UNDERSTANDING SMART CONTRACTS

UNDERSTANDING SMART CONTRACTS

CONTENT

This publication introduces smart contracts, a new instrument which is deployed on a distributed ledger technology[1] (DLT) such as a blockchain[2].

A smart contract is a collection of code (its functions) and data (its state) that resides at a specific public address on a DLT (e.g., Ethereum Blockchain), typically written in programming language such as Solidity or JavaScript. A smart contract constitutes a set of rules agreed by the involved parties which are a translation of all contractual terms into logic. Its purpose is to automate the contracting process and enable monitoring and enforcement of contractual promises with minimal human intervention. It may execute transactions (e.g., exchange of money, property register, shares transfer or anything of value) and/or enforce agreements based on the fulfilment of the terms of the agreement in a transparent manner, eliminating the need for a middleman and keeping the system conflict-free. When a predefined condition is met, the smart contract code executes on its own and provides an output such as a transfer of crypto assets to a crypto wallet, or a data feed to an internal or external source.

An example of a smart contract is the issuance of new crypto assets through an Initial Coin Offering (ICO)[3]. When the conditions defined in the smart contract are met (e.g., validation of the payment of certain cryptocurrencies, etc.), the smart contract executes the pre-defined actions including issuance of new crypto assets as specified in the ICO.

Currently, most smart contracts are financial or notary in nature. Below are some examples of existing smart contracts:

- Contracts that implement crowdfunding services, gathering funds from investors in order to fund projects (often known as ICOs)
- Contracts which provide insurance on setbacks which are digitally provable (e.g., insurance policies for flights; if a flight is delayed or cancelled, one obtains a refund)
- Contracts that allow users to write the hash of a document on the DLT, so that they can prove document existence and integrity
- Contracts that declare copyrights on digital art files, like photos or music.

A smart contract shares similar properties as a vending machine which has hard coded rules that define what happens when certain conditions are met and then executes certain actions when those conditions are fulfilled.

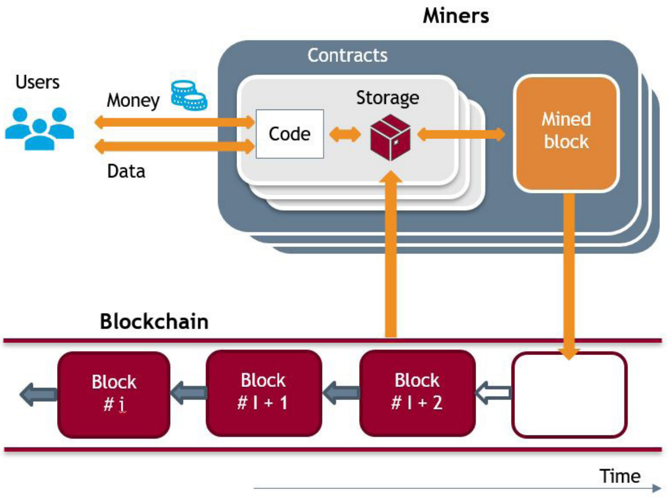
[1] Distributed ledger technology (DLT) is a digital system for recording the transaction of assets in which the transactions and their details are recorded in multiple places at the same time. Unlike traditional databases, distributed ledgers have no central data store or administration functionality. Blockchain is the first fully functional Distributed Ledger Technology.

[2] A blockchain is a continuously growing list of records called blocks, which are linked and secured using cryptography. Each block typically contains a cryptographic hash of the previous block, a timestamp and transaction data. By design, a blockchain is resistant to modification of the data. It is "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way" for use as a distributed ledger. A blockchain is typically managed by a peer to peer network collectively adhering to a protocol for inter-node communication and validating new blocks. Once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks, which requires consensus of the network majority.

[3] An ICO is similar to an IPO (i.e. an initial public offering) of a company's shares on a stock exchange, except that an ICO involves the issuance of new crypto assets rather than shares.

By its nature, a smart contract deployed on a DLT such as a blockchain is unchangeable, irreversible, precise, public and autonomous. Transactions and smart contract details are visible to everyone who has permission to access the DLT.

