Blockchain Honor Degree Sem VI

HBCC 601 : Blockchain Platforms

Module - 6 : Other Blockchain Platforms (6 Hours)

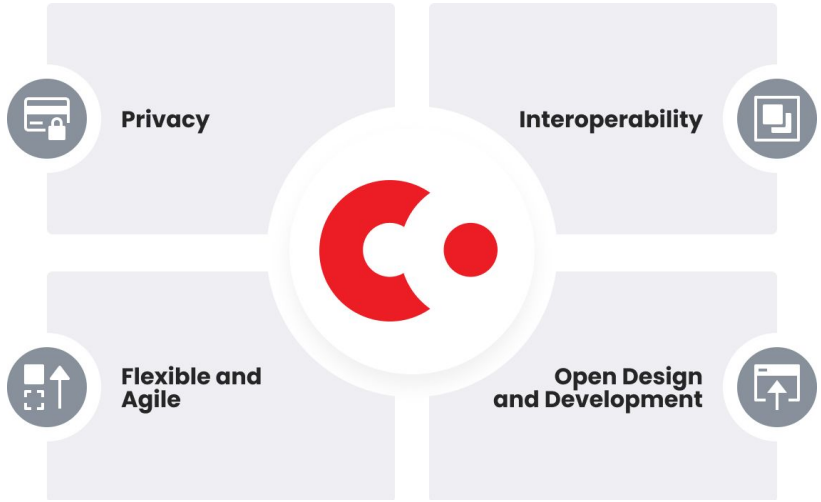
Instructor : Mrs. Lifna C S

Agenda

- **Corda**
- Ripple
- Quorum
- Other Emerging Blockchain Platforms
- Blockchain in DeFi: Case Study on any of the Blockchain Platforms
- Comparison Tables

Corda

- **open source blockchain platform to solve complex business problems**.
- Permissionless blockchain platforms, in which all data is shared with all parties, are largely unsuited for businesses.
- designed **to only share data with relevant parties.**
- **flexible and scalable**
- **ensures a high level of privacy and security.**
- **Distributed ledger applications on a high performance, enterprise grade platform**.



Courtesy : R3 Training

R3 Timeline

SEPT. 2014 | First R3 roundtable held in NY; attended by eight banks to understand blockchain

SEPT. 2015 | Nine banks form a partnership with R3 to design and deliver advanced DLTs to global financial markets

BARCLAYS BBVA CommonwealthBank CREDIT SUISSE

Goldman Sachs J.P.Morgan STATE STREET UBS Royal Bank of Scotland

SEPT. 2015 | R3 launches the Architecture Working Group to architect an enterprise-grade blockchain

DEC. 2015 | R3 consortium grows to 42 bank members

NOV. 2016 | Corda Open Source GA

c·rda

JULY 2018 | Corda Enterprise 3.0 launches with 100% interoperability with Corda 3.0 open source

c·rda enterprise

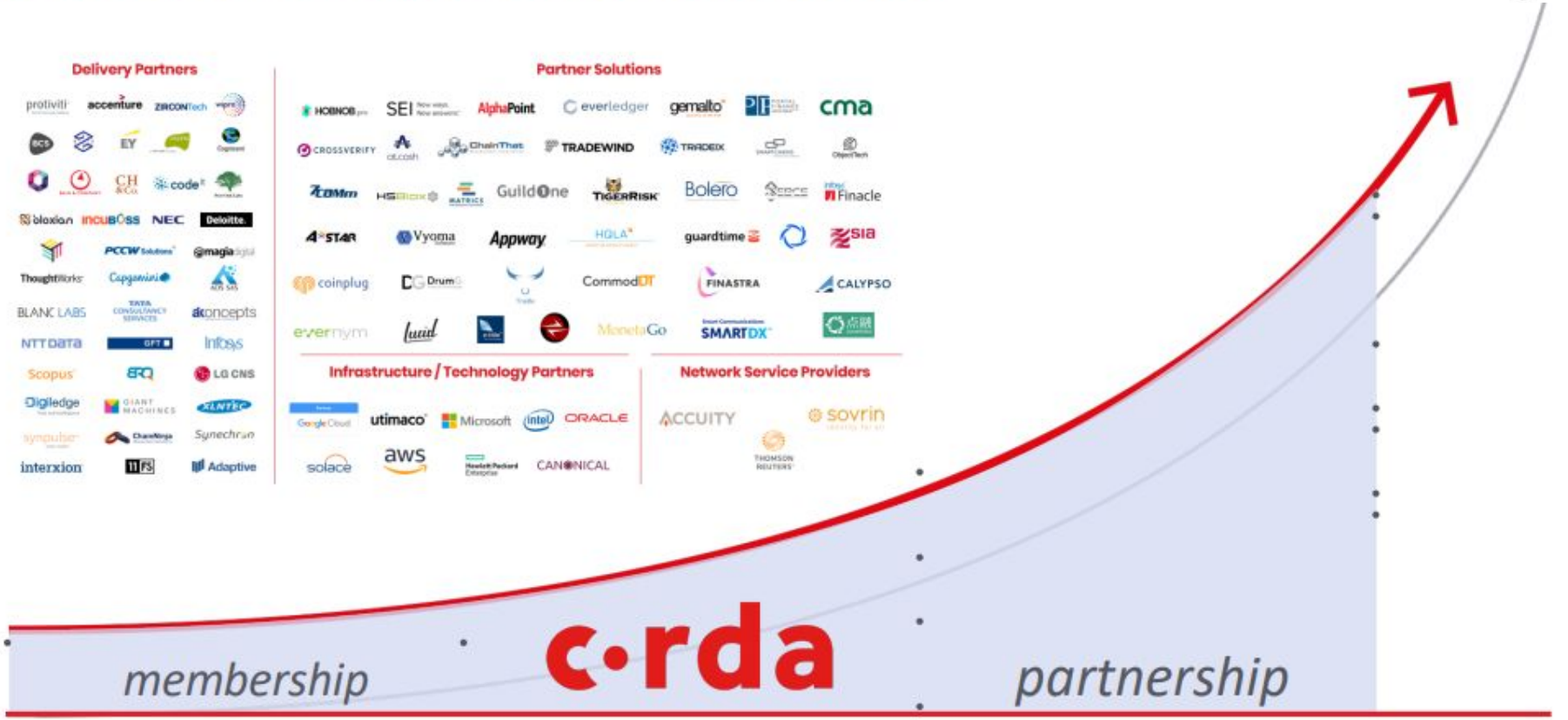
AUG. 2018 | Ecosystem grows to 200+ financial institutions, regulators, trade associations, professional services firms and tech companies

Jan. 2019 | Corda Network launches with new governing foundation, enabling interoperability

