Oracles

note This is a conceptual overview of oracles. For more detailed information on how to use oracles in your applications, check out ourthird-party oracles documentation . In this conceptual overview, we'll explore oracles, how they work, and some general applications. This overview will provide a foundational understanding and set expectations for developers who want to integrate oracles into their applications.

What are oracles?

Oracles are third-party services that provide smart contracts with external information. They act as a bridge between blockchains and the outside world, which expands their functionality by enabling smart contracts to access data beyond their native networks.

Types of oracles

Oracles can be classified based on their source, direction of information, trust, and how they provide information to smart contracts. Some common types of oracles include:

- Inbound and Outbound oracles
- : Inbound oracles share information from external sources to smart contracts, while outbound oracles send information from smart contracts to the external world.
- Centralized and Decentralized oracles
- : A centralized oracle is a single entity and sole data provider for a smart contract. Decentralized oracles increase reliability by relying on multiple sources of truth and distributing trust among participants.
- Push and Pull oracles
- : Push oracles proactively provide data to smart contracts without being explicitly requested. They push data to the smart contract when a specified event or condition occurs. On the other hand, pull oracles require smart contracts to request data explicitly. They pull data from external sources in response to a query from the smart contract.
- Software oracles
- : These oracles interact with online sources of information, such as databases, servers, or websites, and transmit the data to the blockchain. They often provide real-time information like exchange rates or digital asset prices.
- Hardware oracles
- : These oracles obtain information from the physical world using electronic sensors, barcode scanners, or other reading devices. They "translate" real-world events into digital values that smart contracts can understand.

How do push oracles work?

Push oracles proactively provide data to smart contracts without being explicitly requested. When a specified event or condition occurs, the push oracle triggers the smart contract with the relevant data. For example, a push oracle might send weather data to a smart contract once the temperature reaches a certain threshold.

How do pull oracles work?

Pull oracles require smart contracts to request data explicitly. A smart contract sends a query to the oracle, retrieving and relaying the requested information to the contract. For example, a smart contract might request the current price of a specific digital asset from a pull oracle.

Use cases for oracles

Oracles serve a purpose in various applications across industries. Some general use cases include:

- Prediction markets
- : Oracles provide real-world data to prediction market platforms, allowing users to bet on future events or outcomes.
- Supply chain management
- : Hardware oracles can track the location and status of goods throughout the supply chain, enabling smart contracts to automate various processes and improve efficiency.
- Insurance
- : Oracles can supply data about events such as natural disasters, accidents, or price fluctuations, allowing smart contracts to automate claims processing and payouts.
- Decentralized finance (DeFi)
- : Oracles provide critical price and market data to DeFi applications, enabling them to operate efficiently and securely.

In summary, oracles are a crucial component of the blockchain ecosystem, bridging the gap between on-chain and off-chain data sources. They enhance the functionality of smart contracts and enable a wide range of applications across various industries. As blockchain technology continues to evolve, developing secure and reliable oracles will remain essential in unlocking the full potential of smart contracts and decentralized applications.

Resources

You can learn more about oracles in our third-party oracles documentation . Edit this page Last updatedonJan 27, 2025

Previous Solidity support Next Overview