Smart contracts can be used for simple economic transactions, for instance, to send funds or to execute more complex transactions such as registering ownership and property rights.



Input data to a smart contract may be manual (e.g., individual transfer of cryptocurrency) or triggered autonomously by a data feed from sources outside the DLT (i.e., oracles[4]), other smart contracts, smart devices (often referred as Internet of things (IOT), or other.

Transactions executed by a smart contract:

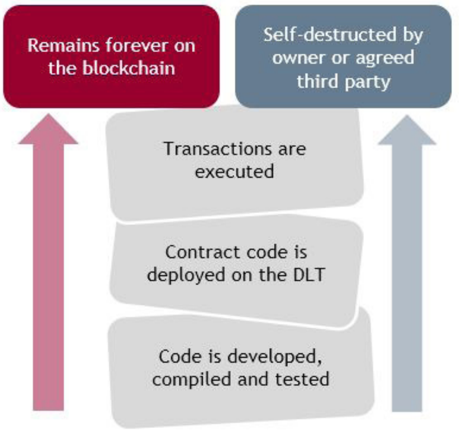
- Need to be validated and accepted by validating nodes[5] or miners (i.e., there must be consensus)
- Need to be associated with parties that have sufficient resources for transaction fees
- Are kept in different blocks on the DLT
- Are linked to the relevant smart contract
- Have a distinct public address (transaction hash)
- Have a transaction status (i.e., success, fail or pending)
- Their details can be viewed on the DLT general ledger.

Each smart contract binds to a contract account (i.e., contract owner), has a public address which can be used to receive funds or input data into the contract, and can be linked to other smart contracts on the same DLT. Some smart contracts may have been developed and deployed by a third-party service organisation on behalf of the contract owner.

The contract account needs sufficient funds (i.e., the platform's accepted cryptocurrencies or tokens) to deploy a new smart contract which is created by a transaction that sends the contract code to the DLT. Development and deployment of a new smart contract is possible by various web-based tools, installed tools or from the contract owner's crypto wallet that supports smart contracts development.

Once deployed, the same smart contract code is kept on all the DLT nodes (in a block) making it possible to input data or send cryptocurrencies from any node separately. Accordingly, transactions are executed on each node participating in the network as part of their verification of new blocks.

Since the smart contract code is embedded on a DLT, it is unchangeable and irreversible. Transactions are executed as long as enough funds or fuel exists (such fuel is called 'gas' on the Ethereum Blockchain), unless its owner or another party as defined in the contract terminates the smart contract by activating the self-destruct function (in order to terminate a smart contract, proper code must be embedded in the smart contract).


Remains forever on the blockchain
Self-destructed by owner or agreed third party
Transactions are executed
Contract code is deployed on the DLT
Code is developed, compiled and tested

A temporary delay in executing smart contract transactions is impossible unless the smart contract owner inserted this possibility into the code, with or without the other parties' consent.

SMART CONTRACT INTERACTION WITH EXTERNAL DATA SOURCES

Parties engaged in a smart contract can specify an external independent service (e.g., an oracle) to carry out the data input (e.g., an event or condition) for activating the smart contract algorithm and executing the transaction. The primary task of such oracle use is to capture, verify real-world occurrences and feed the smart contract with that data in a secure and trusted manner. Binding a smart contract to an external data source requires the smart contract code to precisely point to the external source's location, define the frequency of data reading, etc.

An oracle may feed the smart contract with various input types and formats, from different sources (e.g., weather temperature, successful payment, price fluctuations or flights delays). In certain circumstances, oracles may be part of multi-signature contracts where two parties sign a contract for the future release of funds or for the registration of assets only if certain conditions are met. An oracle may also be another smart contract that inputs data to the primary smart contract.

Different types of oracles exist based on the planned use of the smart contract and the data source. For example, software oracles handle information available online or hardware oracles obtain input from physical world appliances (i.e., sensors, IOT, etc.).