Feb. 2019 | Corda 4.0 launches

Courtesy: R3 Training

artner Network 200+, CorDapps hit production



scover the power of blockchain

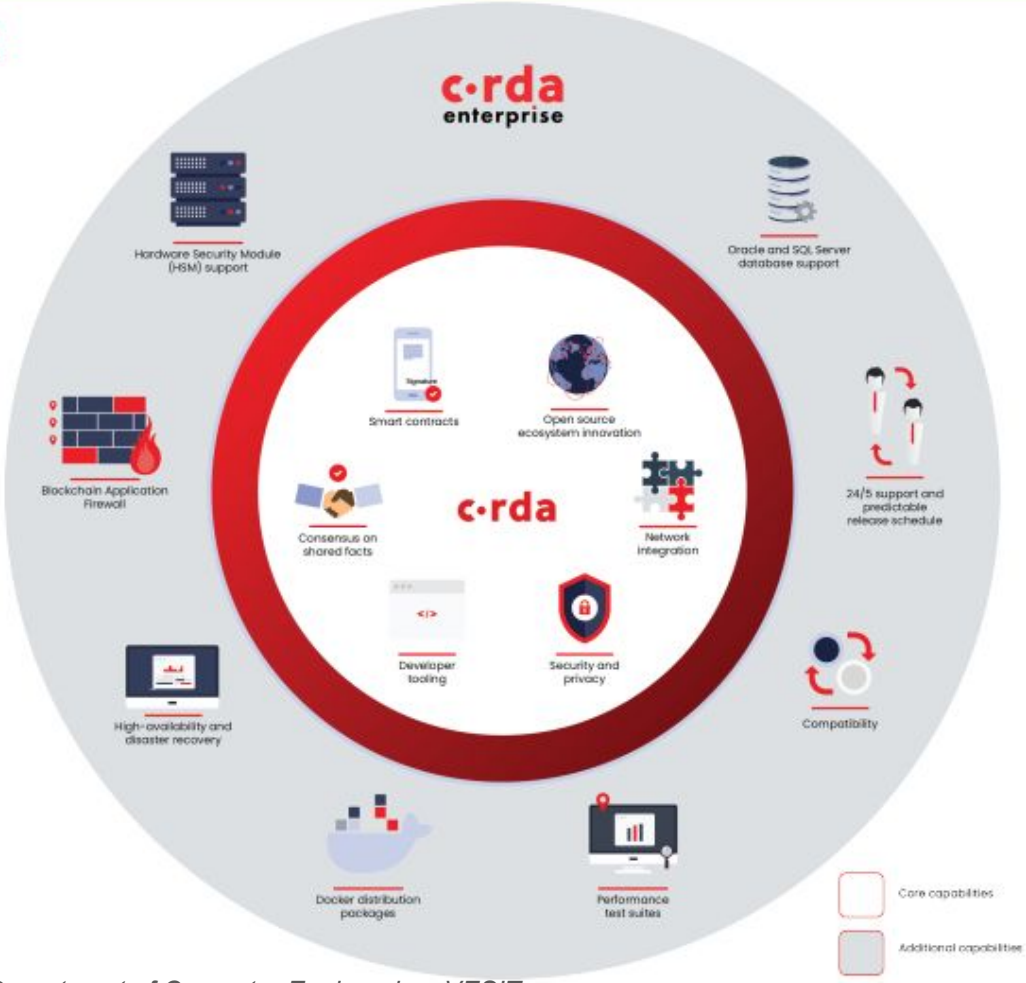
R3 rethought the blockchain concept from top to bottom to build a different kind of blockchain.

Corda removes costly friction in business transactions by enabling institutions to transact directly using smart contracts, while ensuring the highest levels of privacy and security.

Corda adoption is through R3 Ecosystem participation. Blockchain technology is dependent on a network effect and R3 offers a thriving ecosystem of 200+ firms to drive industry-wide collaboration.

Courtesy: R3 Training

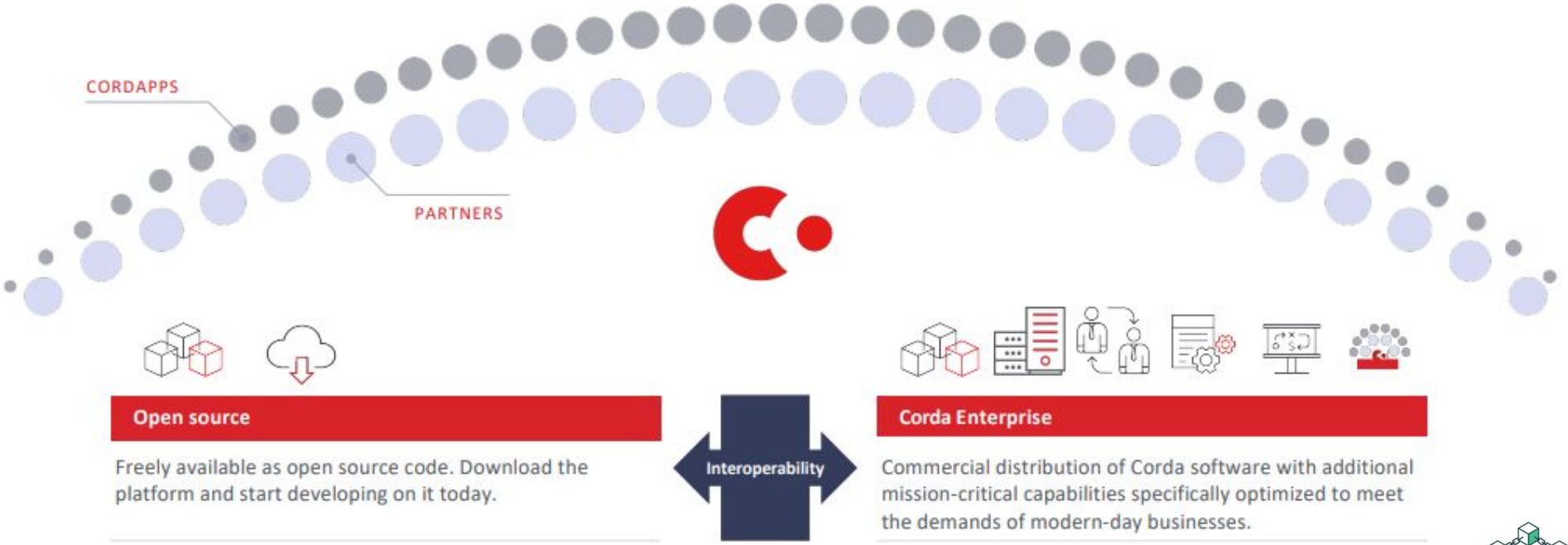
orda and Corda Enterprise: seamless interoperability



Courtesy: R3 Training

lockchain for every business in every industry

Select a version of Corda that fits your unique needs – regardless of industry, size, and stage of development



