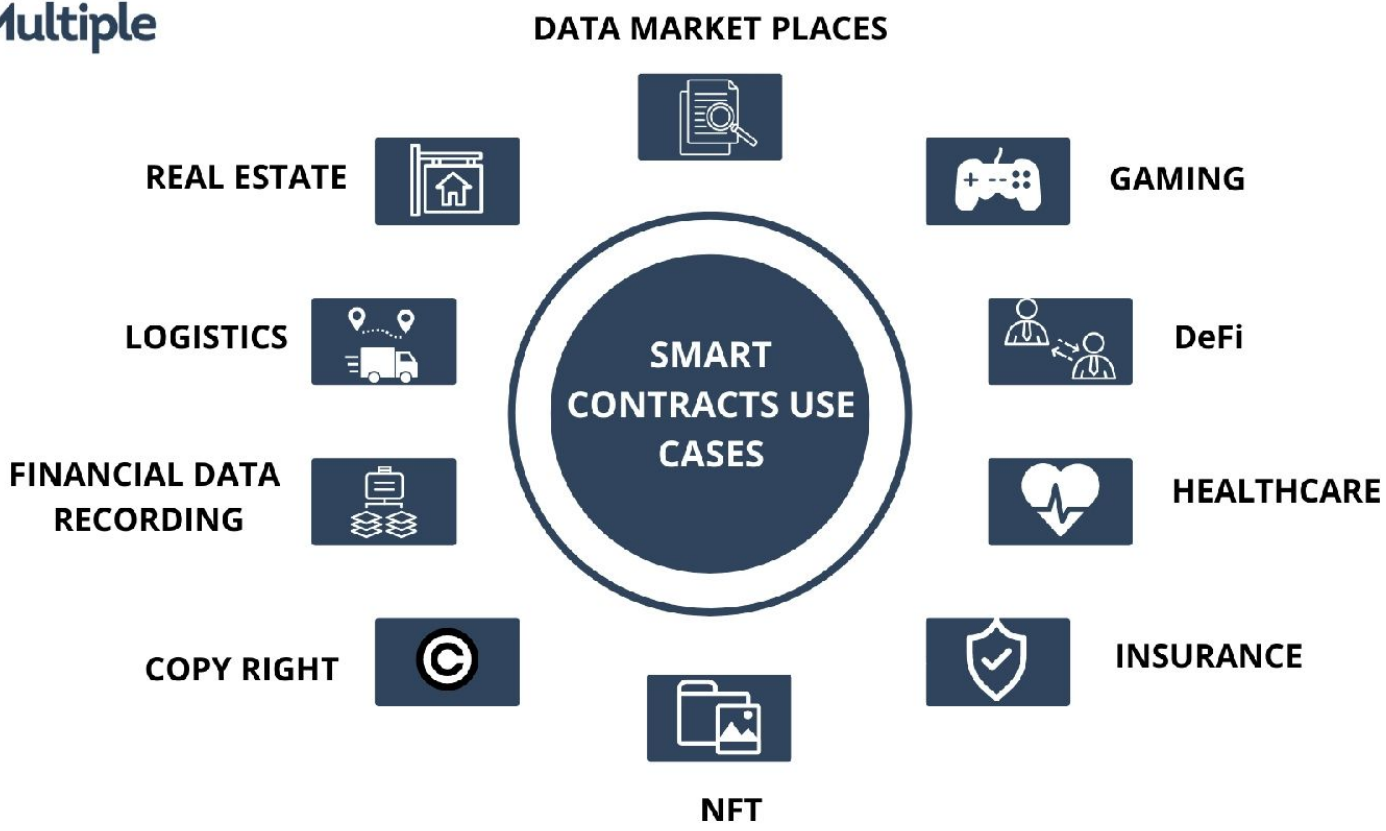


Courtesy : 101 Blockchains



Courtesy : Science Direct Paper

Top 10 Use cases of Smart Contract 2023



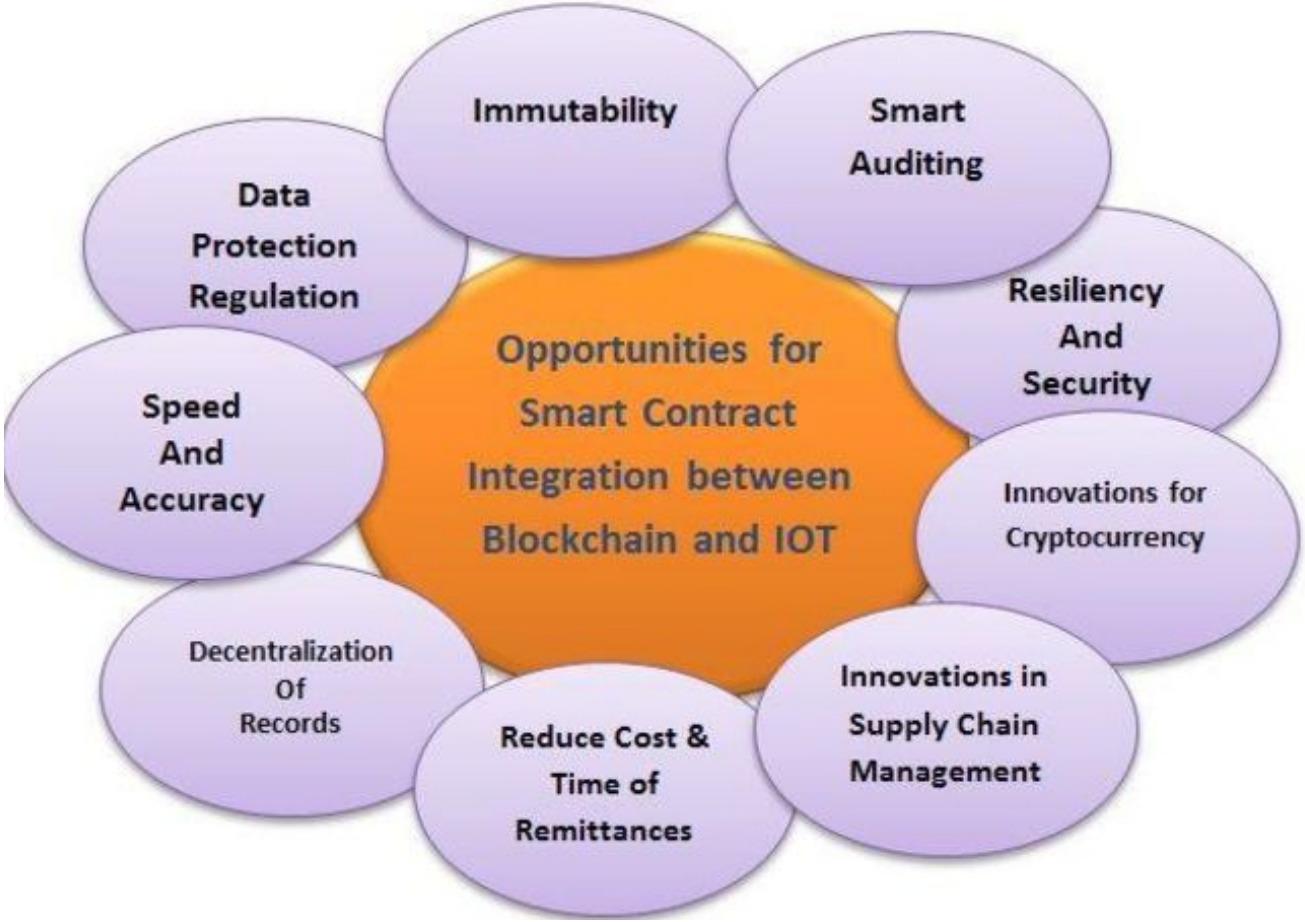
Courtesy : AI Multiple

Common Use cases of Smart Contract

- **DeFi**
  - Cryptocurrencies and smart contracts have allowed decentralized finance platforms to provide financial services without a need for a middleman.
  - DeFi had a total value locked of $94 billion by end of 2021.
  - DeFi has evolved to be more than just peer-to-peer transactions.
  - Smart contracts have enabled sophisticated transactions such as lending, borrowing, and derivative transactions on DeFi platforms.
  - For example:
    - AAVE is a DeFi platform that allows borrowing and lending in different cryptocurrencies.
    - Opyn is a DeFi derivative trading platform that utilizes smart contracts for options trading.
- **NFTs**
  - $17 billion worth of Non-fungible tokens (NFTs) were traded in 2021, making it one of the most impactful smart contract use cases.
  - Even though the market has cooled down in the 2nd quarter of 2022, NFTs have real-life use cases which can lead to long-term use of NFTs.
  - Smart contracts have enabled the creation of non-fungible tokens (NFTs) by allocating ownership and managing the transferability of NFTs. These contracts can also be modified to include additional features such as royalty payments and access rights to a platform or software.

