Blockchain Council™

Key concepts of Corda

Key Concepts

Key concepts are developed for those who wants to understand its basic architecture. It contains:

- Network
- Nodes
- Transactions
- Consensus
- Contracts
- Identity
- Flow
- Time windows
- Notary Services
- Oracle Services

The Network

- An Authenticated peer-to-peer network .
- Each node hosts Corda services on JVM and executes applications known as CorDapps.
- The flow of all communication between nodes is straightforward, messages are TLS-encrypted and sent over AMQP/1.0.
- Need-to-know basis shareability.
- IP addresses of nodes through which nodes can be reached are published by network map service.
- The Corda Network consists of two network services:
    1. Notary Services
    2. Oracle Services

Nodes

- A JVM run time environment with a unique identity
- Hosts Corda services and CorDapps.

Architecture

- Persistence Layer
    - Vault
    - Storage Service
- Network Interface
- RPC interface
- The Service hub
- TheCorDApp provider

Transactions

- Transactions in Corda are the proposals to update the ledger
- UTXO (Unspent Transaction Output) model is used by Corda, where every state on the ledger is immutable.
- Ledger evolvement happens while applying transactions.
- **INPUTS:** Zero or more existing ledger states are marked as historic in transaction application.
- **OUTPUTS:** Production of zero or more new ledger states in transaction application.

Types of Transactions:

- Notary change transactions which are used to change a state's notary
- General transactions used for everything else.

Conditions :

- Validity
- Uniqueness

Consensus

Transactions must achieve uniqueness and validity consensus, to be committed

Validity Consensus: It is the process of checking following conditions hold both, for every transaction in transaction chain that generates input to the proposed transaction and for the proposed transaction.

Uniqueness Consensus: Uniqueness Consensus must be achieved by a transaction proposal to be valid.

Contracts

- Contract of every input and output states must accept a **valid transaction.**
- Contracts are always written in JVM programming language, e.g., Kotlin or Java
- The Contract execution is deterministic, and the transaction's content alone decides the acceptance of a transaction

Verification and Validity of Transaction

- The contract must be pointed by each state.
- As input, a contract takes a transaction, and as output, it states that whether the transaction can be considered as valid based on contract rules.
- The validity of transaction depends if the contract of every input state and every output state acknowledge it to be valid.

Identity

By identities in Corda it represents:

- An Organisation’s **Legal Identity**
  - Parties involved in transaction use their legal identities
  - Doorman must sign and attests the identities by the X.509 Certificates.
  - Only attested and well-known identities are published on the network map
- Identities can be confidential, which are only shared on a need to know basis
- Nodes must verify the identity of the owner of a public key, which can only be achieved by X.509 certificates

Flow

- Point-to-point messaging is used in Corda networks
- What information needs to be sent, to which counterparties, and in what order to be sent, it is all required from network participants when coordinating a ledger update.
- Telling a node that how to achieve a specific ledger update.

-> Library of flow is provided by Cordato store common tasks so that developers do not have to redefine the logic for common processes

Time Windows

- Specify a time window within which a particular transaction claim to occur.
- Expressed as windows, as there is no true time in distributed system

Notaries

- Prevents double-spends (see consensus).
- A Network can have several notaries, and each is running different consensus algorithm
- Notary provides Uniqueness Consensus.
- Flag the transaction if double-spent attempt detected and rejects it.
- Each state has their appointed notary which notarise a transaction and all transaction's input states.

Notary in different terms:

- Structure
- Consensus Algorithm

Multiple Notaries:

There can be multiple notaries in each cordanetwork each of which runs different consensus algorithm.

Oracle

- As a part of the command, a fact can be included in a transaction.
- Oracle is defined as a service which will only sign transaction if the included fact is true.
- Oracle provides commands to encapsulate a specific fact(e.g., exchange rate), and Oracle is listed as a required signer.
- Oracle uses Merkle trees to “tear-off” the transaction.

Blockchain Council™

THANK YOU!

Any questions?

You can mail us at

hello@blockchain-council.org