Courtesy: R3 Training

orda is the 3rd Generation Blockchain: Open & Interoperable, With Priva

GENERATION 1

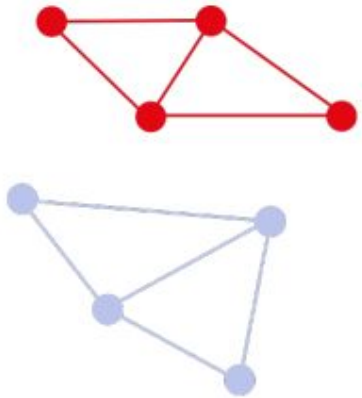
Public blockchain

- *Bitcoin / Ethereum*
- *Poor privacy*
- *Network inefficiency*

GENERATION 2

Quorum

IBM HYPERLEDGER

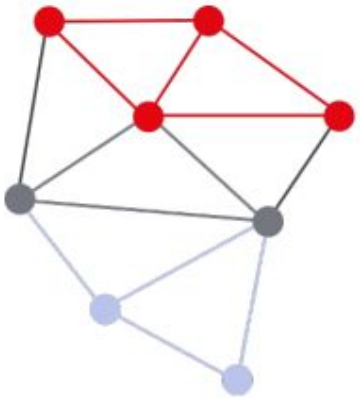


Siloed private blockchain

- *Multiple Siloed Private Networks*
- *Fabric / Quorum*
- *Stranded assets*

GENERATION 3

c·rda



Next-gen blockchain

- *Private but interoperable business networks with transferable assets*
- *Enables Delivery Vs Payment (DvP)*

Courtesy : R3 Training

rda: the 3rd Generation Blockchain

Blockchain for business

Corda is the world's only blockchain platform built specifically for businesses that offers privacy, scalability and interoperability

Applicable to all industries

Designed to meet the standards of one of the most complex and highly regulated industries in the world, Corda can be applied seamlessly to all other areas of commerce

Cross Industry ecosystem

Blockchain benefits are best realized when different industry participants come together to create a shared platform. R3 offers a thriving network of 200 + companies embracing this technology to solve real-world problems

Courtesy: R3 Training

Corda is open source! github.com/corda/corda

corda

Strong Identity

Performance & Scalability

Privacy

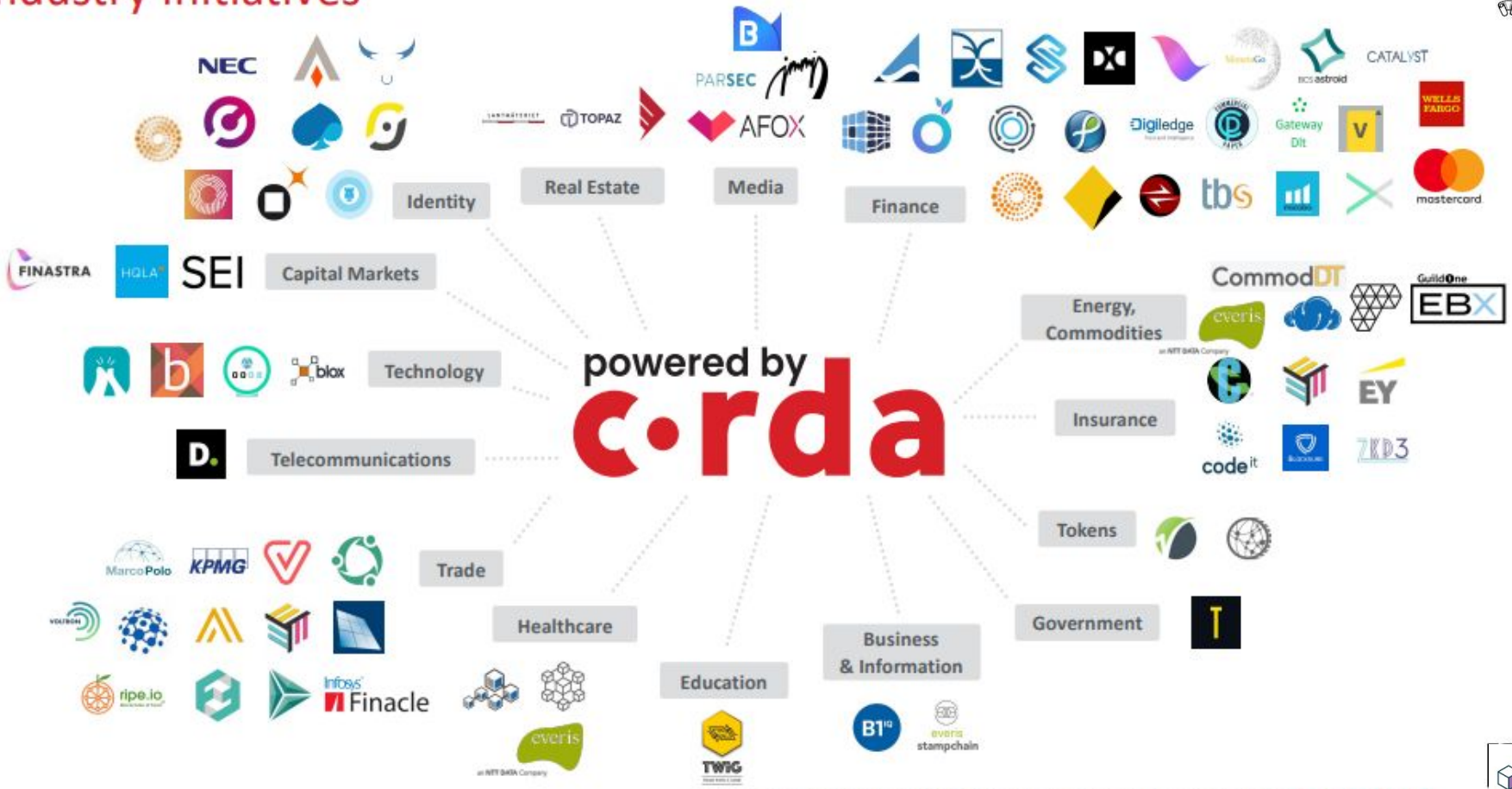
Global Interoperability

Consensus

Open Source & Network

Courtesy: R3 Training

Industry Initiatives



Illustrative example. For a full list of partners and applications visit marketplace.r3.com

inastra – Fusion LenderComm

Fusion LenderComm digitizes communication with lenders – driving efficiencies in the process, saving agents time and money, and eliminating operational risk.

Industry problem

- Coordinating agents in the syndicated lending process is a timely and complicated procedure
- Syndicated lending is currently a paper-based process

Fusion LenderComm use case

- Fusion Lendercomm solution aims to connect lenders across the industry while digitizing the syndicated loan process

Corda Solves

- Highly secured Corda nodes maintain all digitized transaction history
- Provides every lender a personal view of their own deals
- Each message is time-stamped and provides a personalized audit trail

Benefits of LenderComm, powered by Corda

- Seamless collaboration between agent and lenders
- Fully automated, secure communication with lenders
- Real-time data
- Cloud-based technology for quick and easy adoption

Developed in collaboration with some of the world's top banks

STATE STREET
BNY MELLON
BNP PARIBAS
ING
HSBC

Courtesy : R3 Training

radewind Markets

First Production Example of a Digital Asset Backed by Regulated Custodian – all settled via Corda

Industry problem

- Commodities large capital investment limits market accessibility
- Banks that trade physical commodities face costs and frictions from antiquated post trade systems

Tradewind's Vaultchain use case

- Precious metals investors to execute trades with a secure, low-cost solution
- Physical gold and silver available today with platinum and palladium to follow

Corda Solves

- Immutable records of ownership
- Direct balance verification on Corda
- Flexible account and inventory management
- Connectivity by API and Web user interface

Benefits of Vaultchain, powered by Corda

- Increased investor pool
- Reduced post trade costs & friction
- Vaults & refiners can easily interact with customers & investors
- Increased insight into market physical demand & pricing

The ownership of precious metals is going digital

Tradewind's platform will lead the transformation, allowing physical commodities market participants to adapt quickly and easily.