Courtesy : AI Multiple

Smart Contract Opportunities



Courtesy : Semantics Scholar Paper

Smart Contract Opportunities

1. **Trust and Transparency:** Smart contracts operate on decentralized blockchain networks, ensuring trust and transparency in transactions. Parties involved can rely on the code's execution rather than trusting a centralized authority.
2. **Reduced Intermediaries:** By automating contract execution, smart contracts eliminate the need for intermediaries like banks, lawyers, or notaries, reducing costs and delays associated with traditional contracts.
3. **Efficiency:** Smart contracts execute automatically once predefined conditions are met. This reduces manual errors, speeds up processes, and lowers administrative costs.
4. **Security:** Blockchain technology provides a high level of security through encryption and consensus mechanisms. Once data is on the blockchain, it is challenging to alter, enhancing data integrity.
5. **Global Reach:** Smart contracts can be accessed and executed globally, enabling businesses to interact with international partners and customers seamlessly.
6. **Immutable Records:** Transactions recorded on a blockchain are tamper-resistant and cannot be altered once confirmed, ensuring a reliable and permanent record

Smart Contract Challenges

Courtesy : Semantics Scholar Paper

Smart Contract Risks

1. **Coding Errors:** Smart contracts are only as good as the code written to create them. Coding errors or vulnerabilities can lead to significant financial losses or breaches of privacy. The code must be thoroughly audited and tested.
2. **Regulatory Uncertainty:** The legal and regulatory framework for smart contracts varies by jurisdiction. The lack of clarity in some areas can lead to legal challenges or compliance issues.
3. **Irreversible Transactions:** Once a smart contract is deployed, its actions are irreversible. If there's a mistake or dispute, it can be challenging to resolve without external intervention.
4. **Oracles and External Data:** Smart contracts often rely on external data sources (oracles) to trigger actions. If these sources provide incorrect or manipulated data, it can lead to undesirable outcomes.
5. **Scalability:** Blockchain networks, especially Ethereum, face scalability challenges, resulting in slow transaction times and high fees during periods of high demand.
6. **Privacy Concerns:** While blockchain provides transparency, it may not be suitable for contracts that require complete privacy or confidentiality.
7. **Human Element:** Smart contracts can't account for all real-world scenarios. Human intervention may still be needed for complex negotiations or unforeseen circumstances.
8. **Upgrades and Forks:** Blockchain networks can undergo upgrades or forks, potentially impacting the functionality or compatibility of smart contracts.
9. **Cost of Deployment:** Deploying smart contracts on some blockchain networks can be costly due to gas fees (transaction costs) and development expenses.

What Smart Contracts does not promise to do ?

Ease of Correction

Cases of Loophole

Third Party Elimination

Legal Unclarity

Management of Vague Terms & Conditions

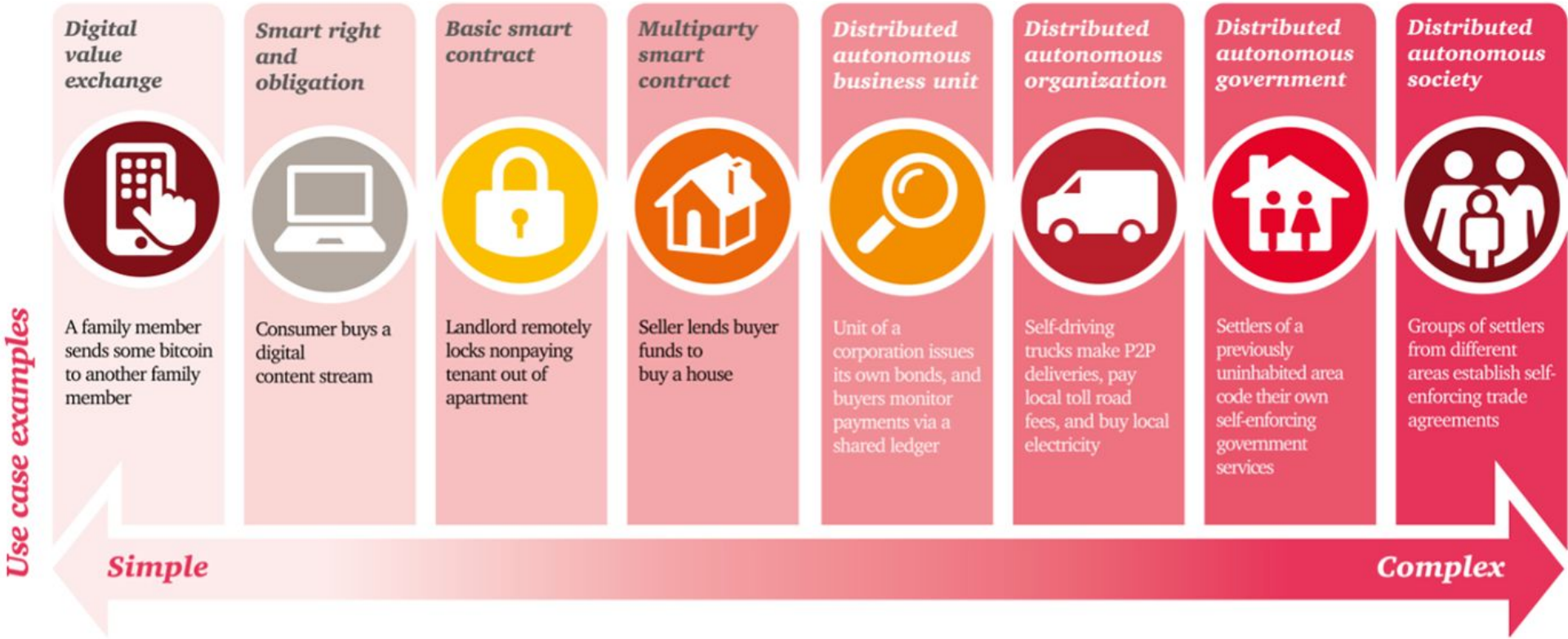
Courtesy : Appinventiv

Blockchain based Smart Contract Integration Platforms

Platform	Blockchain	Smart Contract Language	Consensus Protocol	Cryptocurrency	System Complexity	Scalability
BitCoin	Public & Private	IVY for Bitcoin Language	PoW	BTC	Medium	Block size 3-7 Tx/Sec
Ethereum	Public & Private	Solidity	PoS	Ether (ETH)	High	Block size 5-20 Tx/Sec
HyperLedger Fabric	Private	Java, Node.js and Go	PBFT/SIEVE	None	High	Block size 100-3000 Tx/Sec
NEM	Public	Java and Node.js	PoI	XEM	Medium	Block size 1,000-10,000 Tx/Sec
Stellar	Public	Stellar SDK & Go	PBFT / FBA	Lumens (XLM)	High	Block size 1,000-1,500 Tx/Sec
Waves	Public	RIDE	LPoS	WAVES	Medium	Block size 100 Tx/Sec
Lisk	Public	Lisk JScript	DPoS	LSK	Medium	Block size 25Tx/Sec
NXT	Public	TC Script	PoS	NXT	Medium	Block size 5-20 Tx/Sec
Monax	Private	Monax SDK and Solidity	PoS	MultiAsset	High	-
Qtum	Public	QSCL and Solidity	PoS	QTUM	Medium	Block size 70-140Tx/Sec

Courtesy : Semantics Scholar Paper

Smart Contracts by Complexity Level



Courtesy : LinkedIn - Marvin Fedler

Solidity Programming

- Refer the pdf attached Solidity Quick Guide here for Programming basics
- Remix IDE - LearnETH platform and complete the following courses during this week

1. Solidity Beginner Course (19 Assignments) -
2. Solidity ERC20 Token Course (6 Assignments) - Learn to understand and create ERC20 tokens.
3. Solidity NFT Course (6 Assignments) - Learn how to create your own NFTs (non-fungible tokens).
4. NFT Auction Contract (3 Assignments) - Learn how to write an ERC721 (NFT) auction contract.

ERC Standards

WHAT ARE ERC STANDARDS?

Ethereum Request for Comments (ERC) is a document that smart contract programmers are using in the Ethereum blockchain platform to write. These are rules that the Ethereum-based tokens must comply with.

CREATION PROCESS

1. Developers submit new ERC policies through Ethereum Improvement Proposal
2. Ethereum community reviews these documents
3. The community may ask for revision or comment in favor or oppose
4. After acceptation, the developers implement it

Courtesy : 101 Blockchain - ERC Standards

ERC20

ERC Standards

DIFFERENT ETHEREUM REQUEST FOR COMMENTS

ERC 20

- Most popular token standard.
- Used in most of the ICOs.
- Fungible token standard.
- Allows the implementation of standard API within a smart contract.

ERC 721

- A standard for non-fungible tokens.
- Allows the implementation of standard API for NFTs within a smart contract.
- Offers functionality to transfer and track NFTs.

ERC 165

- A standard for a method, instead of tokens.
- Covers how interfaces are identified.
- States how any contract can publish the interfaces after the implementation.
- States how to detect when a contract implements ERC-165.
- Covers the way to detect when a smart contract uses any given interface.

ERC 777

- Reduces friction in crypto transactions.
- Not in use, still in EIP phase.
- Gets rid of the double transaction verification of ERC 20.
- Lowers transaction overhead.
- Allows users to reject incoming tokens from a blacklisted address.

Courtesy : 101 Blockchain - ERC Standards

ERC20

ERC Standards

ERC 223

- Prevents accidental burns of tokens, a bug in ERC 20.
- Developers can either accept or decline tokens arriving at their smart contract addresses.
- Rejected transactions will fail but won't burn the tokens.
- Not in use, still in EIP phase.

ERC 827

- An extension of ERC 20.
- Wallets and exchanges can reuse tokens.
- Token holders can transfer token while also approving a 3rd party to spend it.
- Not in use, still in EIP phase.

