Pay, settlement
Verification, uniqueness (I/P, O/P, peers)
B-> 50 lakhs
Notaries: Source R3 official documentation

Summary

- The notary service prevents “double-spends”.
- The notary also acts as the time-stamping authority. If a transaction includes a time window, it can only be notarized during that window.
- Notary clusters may optionally also validate transactions, in which case they are called “validating” notaries, as opposed to “ ”.
- A network can have several notary clusters, all running different consensus algorithms .
-

Overview

The notary is Corda’s uniqueness consensus service. It prevents double-spends by ensuring each transaction contains only unique input states. A notary service is formed by one or more notary workers that together form a notary cluster. The notary’s role is to ensure a transaction contains only unique input states. The cluster’s signature is obtained once it verifies that a proposed transaction’s input states have not already been consumed by a prior transaction. Upon determining this, the notary cluster will either:

- Sign the transaction in cases where all input states are found to be unique.
- Reject the transaction and flag that a double-spend attempt has occurred in cases where any of the input states are identical to those already encountered in a previous transaction.

Every state has an appointed notary cluster, so the cluster will only notarize a transaction if it is the appointed notary cluster of all the transaction’s input states.

Validation

A notary cluster can be configured to provide validity consensus by validating each transaction before committing it. There are therefore two notary deployments available:

- The non-validating notary, where the transaction is not checked for validity.

- The validating notary, where the transaction is checked for validity.

Data visibility

Below is a summary of which specific transaction components have to be revealed to each type of notary:

Transaction components	Validating	Non-validating
Input states	Fully visible	References only [1]
Output states	Fully visible	Hidden
Commands (with signer identities)	Fully visible	Hidden
Attachments	Fully visible	Hidden
Time window	Fully visible	Fully visible
Notary identity	Fully visible	Fully visible
Signatures	Fully visible	Hidden
Network parameters	Fully visible	Fully visible

Both types of notaries record the calling party's identity: the public key and the X.500 Distinguished Name.

[1] A state reference is composed of the issuing transaction's ID and the state's position in the outputs. It does not reveal what kind of state it is or its contents.

Multiple notaries

Each Corda network can have multiple notary clusters. This has several benefits:

- Privacy - with both validating and non-validating notary clusters on the same network, nodes can choose the preferred notary cluster on a per-transaction basis.

- Load balancing - spreading the transaction load over multiple notary clusters allows higher transaction throughput for the platform overall.
- Low latency - latency can be minimized by choosing a notary cluster physically closer to the transacting parties.

Changing notaries

Remember that a notary cluster will only sign a transaction if it is the appointed notary cluster of all the transaction's input states. However, there are cases in which it may be necessary to change a state's appointed notary cluster. These include:

- When a single transaction needs to consume several states that have different appointed notary clusters
- When a node would prefer to use a different notary cluster for a given transaction due to privacy or efficiency concerns

Before these transactions can be created, the states must first all be re-pointed to the same notary cluster. This is achieved using a special `notary-change` transaction that takes:

- A single input state
- An output state identical to the input state, except that the appointed notary cluster has been changed

The input state's appointed notary cluster will sign the transaction if it does not constitute a double-spend, at which point a state will enter existence with all the properties of the old state, but with a different appointed notary cluster.

When a notary service is not expected:

1. When a transaction does not have a I/P state
2. There is no time window
3. Peers are not authenticated

key(transaction Id, Output index)
value (transaction Id, input index, requested peer): indicates ID of a transaction, which is used to determine the transaction, input index helps to identify the historical transaction, peer requesting your notary service.-> Add state to input index of merkle tree
Otherwise throw exception