Courtesy: R3 Training

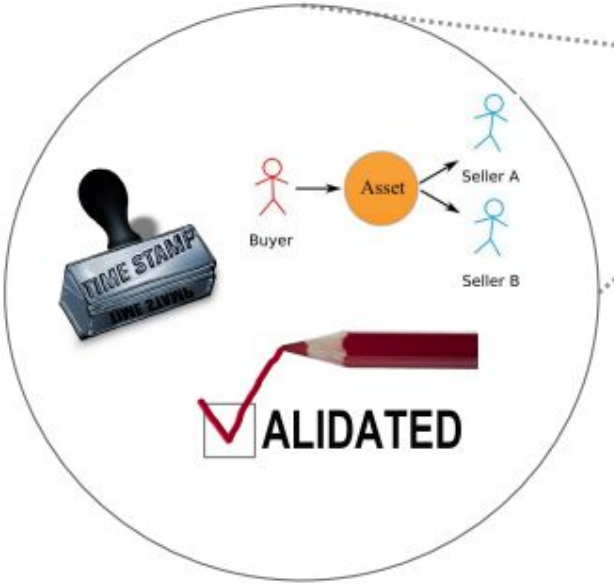
roven blockchain use cases

Cash	Identity	Insurance	Trade Finance	Assets
▪ Domestic CBDC ▪ Cross Border CBDC ▪ Cash on Ledger	▪ KYC Sharing ▪ Self Sovereign Identity ▪ Self Sovereign Connectivity	▪ Policy Placement ▪ Underwriting ▪ Claims Processing	▪ Documentary Trade Services ▪ Open Account Services	▪ Syndicated Loans ▪ Securities ▪ Gold Royalties

At R3, we have run +100 POCs across numerous industries and sectors.

Courtesy: R3 Training

Corda is a permissioned network that provides P2P communication on a need-to-know basis



Notary

Colorado River Authority

Address: 17.149.112.236:10005

Public key: 5h54h5wv632vhy55

Titan Technology Partners

Address: 115.187.28.40:10005

Public key: 5hw03nnk43jknkj4n

Bays Transcorp Inc.

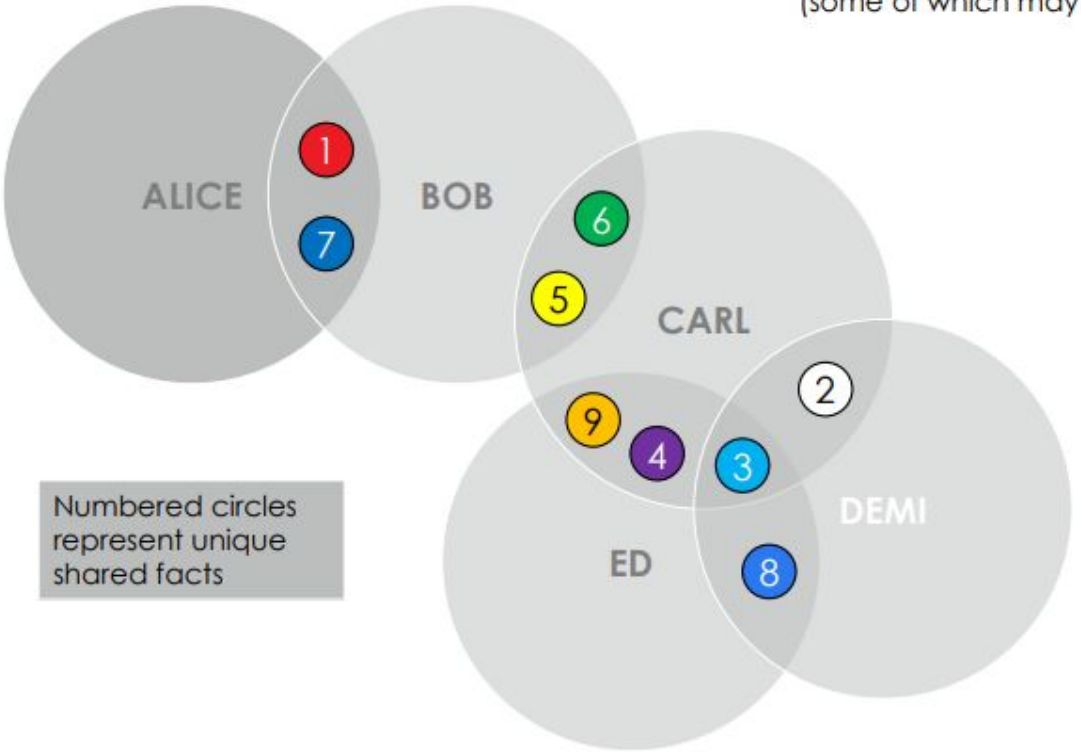
Address: 120.65.158.170:10005

Public key: t453wv84bvt3cj5w3h

Courtesy: R3 Training

The ledger from each peer's point of view is the union of all intersections with other network peers

(some of which may be the empty set)