ERC 621

- An extension to ERC 20 standard
- Uses two functions -'increaseSupply' and 'decreaseSupply.'
- Can increase or decrease the token supply.
- Not in use, still in EIP phase.

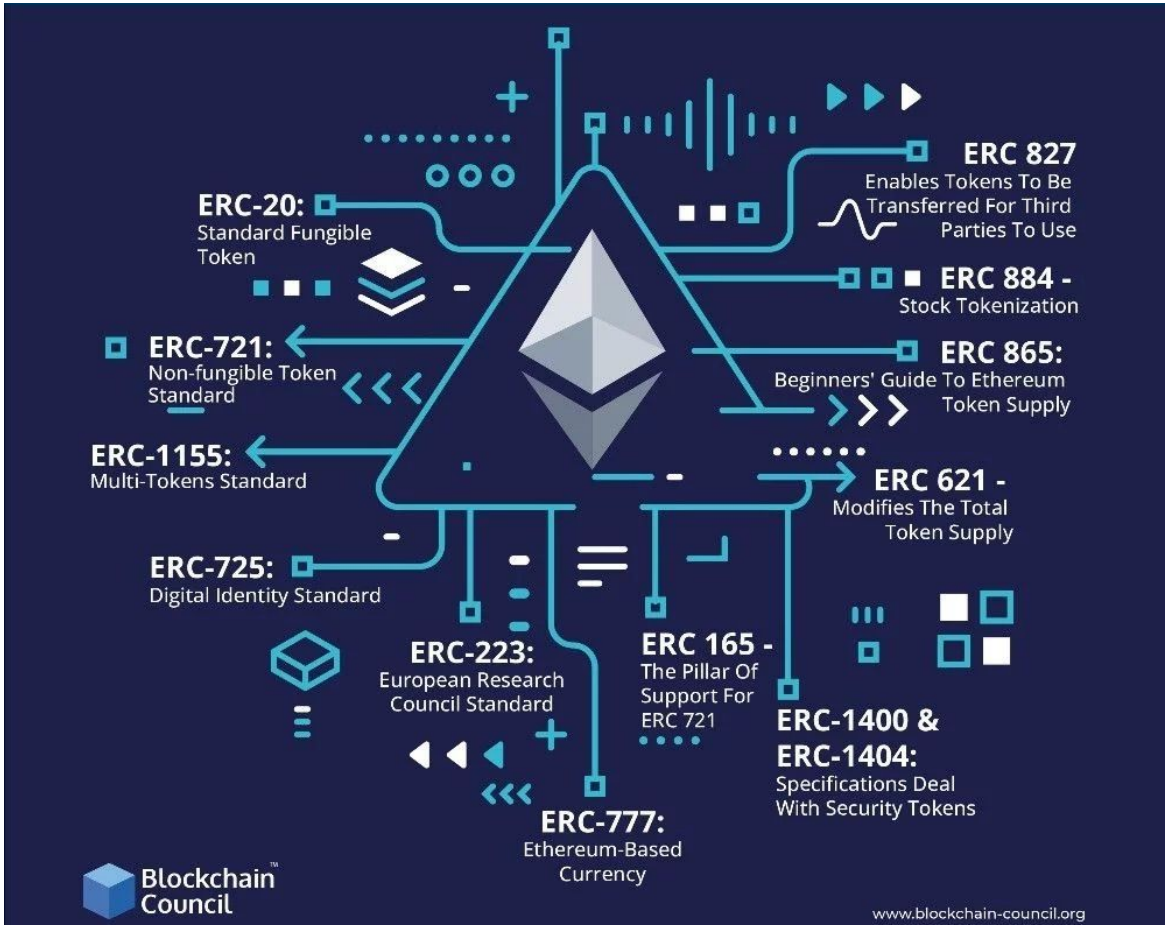
ERC 884

- Allows companies to use blockchain to maintain share registries.
- Identity verification and mandatory whitelisting of token holders.
- Only whole value of tokens, i.e., no partial value.
- Recording of information regulators mandate.
- Not in use, still in EIP phase.

Created by 101blockchains.com

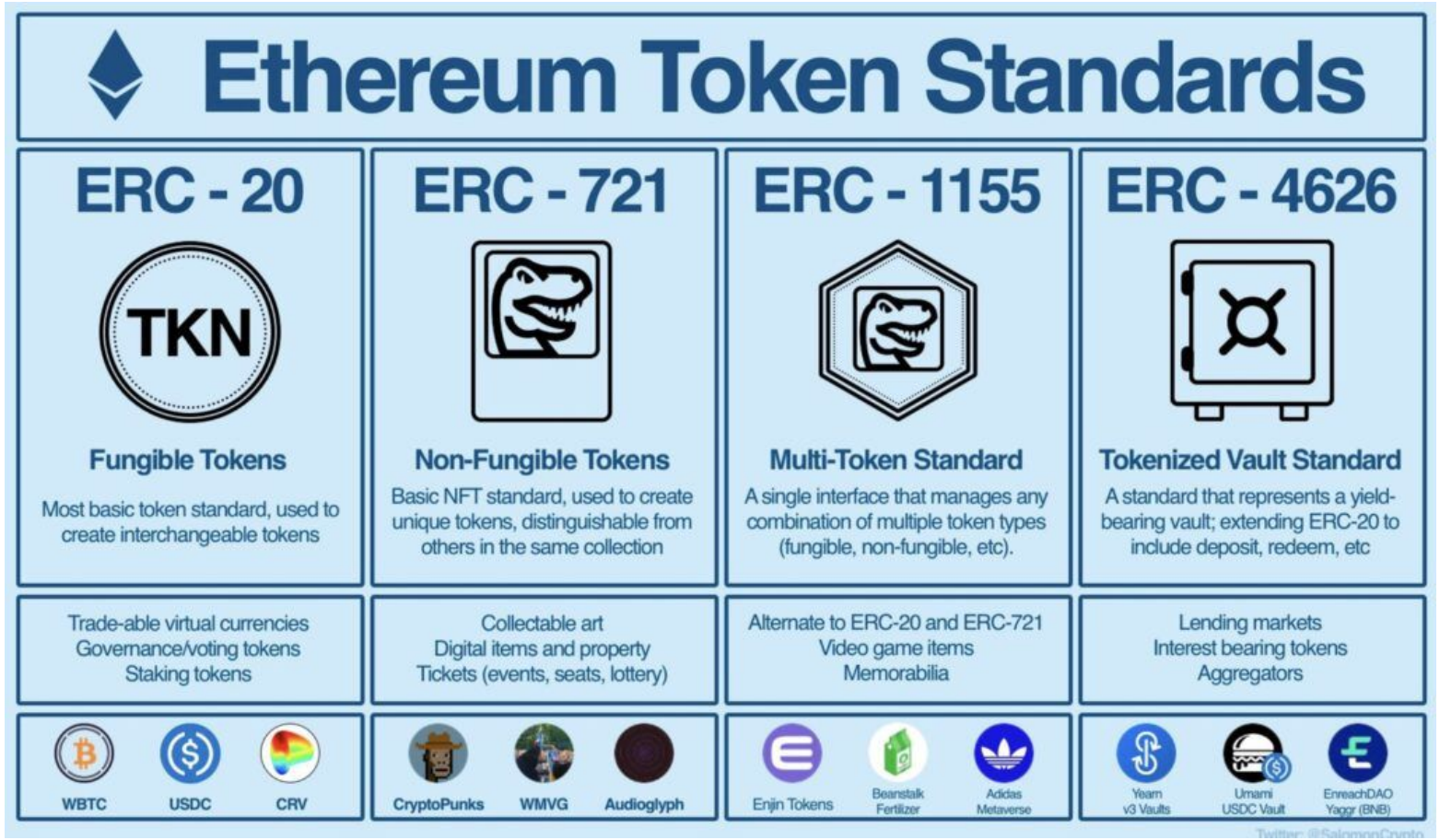
Courtesy : 101 Blockchain - ERC Standards

ERC Token Standards List



Courtesy : [Blockchain Council - ERC Token Standards]

ERC Token Standards List



Courtesy : [Salomon - ERC Token Standards]

ERC20 Token

- ERC-20 is the **technical standard for fungible tokens created using the Ethereum blockchain.**
- A **fungible token is interchangeable with another token**—where the well-known non-fungible tokens (NFTs) are not interchangeable.
- ERC-20 allows developers **to create smart-contract-enabled tokens that can be used with other products and services.**
- These **tokens are a representation of an asset, right, ownership, access, cryptocurrency, or anything else that is not unique in and of itself but can be transferred.**
- A **list of functions and events** that must be implemented into a token for it to be considered ERC-20 compliant.
    - **functions** describe **what must be included in the smart-contract-enabled token,**
    - **events** describe **an action.**
- ERC-20 guides the creation of new tokens on the Ethereum blockchain so that they are interchangeable with other smart contract tokens.

