Corda is a Distributed Ledger Technology to be used by businesses, such as financial institutions, to keep a shared ledger of transactions and thus removing the need for the involved parties to constantly check that each of their books are in line after interacting with each other. This is the primary problem that Corda is trying to solve.

It also removes the need for all parties on the Corda network to know about each and every transaction.

Software requirements

* Oracle JDK 8 JVM - minimum supported version 8u171
* IntelliJ IDEA - supported versions 2017.x and 2018.x (with Kotlin plugin version 1.2.51)
* Git

NOTE -:

* Corda runs in a JVM. JVM implementations other than Oracle JDK 8 are not actively supported.
* Applications on Corda (CorDapps) can be written in any language targeting the JVM. However, Corda itself and most of the samples are written in Kotlin. Kotlin is an [official Android language](https://developer.android.com/kotlin/index.html).

Key concepts

*The network*

* A Corda network is made up of nodes running Corda and CorDapps
* The network is permissioned, with access controlled by a doorman
* Communication between nodes is point-to-point, instead of relying on global broadcasts.
* All communication between nodes is direct, with TLS-encrypted messages sent over AMQP/1.0.
* Each network has a **network map service** that publishes the IP addresses through which every node on the network can be reached, along with the identity certificates of those nodes and the services they provide.
* Each network has a doorman service that enforces rules regarding the information that nodes must provide and the know-your-customer processes that they must complete before being admitted to the network.
* To join the network, a node must contact the doorman and provide the required information. If the doorman is satisfied, the node will receive a root-authority-signed TLS certificate from the network’s permissioning service.

*Network services*

Nodes can provide several types of services:

1. One or more Notary services –

* May be run on a single node, or across a cluster of nodes.
* It guarantees the uniqueness and the validity of ledger updates.

1. Zero or more **oracle services –**

* It signs transactions