ALICE = { 1 7 }

BOB = { 1 7 5 6 }

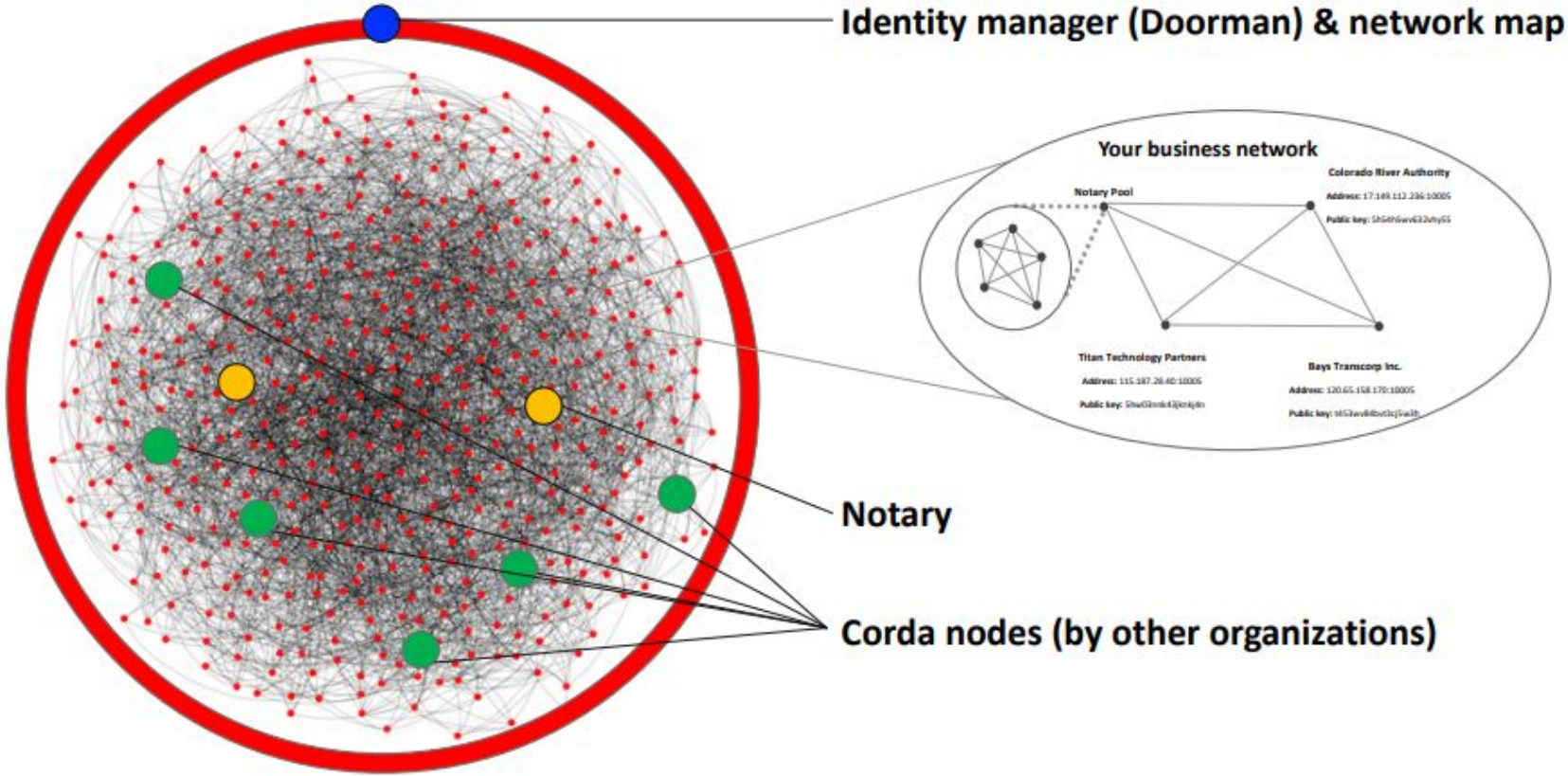
CARL = { 2 3 4 5 6 9 }

DEMI= { 2 3 8 }

ED= { 3 4 8 9 }

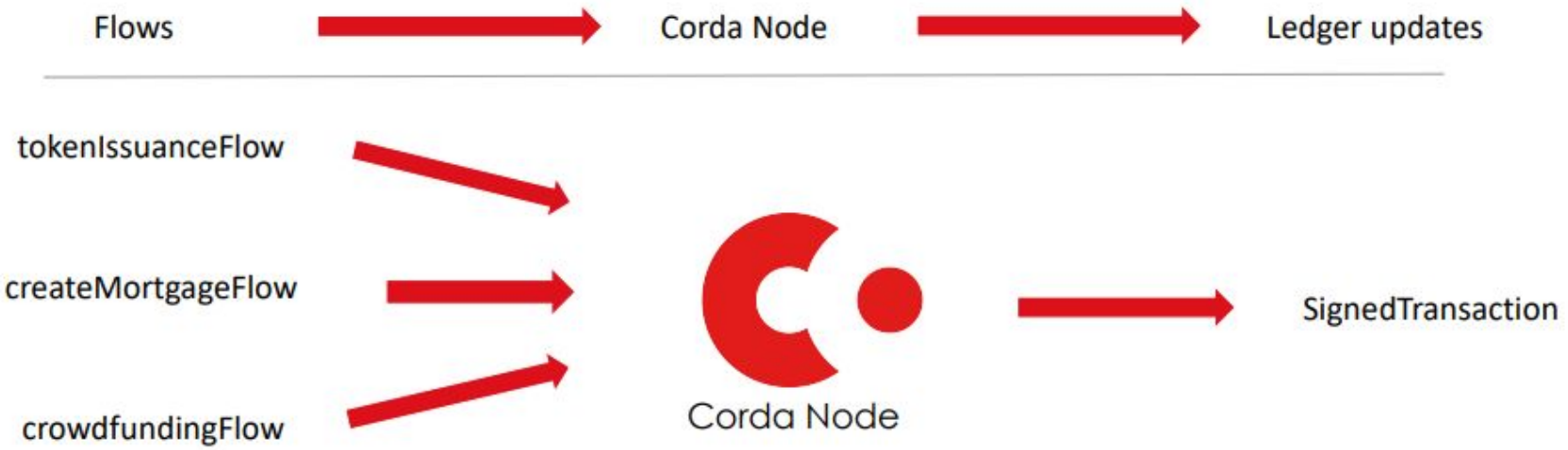
Courtesy: R3 Training

Corda Network



Courtesy : R3 Training

orda nodes abstract away the complexity of updating the ledger



The Node helps abstract away:

Messaging | Storage | Peer discovery | Data distribution

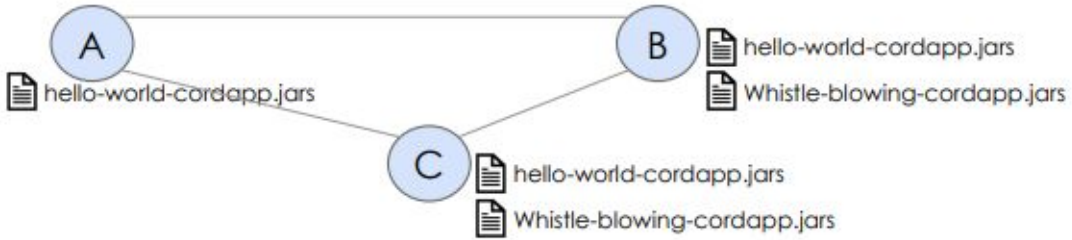
Concurrency | Disaster recovery | Key mgmt. | *and more!*

Courtesy: R3 Training

CorDapp

Corda - Decentralized - Application

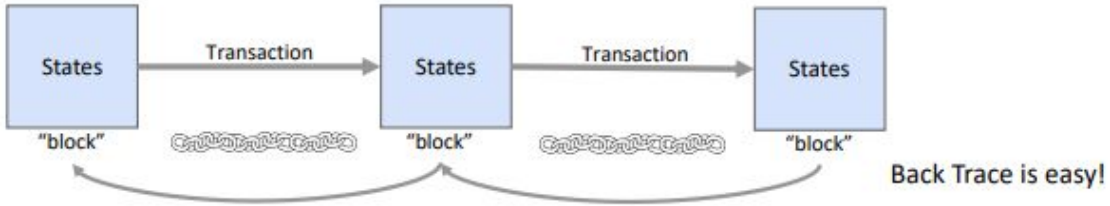
- **Decentralized Application**: computer application that runs on a distributed computing system. It is also sometimes referred as smart contracts.
- **CorDapps** are binary jars that are stored inside the Corda nodes, and each node can carry multiple CorDapps.