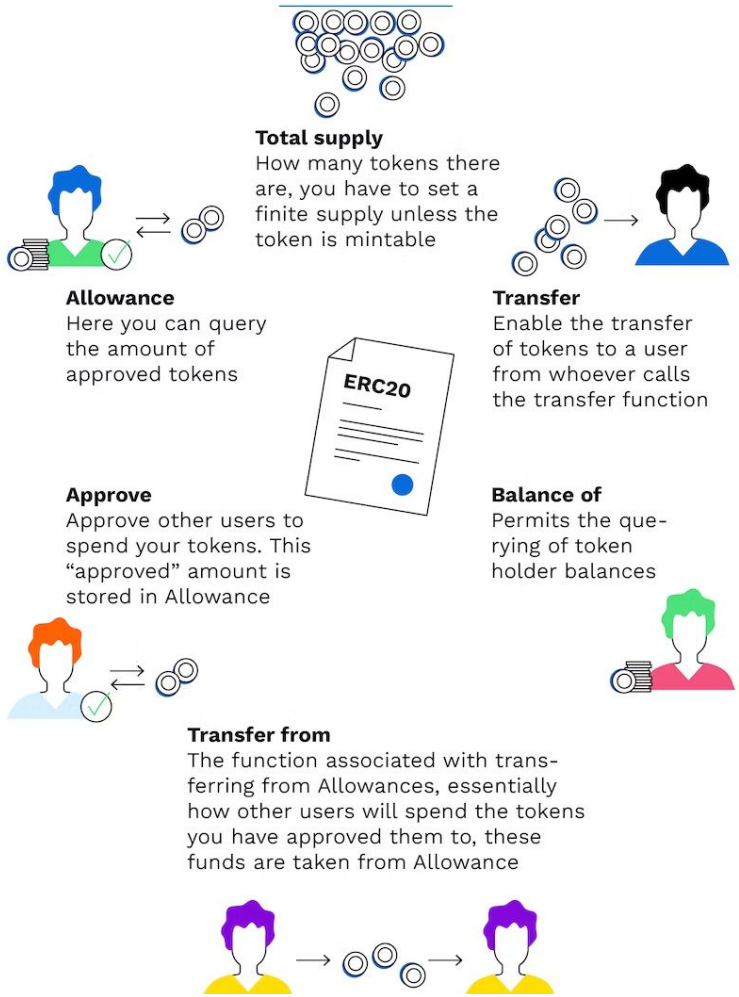
Courtesy : Investopedia - ERC 20

ERC20 Token Features

Courtesy : [Coinsclone - ERC 20]

ERC20 Token Functions

1. **TotalSupply:** The total number of tokens that will ever be issued
2. **BalanceOf:** The account balance of a token owner's account
3. **Transfer:** Automatically executes transfers of a specified number of tokens to a specified address for transactions using the token
4. **TransferFrom:** Automatically executes transfers of a specified number of tokens from a specified address using the token
5. **Approve:** Allows a spender to withdraw a set number of tokens from a specified account, up to a specific amount
6. **Allowance:** Returns a set number of tokens from a spender to the owner



Courtesy : Investopedia - ERC 20
Bitpanda - ERC 20

ERC20 Token

The events that must be included in the token are:

1. **Transfer:** An event triggered when a transfer is successful
2. **Approval:** A log of an approved event (an event)

The following functions are optional and are not required to be included, but they enhance the token's usability:

1. Token's name (optional)
2. Its symbol (optional)
3. Decimal points to use (optional)

Courtesy : Investopedia - ERC 20

ERC20 Token

What is the significance of ERC-20 tokens?

ERC-20 tokens have become the standard for creating new tokens on the Ethereum Blockchain. This standardization makes it easier for developers to create new tokens and for users to interact with them, as they all follow the same set of rules. Additionally, ERC-20 tokens have enabled the creation of decentralized applications (dApps) on the Ethereum network, which has the potential to revolutionize industries.

What are the risks associated with ERC-20 tokens?

Like all cryptocurrencies, ERC-20 tokens are volatile and their value can fluctuate rapidly. Additionally, there is a risk of fraud, scams, and theft. It's important to do your research and be cautious when investing in ERC-20 tokens or any other cryptocurrency.

Courtesy : Blockchain Council

ERC20 Token

How do ERC-20 tokens differ from other cryptocurrencies?

ERC-20 tokens are built on the Ethereum Blockchain and follow a standardized set of rules and guidelines. This makes it easier for developers to create new tokens and for users to interact with them. Other cryptocurrencies may have different rules and guidelines, making them more complex to work with.

How can I use ERC-20 tokens?

ERC-20 tokens can be used for a variety of purposes, including investment, trading, and participating in decentralized applications. Some ERC-20 tokens can also be used for purchasing goods and services, although this is less common.

Courtesy : Blockchain Council

ERC20 Token

Popular Digital currencies use the ERC-20 standard.

- Tether USD (USDT)
- USD Coin (USDC)
- Shiba Inu (SHIB)
- Binance USD (BUSD)
- BNB (BNB)
- DAI Stablecoin (DAI)
- HEX (HEX)
- Bitfinex LEO (LEO)
- MAKER (MKR)

Courtesy : Investopedia - ERC 20
Blockchain Council

ERC20

ERC20 Token - Advantages

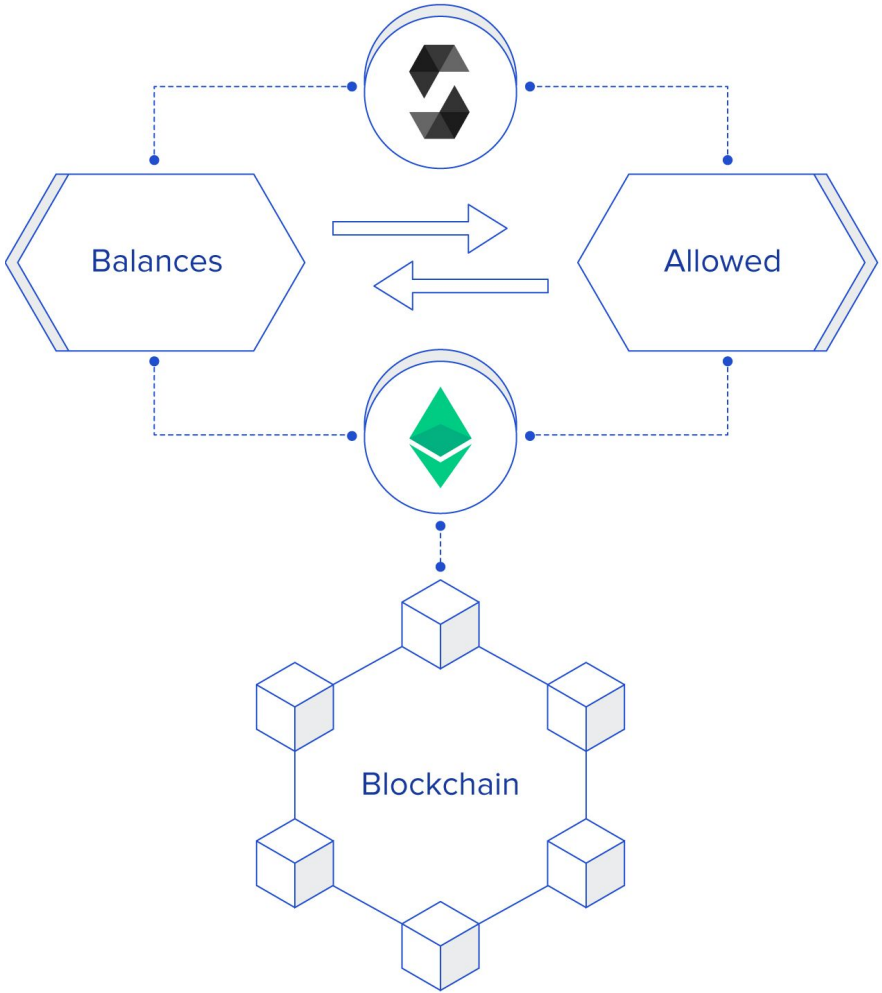
1. **Interoperability:** ERC-20 tokens adhere to a standardized set of rules, making them compatible with a wide range of wallets, exchanges, and applications. This interoperability fosters easy integration and widespread use across different platforms.
2. **Ease of Development:** Creating ERC-20 tokens is relatively straightforward due to the well-defined standard. This simplicity has lowered the barrier for developers to create their own tokens and launch initial coin offerings (ICOs) or token sales.
3. **Widespread Adoption:** The ERC-20 standard has gained widespread adoption within the Ethereum ecosystem. Many tokens, including some of the most well-known cryptocurrencies and utility tokens, are based on the ERC-20 standard.
4. **Liquidity and Trading:** ERC-20 tokens can be easily traded on various decentralized and centralized exchanges. Their widespread availability has contributed to liquidity and trading opportunities for token holders.
5. **Token Integration:** ERC-20 tokens can be integrated into various decentralized applications (dApps), allowing developers to build applications that utilize tokens for specific use cases, such as voting, rewards, governance, and more.
6. **Token Standards Evolution:** The success of ERC-20 paved the way for the development of other token standards, such as ERC-721 (non-fungible tokens) and ERC-1155 (multi-token standard), catering to different token use cases and enhancing the Ethereum ecosystem's versatility.

