Smart Contracts

Lars Erik Wik

May 3, 2022

Summary

Smart contracts are programs stored on a public ledger which is ran in a virtual machine by peers in a blockchain network. The program code is invoked by transactions from interacting entities, which can be users or other smart contracts. Some smart contracts like those in Ethereum use "gas" to incentivise efficient smart contracts and to prevent infinite loops.

Smart contract programming involves interaction with complex state machines, and is thus prone to logical errors. A simple type of error is a contract leaking funds. Once a smart contract is added to the ledger, it cannot be modified. Thus a smart contract developer must deploy a new smart contract in order to add features or fix bugs.

Additionally the consensus requirements needed in order to maintain the ledger greatly affects a smart contracts performance in terms of throughput and delay. Running the contract code also requires nodes to utilize vast amounts of computational power and storage capacity. Which again makes smart contracts very expensive to compute.

In order for the blockchain network to reach consensus, a smart contract program must be completely deterministic. This greatly affects the number of use cases a smart contract can solve.