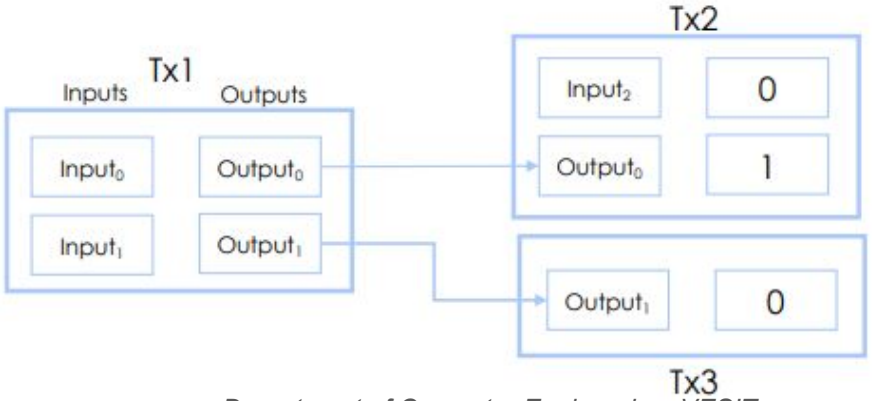
Courtesy: R3 Training

low "blocks" chain up in a CorDapp…

- Data are stored as States in Corda node's database. And States are updated via transactions.



- Corda adopts the UTXO (Unspent Transaction output model), so data is never deleted from the database. Hence, Corda holds the immutable nature of DLT system.

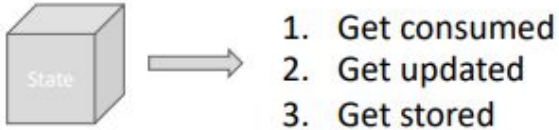


Courtesy: R3 Training

Components of a CorDapp (Smart Contracts on a Corda network)

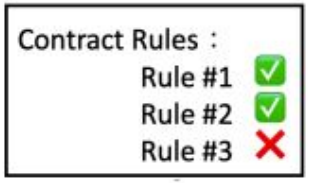
1. State:

   The object in Corda



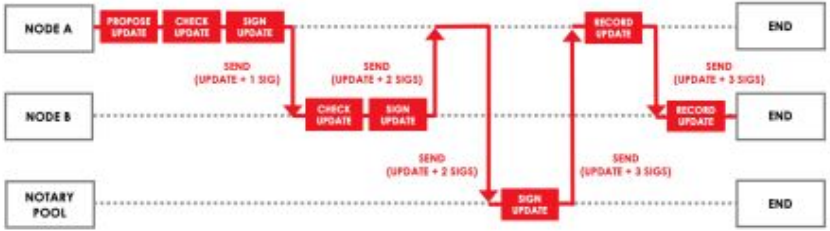
2. Contract:

   Verify the transactions



3. Flow:

   Execute the business logic



Courtesy : R3 Training

Key Concepts in Corda

- Corda as a network of interconnected graphs.
- Nodes.
- States.
- Bilateral ledger.
- Transactions.
- Contracts.
- Commands.
- Timestamps.
- Attachments.
- Flows.
- Consensus.
- Notary services.
- Oracles.
- CorDapps.
- Node services.
- Corda networks.

Courtesy: R3 Training

Key Concepts in Corda

Corda Network

- Corda networks are **fully connected graphs with connectivity from every peer to every other peer.**
- There is **no global broadcast or gossip.**
- **Peers communicate reliably, asynchronously, and securely using AMQP over TLS** (think SMTP).
- A **Network Map service** publishes a list of peers (think directory service).

Nodes

A node is a piece of software that:

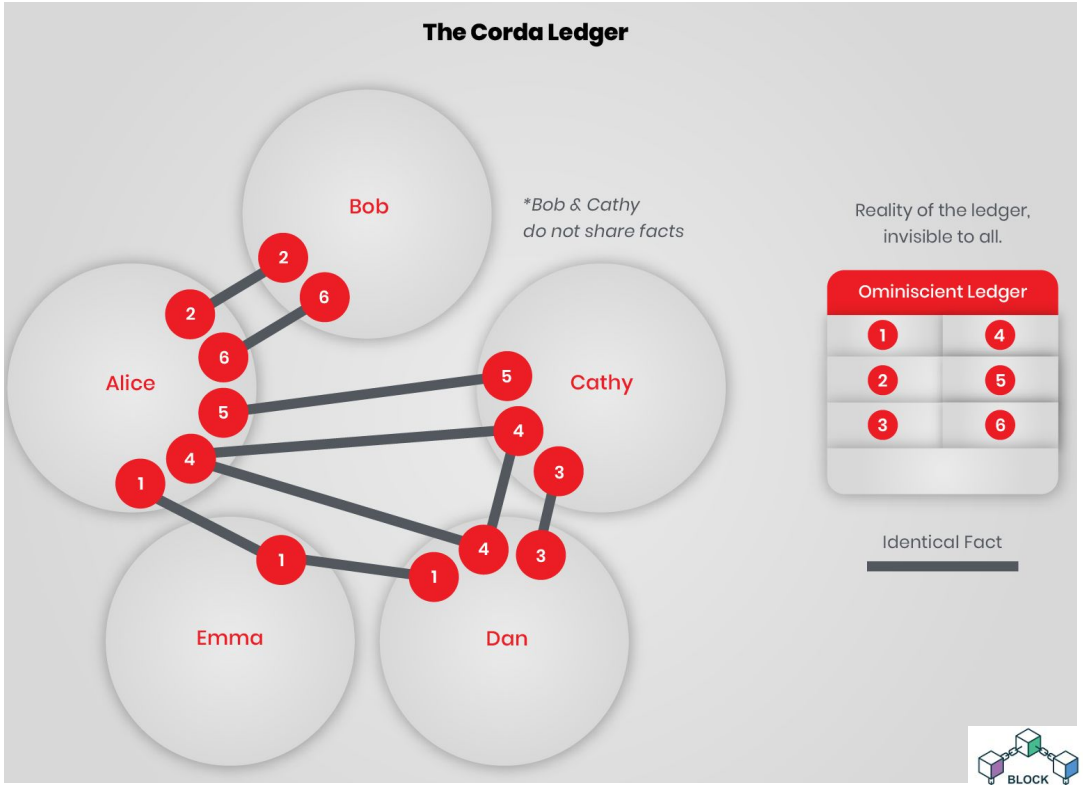
- participates in a Corda system or network,
- and runs Cordapps.

Courtesy : R3 Training

Key Concepts in Corda

States

- The aggregate of all states held by all nodes of the network is the **distributed ledger state**.
- There is **no central ledger, and not all nodes know all states, so the overall ledger is subjective from the perspective of each participant.**
- Most **states are found in / on at least two different nodes**.