ERC20 Token - Disadvantages

1. **Limited Functionality:** ERC-20 tokens are primarily suited for fungible assets. If a project requires more complex features, such as non-fungible tokens (NFTs) or unique attributes for each token, ERC-20 might not be the ideal choice.
2. **Scalability and Gas Fees:** The Ethereum blockchain has faced scalability challenges, leading to network congestion and high gas fees during periods of heavy usage. This can impact the cost and speed of transactions involving ERC-20 tokens.
3. **Lack of Native Support for Advanced Features:** ERC-20 tokens lack built-in support for certain features, such as royalties, more intricate tokenomics, and on-chain interactions that other token standards might offer.
4. **Security Risks:** Although the ERC-20 standard provides a framework for secure token development, improper coding practices or vulnerabilities can still expose tokens to potential security risks and vulnerabilities.
5. **Lost Tokens:** ERC-20 tokens can be accidentally sent to the wrong addresses or lost due to mistakes in handling private keys. Unlike centralized systems, there's no straightforward way to recover lost tokens.
6. **Regulatory Concerns:** Due to the ease of creating tokens using the ERC-20 standard, there have been instances of fraudulent token sales or scams. This has led to regulatory scrutiny in some cases.

ERC20 Token

Writing an ERC20 Token in Solidity



Courtesy : Toptal - ERC 20

ERC 721 Token

- An ERC-721 token is a **type of non-fungible token (NFT) standard** on the Ethereum blockchain.
- ERC-721 tokens are **unique and distinct from each other.**
- Each ERC-721 token has **individual properties, attributes, or characteristics,** making them ideal for representing ownership of one-of-a-kind digital or physical assets.
- The term "ERC-721" stands for "**Ethereum Request for Comment 721**," and
- it refers to the **standard that defines the rules and functions for creating and managing non-fungible tokens on the Ethereum platform.**
- This standard was introduced **to enable the creation of digital assets that can represent collectibles, virtual items, artwork, real estate, in-game items,** and more, where each token holds unique value.

ERC 721 Token Characteristics

1. **Uniqueness:**
   - Each ERC-721 token has a distinct identifier, and no two tokens are the same. This allows for the representation of rare or unique items that hold individual value.
2. **Properties and Metadata:**
   - ERC-721 tokens can include metadata, attributes, and additional information that provide context and value to the token. For example, a digital artwork NFT could include the artist's name, creation date, and other relevant details.
3. **Ownership and Transfers:**
   - Ownership of an ERC-721 token is recorded on the blockchain. Tokens can be transferred between addresses using the standard transferFrom function. Ownership changes are transparent and verifiable on the blockchain.
4. **Use Cases:**
   - ERC-721 tokens have a wide range of use cases, including digital art ownership, virtual collectibles, unique in-game items, domain names, real estate representation, identity verification, and more.

ERC 721 Token Characteristics

5. **Standard Interfaces:**
   - ERC-721 defines a set of standard functions for interacting with tokens, including querying ownership, transferring ownership, and querying token metadata.
6. **Decentralized Applications (dApps):**
   - Developers can build decentralized applications that utilize ERC-721 tokens. These dApps can create marketplaces, games, platforms for trading collectibles, and other innovative services that leverage the uniqueness of NFTs.
7. **Ecosystem Growth:**
   - The popularity of ERC-721 tokens has led to the growth of NFT marketplaces and platforms, where users can buy, sell, and trade NFTs. These tokens have gained significant attention, especially in the realms of art, entertainment, and gaming.
8. **Gas Costs and Scalability:**
   - Transacting with ERC-721 tokens can be more gas-intensive than ERC-20 tokens due to the uniqueness and additional metadata associated with each token. Scalability remains a consideration, especially during periods of high network congestion.

Comparison between ERC20 & ERC721

Criteria	ERC-20	ERC-721
Fungibility	Fungible in nature.	Non-fungible in nature.
Token Identity	There is no specific disparity among the different tokens.	Each token has a specific identity and could be easily distinguished.
Collecting Tokens	ERC-20 tokens are not collectible.	You can collect ERC-721 tokens like fiat currency.
Value Fluctuation	The value of ERC-20 tokens remains the same.	The value of ERC-721 tokens fluctuates according to rarity and uniqueness.
Adoption	Commonly adopted.	Limited levels of acceptance.
Substitutes	Easier for substitution.	No scope for substitution.
Divisibility	Can be divisible into decimals.	ERC-721 tokens are not divisible.
Ownership Functions	No special ownership functions are allocated.	ERC-721 tokens can enable special ownership functions.

Courtesy : 101 Blockchains

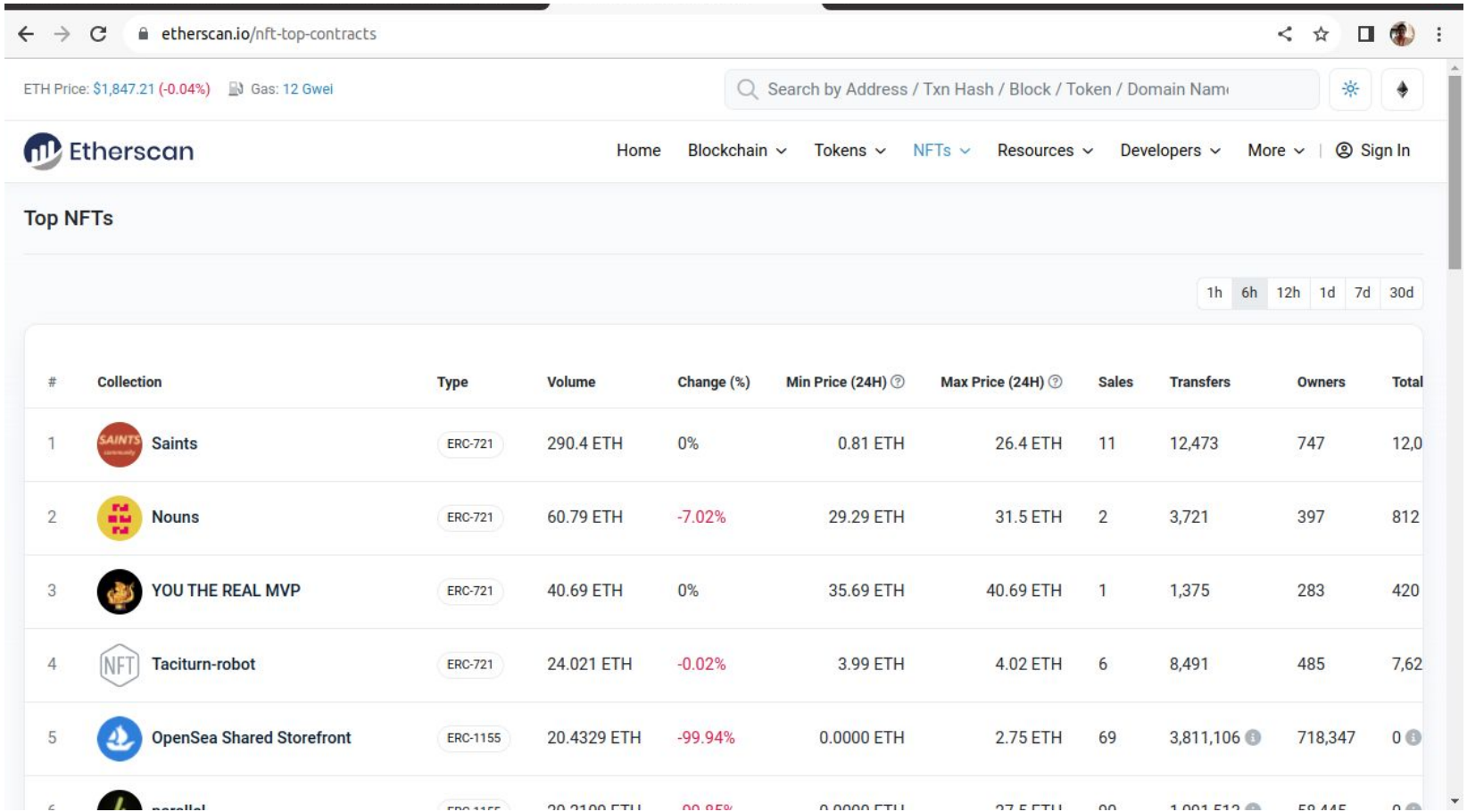
Comparison between ERC20 & ERC721

Aspect	ERC-20 Tokens	ERC-721 Tokens
Type	Fungible tokens (identical and interchangeable)	Non-fungible tokens (unique and distinct)
Individuality	Tokens are identical, no unique attributes	Each token has unique attributes or properties
Use Cases	Currency, utility tokens, rewards, ICOs	Digital collectibles, gaming items, unique assets
Transfers	Can be transferred on a one-to-one basis	Transfers may involve individual token properties
Standard	Single standard (ERC-20)	Single standard (ERC-721)
Functions	Basic functions: transfer, balanceOf, approve, etc.	Advanced functions for managing unique tokens
Properties	No inherent properties or metadata	Tokens can have metadata, name, and attributes

Comparison between ERC20 & ERC721

Aspect	ERC-20 Tokens	ERC-721 Tokens
Interoperability	High interoperability across platforms	May require specialized interfaces for certain apps
Token Identifiers	Token IDs are not unique across contracts	Token IDs are unique within a contract
Example	Ethereum (ETH), Chainlink (LINK)	CryptoKitties, Decentraland LAND
Scalability	Scalable for large quantities of tokens	Potentially less scalable due to unique properties
Gas Costs	Lower gas costs due to standardized operations	Higher gas costs for more complex transfers
Complexity	Simpler to implement	More complex due to individual token attributes
Registries	Not required; tokens can exist independently	Often use token registries for unique IDs

