Explained: Smart Contracts And dApps

cryptopolitan.com/explained-smart-contracts-and-dapps

Alden Baldwin

Smart Contracts and Decentralized Applications (dApps) are two of the most significant innovations in the world of blockchain technology. The rise of smart contracts and dApps has changed the way that businesses and individuals conduct transactions and interact with each other. In this guide, we will explain what smart contracts and dApps are, how they work, and their impact on the future.

Smart Contracts

Origins and development

The concept of smart contracts was first introduced in 1994 by computer scientist and legal scholar Nick Szabo. Szabo defined smart contracts as self-executing contracts with the terms of the agreement between buyer and seller being directly written into lines of code. Smart contracts run on a blockchain network, and their execution and enforcement are ensured by the network's consensus mechanism.

How they work

Here's a step-by-step explanation of how smart contracts work:

1. Definition: The first step in creating a smart contract is defining its terms and conditions in code. This code specifies what actions will trigger the execution of the contract, and what the outcome of those actions will be.

2. Deployment: Once the code has been written, the smart contract is deployed to the blockchain network. This can be done by a developer or a user, and the deployment is verified by the network's nodes.

3. Trigger: A smart contract is triggered when a specific set of conditions is met. For example, if a smart contract is used to transfer money from one user to another, the trigger might be the receipt of payment from the first user.

4. Execution: When the trigger conditions are met, the smart contract is automatically executed by the network. This execution can involve sending funds, changing the ownership of assets, or executing any other action specified in the contract code.

5. Validation: Before a smart contract is executed, its terms and conditions are validated by the network's consensus mechanism. This ensures that the contract is functioning as intended and that no fraud or other malicious activity is taking place.

6. Enforceability: Once a smart contract has been executed, its terms and conditions are enforceable by the network. This means that the outcome of the contract is guaranteed to be the same for all participants, and there is no need for intermediaries such as lawyers or notaries.

Advantages

One of the biggest advantages of using smart contracts is their ability to automate the process of contract enforcement. This reduces the need for intermediaries and cuts down on transaction costs. Smart contracts also increase transparency and security, as they are stored on a decentralized blockchain network.

Decentralized Applications (dApps)

Definition

dApps, or decentralized applications, are a type of software application that runs on a decentralized network. Unlike traditional centralized applications, dApps are not controlled by any single entity and have no single point of failure. Instead, they are powered by a decentralized network and run on a set of rules encoded in smart contracts.

Types

1. Financial dApps: Financial dApps are decentralized applications that provide financial services, such as remittances, loans, and investments. Examples of financial dApps include decentralized exchanges, peer-to-peer lending platforms, and robo-advisory platforms. These dApps aim to provide financial services that are more secure, transparent, and accessible than traditional centralized financial services.

2. Gaming dApps: Gaming dApps are decentralized applications that allow users to play games, earn rewards, and trade digital assets. These dApps are based on blockchain technology and use smart contracts to provide secure and transparent game experiences. Examples of gaming dApps include blockchain-based games like CryptoKitties, Axie Infinity, and F1DeltaTime.

3. Social dApps: Social dApps are decentralized applications that provide social networking services, such as messaging, forums, and content sharing. These dApps aim to provide social networking services that are more secure, transparent, and censorship-resistant than traditional centralized social networks. Examples of social dApps include Mastodon, Peepeth, and Minds.

4. Identity dApps: Identity dApps are decentralized applications that provide identity management services, such as verification, authentication, and authorization. These dApps aim to provide identity management services that are more secure, transparent, and privacy-protecting than traditional centralized identity management services. Examples of identity dApps include uPort, Civic, and SelfKey.

5. Supply Chain dApps: Supply chain dApps are decentralized applications that provide supply chain management services, such as traceability, transparency, and efficiency. These dApps aim to provide supply chain management services that are more secure, transparent, and efficient than traditional centralized supply chain management services. Examples of supply chain dApps include VeChain, Ambrosus, and Provenance.

6. Governance dApps: Governance dApps are decentralized applications that provide governance services, such as voting, decision-making, and community management. These dApps aim to provide governance services that are more secure, transparent, and democratic than traditional centralized governance services. Examples of governance dApps include Aragon, DAOstack, and Colony.

How they work

1. Decentralized Network: The first component of a dApp is the decentralized network on which it runs. This network can be based on blockchain technology, such as Ethereum or EOS, or it can be based on other decentralized technologies such as InterPlanetary File System (IPFS).

2. Smart Contracts: The second component of a dApp is the smart contract that powers it. A smart contract is a self-executing contract with the terms of the agreement written into lines of code. When certain conditions are met, the smart contract is automatically executed by the network.

3. Front-End Interface: The third component of a dApp is the front-end interface that users interact with. This interface can be built using a variety of technologies, such as HTML, CSS, and JavaScript, and it allows users to interact with the dApp and execute the actions specified in the smart contract.

4. Decentralized Storage: The fourth component of a dApp is the decentralized storage solution that is used to store data and assets. This can be done using decentralized storage solutions such as IPFS or Swarm, or it can be done using blockchain-based storage solutions such as Ethereum's Swarm or EOS's IPFS-like interplanetary database (IPDB).

5. Network Consensus Mechanism: The final component of a dApp is the network's consensus mechanism. This mechanism is used to validate transactions and ensure that the dApp is functioning as intended. It also ensures that the dApp is secure and that the assets and data stored on the network are protected.

How dApps Use Smart Contracts

dApps (Decentralized Applications) use smart contracts to facilitate, verify, and enforce the negotiation or performance of a contract. Smart contracts are used to enforce the rules and regulations of the dApp, and ensure that all transactions on the dApp are executed in a secure, transparent, and decentralized manner.

Here's a simple example of how a dApp can use smart contracts:

Suppose there is a decentralized betting platform for sports events. The dApp's smart contract defines the rules for placing bets, such as the minimum and maximum bet amounts, the start and end times for betting, and the payout structure for winners. When a user wants to place a bet, they initiate a transaction on the dApp, which triggers the execution of the smart contract.

The smart contract checks if the user's bet is within the specified limits, and if it is, it deducts the bet amount from the user's wallet and adds it to the betting pool. Once the sports event is over, the results are recorded on the blockchain, and the smart contract automatically calculates the payouts for the winners based on the rules defined in the contract.

The smart contract then transfers the winnings from the betting pool to the winners' wallets. In this scenario, the smart contract ensures that all transactions are executed in a secure and transparent manner and that the rules for the betting platform are followed. The smart contract also ensures that there is no need for a central authority to oversee the betting platform and make sure that the rules are followed.

Conclusion

Bottomline is dApps and smart contracts are game-changers in the digital world, offering new and innovative ways for individuals and businesses to interact with one another. Through their decentralized nature, dApps offer greater security, transparency, and democratization, providing users with greater control and ownership over their data and assets.

While there are still some limitations to be addressed, the potential applications of dApps and smart contracts are vast and varied, and their impact on various industries will only continue to grow. Whether it's through the creation of new financial systems, more efficient supply chains, or entirely new business models, the future of dApps and smart contracts is one that is exciting and full of possibilities.