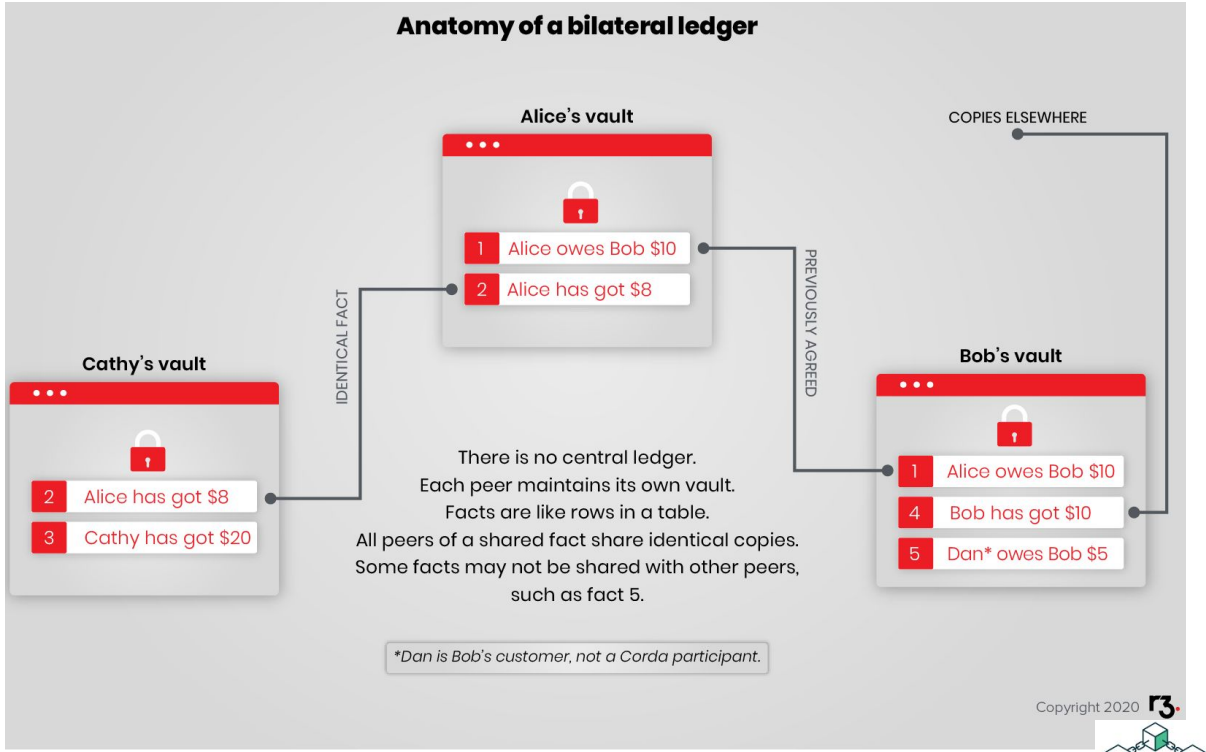
Courtesy : R3 Training

Key Concepts in Corda - States

Bilateral ledger

- **Each node maintains its own *vault* of states.**

- In this example, Alice and Bob both have copies of **their shared states as well as states that are known only to themselves.**



Courtesy : R3 Training

Key Concepts in Corda - states

states are nodes of a **UTXO (Unspent Transaction Output) DAG (Directed Acyclic Graph)**.

Only the "**unspent" states define the current state of the world.**

The **"spent" states are part of the history of the world state**.

States are **unique and can only be spent, or consumed, once.**

States define data.

Here, **Cash is an example of a state**. "Alice has $10" is a state. If Alice spends $1 in a transaction, then the original state ("Alice has $10") is replaced by a new state ("Alice has $9") plus another state because someone else, let's say, Bob has the other dollar. Only slightly more formally, the other state is "Bob has $1".

To help you get accustomed to the Corda way of describing these state transitions, let us recap Alice sending Bob $1.

- Old State: Alice has $10 (consumed)
- New States: Alice has $9 and Bob has $1 (created)

Courtesy: R3 Training

Key Concepts in Corda

Transactions

- Transactions are what **consume states and produce new states**.
- They are **atomic.**
- Transactions **either complete entirely or have no effect**.
- There are **no partially complete**, “in-flight” transactions although they do have a lifecycle with various stages.
- A transaction in Corda is not like in SQL or Ethereum where a command and parameters are passed along, and the state machine “decides” what is the new state.
- Instead, a **transaction references existing input states and future output states**.
- **A transaction that is not yet accepted by the parties** is called a **proposed transaction.**

Courtesy: R3 Training

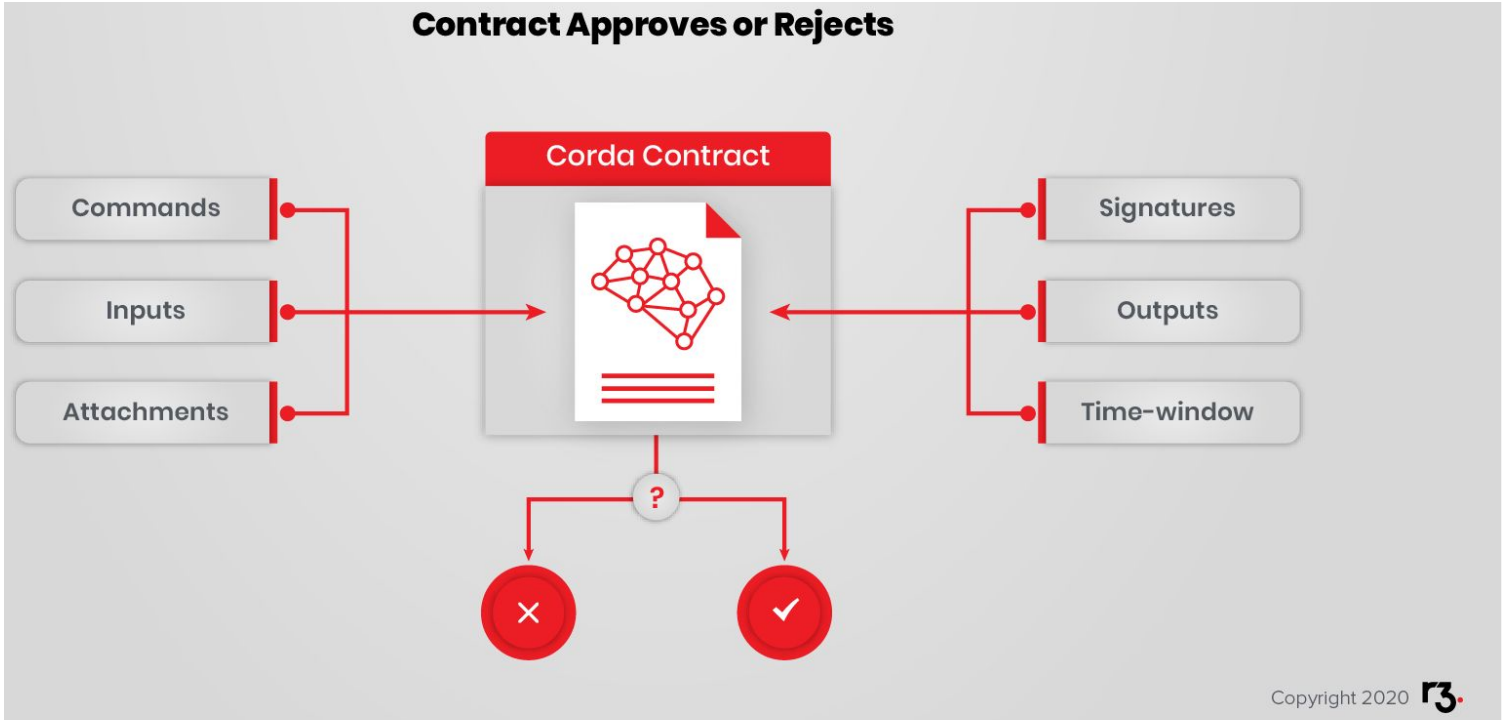
Key Concepts in Corda

Transactions

Eg. Suppose Alice will send all $10 to Bob. A transaction can reference an input state “Alice has $10,” and an output state “Bob has $10”. The transaction consumed “Alice has $10” so that becomes a historical state. It is part of the past and it explains the history of Alice’s account. It is gone because Alice sent it to Bob. The layman’s description of the **transaction is that cash has changed hands**