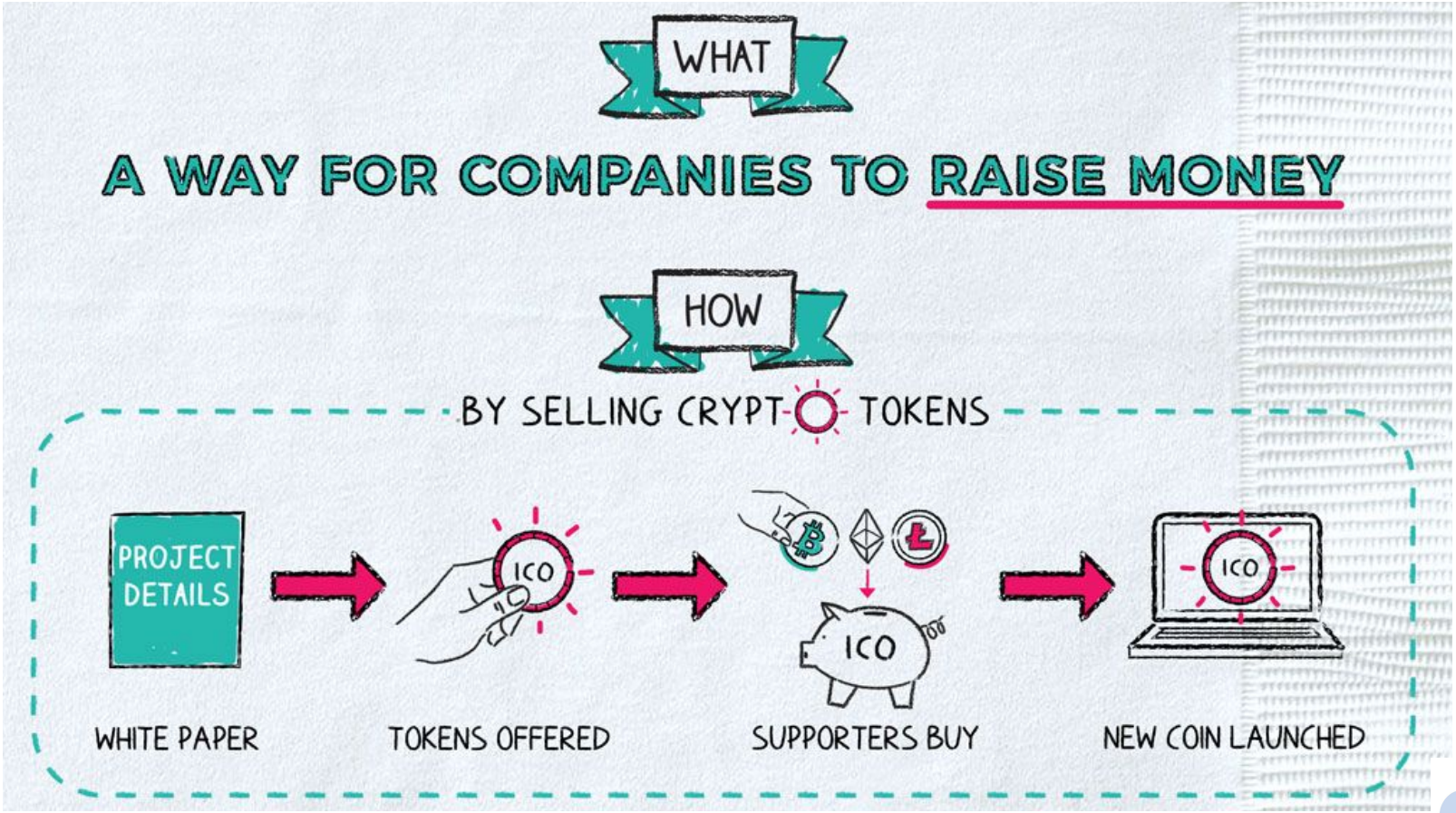
Exploring etherscan.io - top NFT's



#	Collection	Type	Volume	Change (%)	Min Price (24H)	Max Price (24H)	Sales	Transfers	Owners	Total
1	Saints	ERC-721	290.4 ETH	0%	0.81 ETH	26.4 ETH	11	12,473	747	12,0
2	Nouns	ERC-721	60.79 ETH	-7.02%	29.29 ETH	31.5 ETH	2	3,721	397	812
3	YOU THE REAL MVP	ERC-721	40.69 ETH	0%	35.69 ETH	40.69 ETH	1	1,375	283	420
4	Taciturn-robot	ERC-721	24.021 ETH	-0.02%	3.99 ETH	4.02 ETH	6	8,491	485	7,62
5	OpenSea Shared Storefront	ERC-1155	20.4329 ETH	-99.94%	0.0000 ETH	2.75 ETH	69	3,811,106	718,347	0

Courtesy : Etherscan.io

Initial Coin Offerings



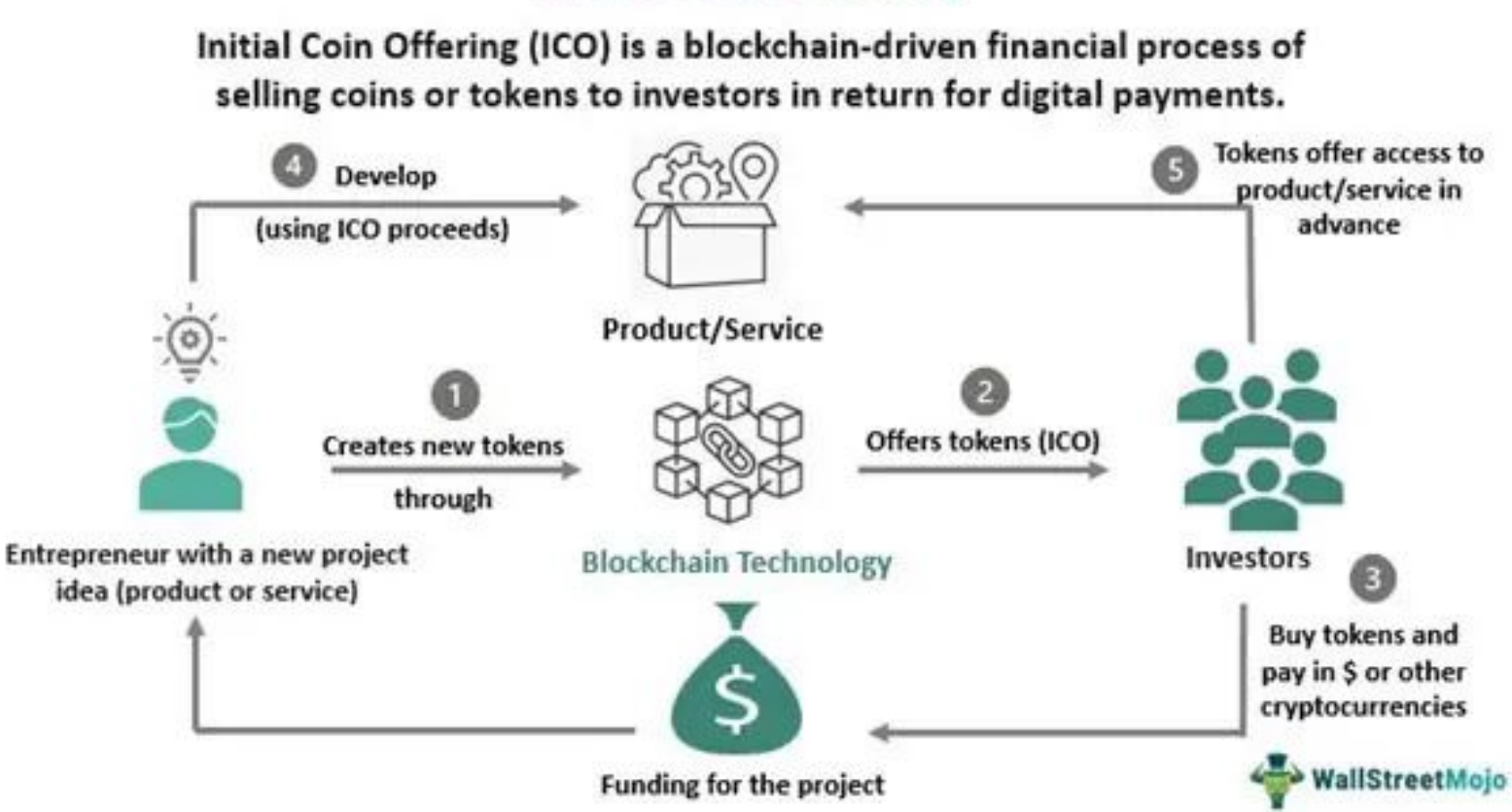
Courtesy : Napkin Finance - ICO

Initial Coin Offerings

- An Initial Coin Offering (ICO) is a fundraising method used by cryptocurrency startups to raise capital by issuing new tokens to investors.
- These tokens are often built on blockchain platforms like Ethereum and can serve various purposes within the project's ecosystem.
- ICOs became popular in the cryptocurrency space as a way to fund the development of new blockchain projects and applications.