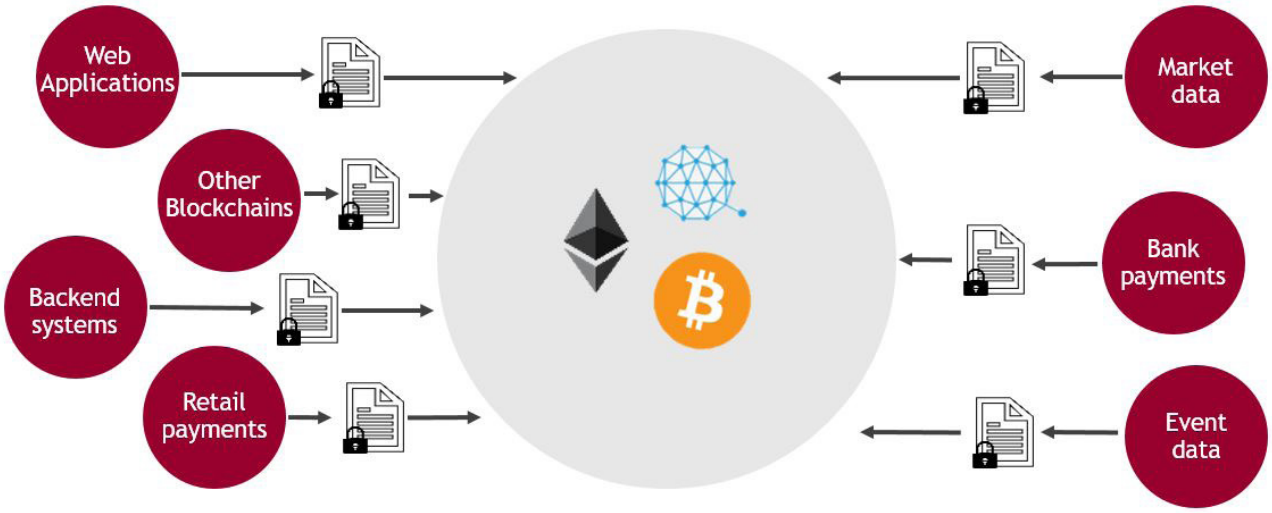
[4] Oracles are the interface between smart contracts and the outside world; they store data that resides outside of the blockchain and that is relevant to the blockchain smart contract(s).

[5] Nodes are connected devices which verify the DLT transactions, store blocks and broadcast transaction history to other nodes.

We should be aware that miners verify that an oracle's input is allowed technically, that sufficient funds are available and that any resulting transactions are legitimate, but they do not confirm the validity of the data feed or if it matches the values of the source data outside of the DLT.

The quality and availability of the oracle input is important for the correct execution of smart contract transactions, as the insertion of an incorrect data feed may result in a transfer of funds or ownership without possibility of refund. Thus, parties may define that execution of a smart contract transaction depends on input data from multiple data sources or oracles.

Since most oracles are developed and maintained by external suppliers, the supplier's reputation as well as existing internal controls should be assessed to identify the potential for fraud through manipulation of the input data to the smart contract.



SUMMARY

Smart contracts, as well as the DLT, are currently in early stages. As a result, processes related to smart contracts and the DLT will evolve over time. Entities who participate in smart contract transactions need to continually update their knowledge and where necessary, seek assistance from service providers with relevant knowledge and experience.

CONTACT:

AUDIT@BDO.GLOBAL

This publication has been prepared and issued by the Global Assurance Department.

‘BDO’, ‘we’, ‘us’, and ‘our’ refer to one or more of BDO International Limited, its network of independent member firms (‘the BDO network’), and their related entities.

Service provision within the BDO network is coordinated by Brussels Worldwide Services BVBA, a limited liability company incorporated in Belgium.

Each of BDO International Limited (the governing entity of the BDO network), Brussels Worldwide Services BVBA and the member firms is a separate legal entity and has no liability for another such entity's acts or omissions. Nothing in the arrangements or rules of the BDO network shall constitute or imply an agency relationship or a partnership between BDO International Limited, Brussels Worldwide Services BVBA and/or the member firms of the BDO network. Neither BDO International Limited nor any other central entities of the BDO network provide services to clients.

BDO is the brand name for the BDO network and for each of the BDO member firms.