Courtesy : Banner ICO

Working of an ICO - Initial Coin Offerings



Courtesy : Wall Street Mojo - ICO

Working of an ICO - Initial Coin Offerings

1. **Develop Project Idea:** The process begins with a cryptocurrency startup having an idea for a new blockchain project or application.
2. **Create New Tokens:** The startup creates new tokens on a blockchain platform like Ethereum. These tokens represent ownership or utility within the project.
3. **Market the ICO:** The startup promotes the ICO through various marketing and communication channels to attract potential investors.
4. **Investors Buy Tokens:** Interested investors purchase the newly created tokens using cryptocurrencies (e.g., Bitcoin or Ethereum) during the ICO.
5. **Funds Raised in Crypto:** The funds raised from investors are typically held in cryptocurrency, which can be used to fund the project's development.
6. **Use Funds for Development:** The startup utilizes the raised funds to develop the project or application as outlined in the ICO's whitepaper.
7. **Tokens Listed on Exchanges:** After the ICO, the project's tokens are often listed on cryptocurrency exchanges, making them tradable.
8. **Token Trading and Liquidity:** Once listed on exchanges, the tokens can be bought, sold, and traded by investors, creating liquidity in the market.

Initial Coin Offering Example

- Two of the most popular and profitable ICOs
  - Ethereum **(CRYPTO: ETH)**
  - **Solana (CRYPTO: SOL)**
- In 2022 is NAGA (CRYPTO: NGC) is gaining popularity.
  - NAGA is a **German fintech company**
  - offering an i**nvestment app with a social network for traders.**
  - It allows traders to copy and follow popular traders and, as a result, learn and grow.
  - The NGC token was launched in December 2017 in partnership with the NAGA Group post raising funds worth $50 million as part of an ICO.
  - The cryptocurrency is valued at a market cap of $103 million,
  - Price of 1 NGC token → $1.31 in 2022

Courtesy : Wall Street Mojo - ICO

Initial Coin Offerings Types

- An ICO regulator may either classify the token offering
  - **As a service**
    - an ICO requires issuing a whitepaper detailing the nature of the business, its structure, purpose, objectives, etc.
  - **As a security for trade**
    - an ICO needs to undergo proper registration and meet other regulatory requirements.
    - Eg : ICOs are registered with and regulated by the U.S. Securities and Exchange Commission

- ICO is classified into two types
  - Private ICO
    - can only have a small number of investors, mostly accredited ones.
    - The accredited investors can either be a financial institution / a high net-worth individual.
    - They can participate in the ICOs with the company offering them a set price.
  - Public ICO
    - refers to crowdfunding
    - Open to all.
    - Anyone around can be an investor.

Note : Given the safety and regulatory concerns, private ICOs are popular than public counterparts.

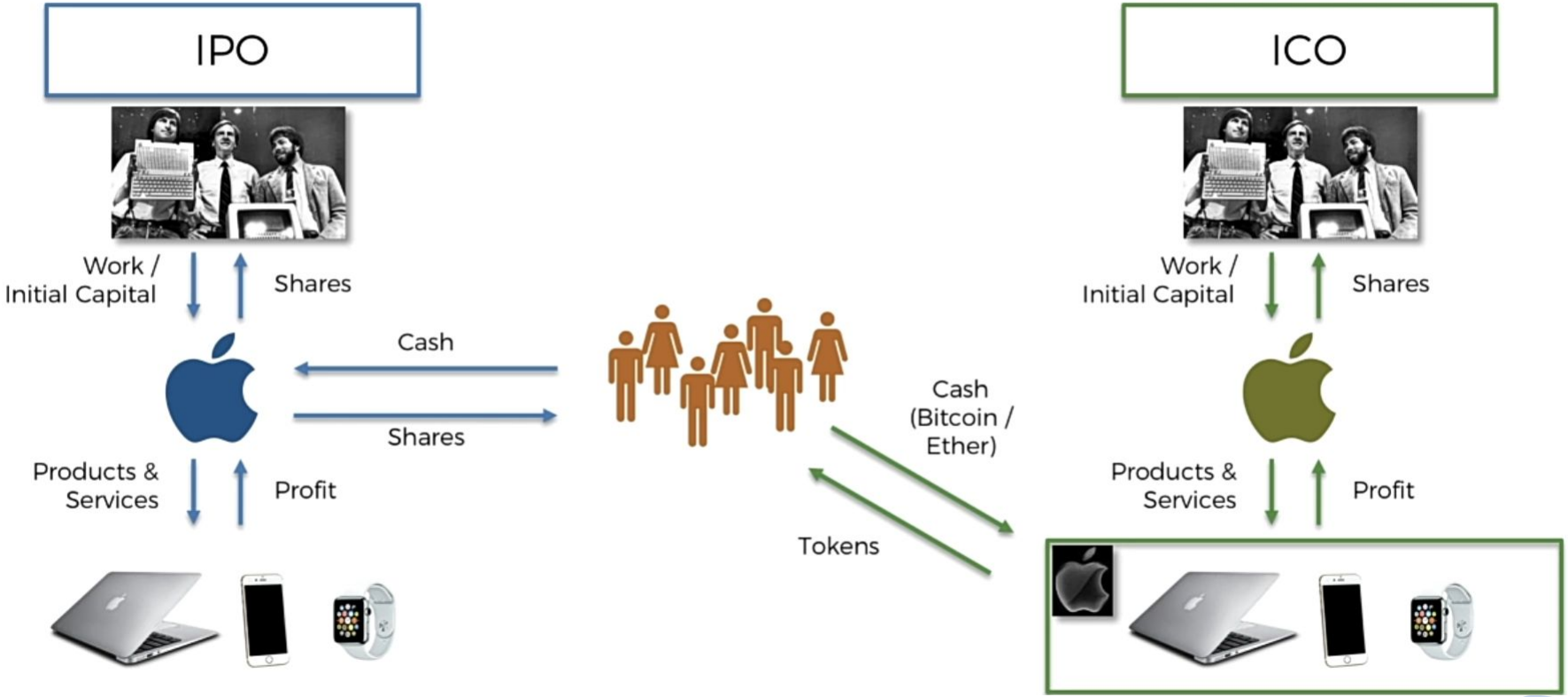
Courtesy : Wall Street Mojo - ICO

Initial Coin Offerings : ICO Vs IPO

Particulars	ICO	IPO
Full Form	Initial Coin Offering	Initial Public Offering
Purpose	Raise funds through cryptocurrencies	Raise capital by selling securities
Platform	Blockchain platform	Primary stock markets
Offers	Tokens or coins in exchange for other cryptocurrencies or legal currencies	Company shares in exchange for payment
Regulation	Mostly unregulated	SEC regulated
Risk Involved	High as no finished product to show for guarantee	Low risk as investors have finished products or projects to assess before deciding whether to invest
Stake	Tokens provide access to the company's new product or service	Securities offer ownership in the company

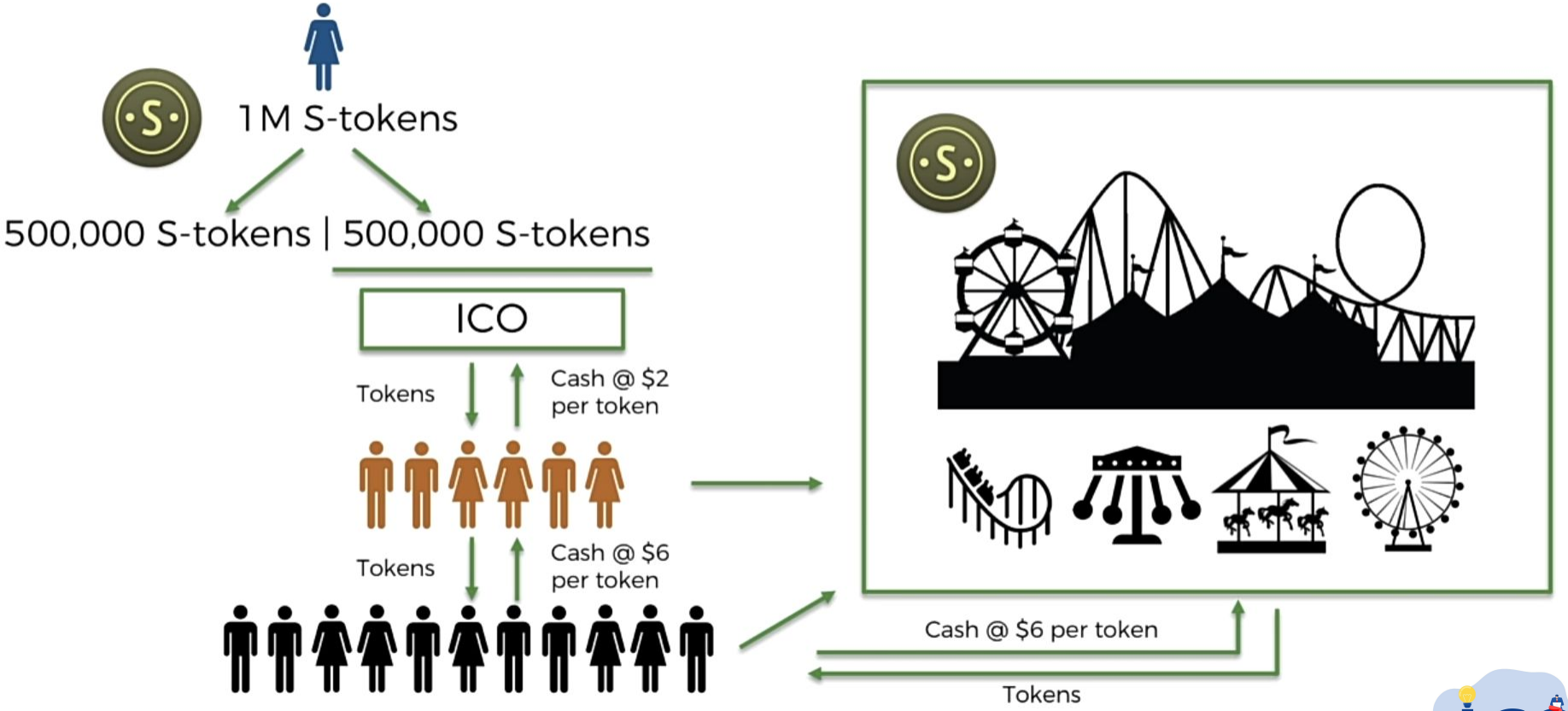
Courtesy : Wall Street Mojo - ICO

Initial Coin Offerings : ICO Vs IPO



Courtesy : SuperData Science

Initial Coin Offerings : ICO Vs IPO



Courtesy : SuperData Science

Initial Coin Offerings : ICO Vs IPO

Courtesy : Slide Team - ICO Vs IPO

Initial Coin Offerings :Advantages

1. **Access to Capital:** ICOs provide a way for startups and projects to raise funds quickly from a global pool of investors without the need for traditional financial intermediaries like banks or venture capitalists.
2. **Global Reach:** ICOs are not limited by geographical boundaries, allowing projects to tap into a global investor base, increasing the potential for significant capital inflow.
3. **Accessibility:** ICOs are open to a wide range of investors, including retail investors. This democratizes investment opportunities and allows smaller investors to participate in early-stage projects.
4. **Liquidity:** Tokens issued through ICOs often become tradable on cryptocurrency exchanges shortly after the fundraising concludes, providing liquidity to investors and enabling trading.
5. **Incentive Alignment:** ICOs can align the interests of the project's team and investors since tokens often have utility within the project's ecosystem, and the team's success is tied to the value of these tokens.
6. **Innovation:** ICOs have funded many innovative blockchain and cryptocurrency projects, pushing the boundaries of technology and fostering competition.

Initial Coin Offerings : Disadvantages

1. **Lack of Regulation:** The lack of clear and consistent regulatory oversight has led to fraudulent and scam ICOs. Investors are at risk of losing their funds due to misleading or malicious projects.
2. **High Risk:** ICO investments are highly speculative, and many projects fail to deliver on their promises or even exit scam, leaving investors with worthless tokens.
3. **Lack of Investor Protections:** Unlike traditional financial markets, ICO investors often have limited legal recourse in case of disputes or fraud.
4. **Volatility:** The value of tokens issued in ICOs can be extremely volatile, subjecting investors to significant price fluctuations.
5. **Limited Due Diligence:** ICOs often lack the level of due diligence and transparency found in traditional fundraising methods, making it challenging for investors to assess the viability of a project.
6. **Regulatory Uncertainty:** Ongoing changes in regulatory frameworks around the world have created uncertainty for ICOs. Some countries have banned or heavily regulated them, while others have embraced them.
7. **Market Saturation:** The ICO market became oversaturated at one point, making it difficult for legitimate projects to stand out and raise funds amidst a sea of offerings.
8. **Hype and FOMO (Fear of Missing Out):** ICOs can be driven by hype and FOMO, leading investors to make irrational decisions based on the fear of missing out on the next big thing

STO Metamask (Ethereum Wallet)

- STO Metamask - Security Token Offerings using Metamask (Similar to ICO)
- a **free crypto wallet software** that people can use to interact in the crypto world.
- **free to use** and **can be installed as an extension on internet browsers**, Google Chrome, Firefox, Brave, and Edge, or downloaded as a smartphone application both on iOS and Android.
- With over **30 million users, MetaMask is one of the most popular cryptocurrency** wallets today.
- used for managing Ethereum and Ethereum-based tokens.
- Metamask allows users to:
  - Buy, receive, send and swap Ether, the main token on Ethereum
  - Buy, receive, send and swap nonfungible tokens (NFTs) in marketplaces
  - Connect to Ethereum dapps
  - Connect to other crypto wallets
  - Play blockchain-based games
  - Access different networks such as the BNB Smart Chain and other testnets

Courtesy : Cointelegraph - Metamask

METAMASK

Benefits of using Metamask (Ethereum Wallet)

Its massive global user and follower base make it a vital part of the Ethereum community.

Utilizes local key storage, giving users full control over their keys. No personal information is stored either.

Allows in-app coin purchasing – users can buy Ether and ERC-20 directly from Coinbase and ShapeShift.

Allows multiple accounts, which is great for people who want to keep their finances separate.

Backs up the user account with hierarchical deterministic settings.

Has an easy-to-use interface that is good for crypto beginners.

Easily connects with popular exchanges like Coinbase and Binance.

Courtesy : Cointelegraph - Metamask

METAMASK

Advantages of Metamask (Ethereum Wallet)

1. **User-Friendly Interface:** MetaMask provides a relatively intuitive and user-friendly interface, making it accessible to both beginners and experienced users.
2. **Security:** MetaMask is considered secure, with features like seed phrases for wallet recovery, password protection, and integration with hardware wallets like Ledger and Trezor for enhanced security.
3. **Cross-Platform:** It is available as a browser extension for Chrome, Firefox, Brave, and Edge, as well as a mobile app for iOS and Android, ensuring accessibility across multiple platforms.
4. **Compatibility:** MetaMask supports Ethereum and Ethereum-based tokens (ERC-20 and ERC-721 tokens), making it versatile for managing various assets within the Ethereum ecosystem.
5. **DApp Integration:** It allows users to interact with a wide range of Ethereum-based DApps directly through their browser extension, simplifying the DApp experience.
6. **Privacy:** While Ethereum is a public blockchain, MetaMask enables users to have a degree of privacy by not revealing their real-world identity when transacting or interacting with DApps.
7. **Community Support:** MetaMask has a strong community and active development, which means regular updates, bug fixes, and improvements.

Courtesy : Cointelegraph - Metamask

METAMASK

Disadvantages of Metamask (Ethereum Wallet)

1. **Security Risks:** While MetaMask itself is secure, users can still fall victim to phishing scams and malicious websites. Users must be cautious and ensure they are using the official MetaMask extension or app.
2. **Limited to Ethereum:** As of my last knowledge update in September 2021, MetaMask primarily supports Ethereum and Ethereum-based tokens. While there are efforts to expand to other blockchains, it may not be the best choice for managing assets on other networks.
3. **Transaction Fees:** Ethereum transaction fees (gas fees) can be high during times of network congestion, making it costly to perform transactions or interact with DApps.
4. **Learning Curve:** For individuals new to cryptocurrencies and blockchain technology, MetaMask may have a learning curve, particularly when it comes to understanding gas fees, wallet management, and security practices.
5. **Lack of Customer Support:** MetaMask is an open-source project, and while it has a strong community, there is no official customer support. Users must rely on community resources for assistance.
6. **Centralization Concerns:** Some users may have concerns about the centralization of DApp interactions through a browser extension like MetaMask, as it relies on centralized infrastructure to communicate with the Ethereum network.

Courtesy : Cointelegraph - Metamask

METAMASK

Setup a Development Environment using Metamask

1. Install MetaMask:

- install the MetaMask browser extension or mobile app on your preferred browser or device.
- You can find MetaMask on Chrome, Firefox, Brave, Edge, and as a mobile app for iOS and Android.

2. Create or Import a Wallet:

- Follow the on-screen instructions to set up a wallet, including creating a strong password and securely storing your recovery seed phrase.
- If you already have a MetaMask wallet, you can import it using your existing seed phrase or private key.

3. Switch to a Test Network:

- In a development environment, it's common to use test networks like the Ropsten, Rinkeby, or Kovan testnets to avoid spending real Ether on testing transactions.
- To switch to a test network in MetaMask:
  - Click on your MetaMask extension or app to open it.
  - Click on the network name (usually "Main Ethereum Network" by default) at the top.
  - Select "Custom RPC" if your desired test network is not listed, and then enter the network's details (such as the network name, RPC URL, and chain ID).

Courtesy : Cointelegraph - Metamask

METAMASK

Setup a Development Environment using Metamask

4. Fund Your Test Wallet:

- To interact with the test network, you'll need test Ether (tEther) instead of real Ether.
- tEther can be obtained from a faucet specific to the test network you're using.
- **Faucets** are websites that provide free test Ether for development purposes. Search online for the test network's faucet and follow the instructions to get some tEther.

5. Start Developing and Testing:

- With your MetaMask wallet configured for the test network and funded with tEther, you can now start developing and testing your decentralized applications (DApps) or smart contracts.
- Use tools like Remix, Truffle, or Hardhat for Ethereum development and connect them to MetaMask to interact with your wallet and the blockchain.
- When interacting with your DApps or smart contracts, MetaMask will prompt you to confirm transactions and sign messages as needed. Remember that transactions on the test network are not real and won't affect your actual cryptocurrency holdings.
- By setting up a development environment with MetaMask, you can test your applications and smart contracts in a safe and controlled environment before deploying them to the Ethereum mainnet.

Courtesy : Cointelegraph - Metamask

METAMASK