Courtesy : R3 Training

Key Concepts in Corda



Courtesy: R3 Training

Key Concepts in Corda

For the cash example,

- the **contract** could **enforce conservation of funds by ensuring that the cash amount of the input is the same as that of the output**. So, you could not have a transaction that had as input “Alice has $10” and as output “Bob has $9”.
- A **simple cash contract** would **enforce a conservation rule stating that the sum of the inputs must equal the sum of the outputs**. Such a structure is sufficiently flexible to apply the rule to more complex situations.
- A **transaction** could, for example, **propose to take $100 from Alice, $200 from Bob and then give $150 each to Carol and Dave.** The **contract would ensure that all the cash inputs equal all the cash outputs.**
- Notice that the cash contract is silent on the **reason for the transaction.** The **cash contract is unconcerned with the reason for spending**. It is concerned with **accounting and ensuring that cash can’t be created or destroyed** (the issuance concern is set aside for simplicity). Contract rules define the assets that are mentioned in states such as “Bob has $10.”

With **states, transactions, and contracts,** Corda has the **basic elements of data and transitions.**

Courtesy: R3 Training

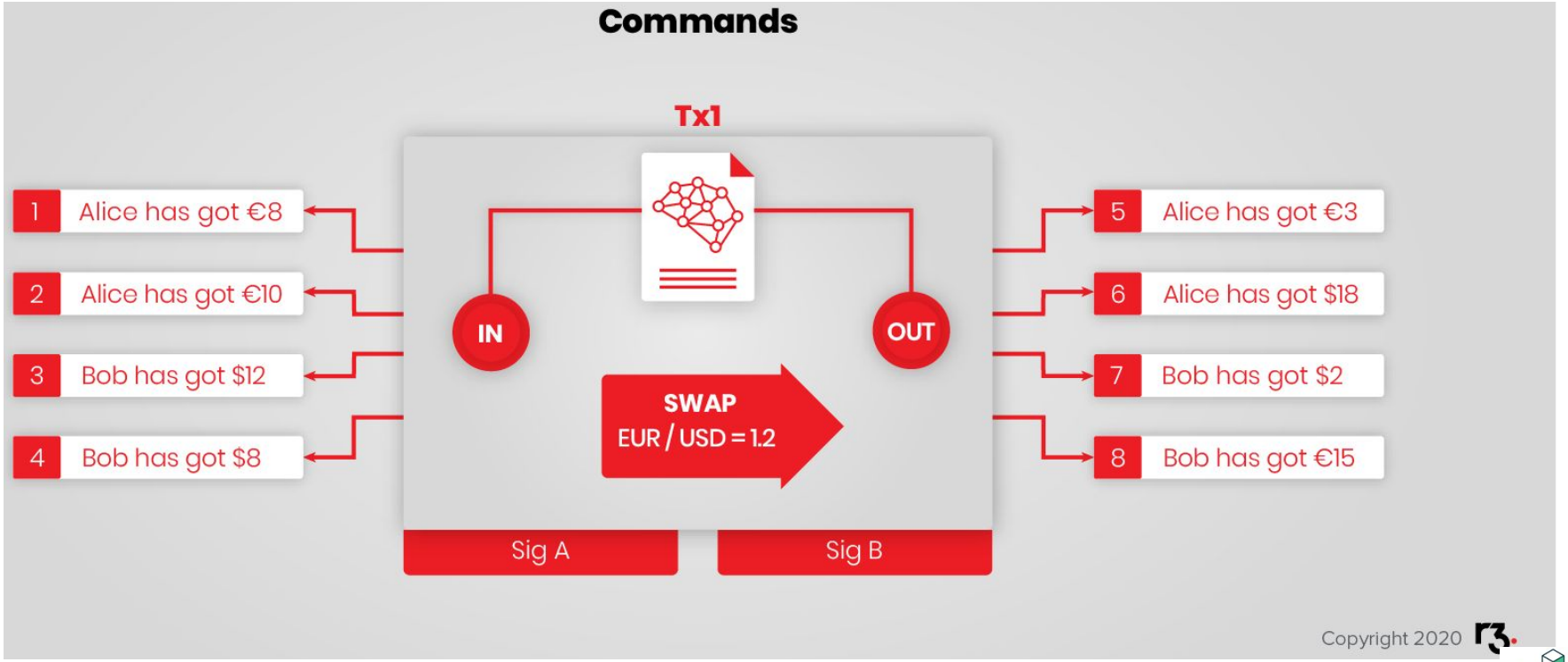
Key Concepts in Corda

Commands

- Commands are **part of a transaction and give them intent.**
- Intent **helps guide interpretation.**
- Using the cash example, if “Alice creates $10” out of thin air (let us suppose she is entitled to), you would call this “minting” or “issuance.” If you consumed this state and output “Bob has $10”, you would call this “paying” or “moving.” If you had no output state, you would call this “burning”, “retiring” or “redeeming.”
- Commands **signal the intention of a proposed state transition which helps the receiver classify received transactions and move on to considering the proposal.**
- **Smart contracts** too, **can branch their verification logic based on the command.**
- Commands **parameterize transactions by hinting at their intent, collecting the inputs, and specifying the list of required signers (by their public keys).**

Courtesy: R3 Training

Key Concepts in Corda



Courtesy: R3 Training

Key Concepts in Corda

Timestamps

In Corda, Timestamps **assert that something happened within a specified window**.

These windows can be **open or closed.**

For example:

- Between the earliest time and the latest time.
- Before the deadline time.
- After the commencement time.

Never:

- At the exact time.

Time ranges **map to our understanding of real-world business processes and Legal Prose that is (presumably) enforceable in law courts**. For example, an offer open to acceptance before a certain deadline becomes invalid quite naturally as the overseeing verification rules disregard acceptance if it arrives after the deadline.

Courtesy : R3 Training

Key Concepts in Corda

Attachments

- Corda **transactions may include attachments that are included with the transaction** but are not part of the transaction itself.
- These are `.ZIP` or `.JAR` files attached to the transactions.
- Attachments can include:
  - **Legal Prose** with the template and parameters
  - **Data files** that support transactions such as calendars, currency definitions, or even financial data
  - **Contract code and associated state definitions** (`.class` files) that define the transaction states mentioned

Courtesy: R3 Training

Key Concepts in Corda

Flows

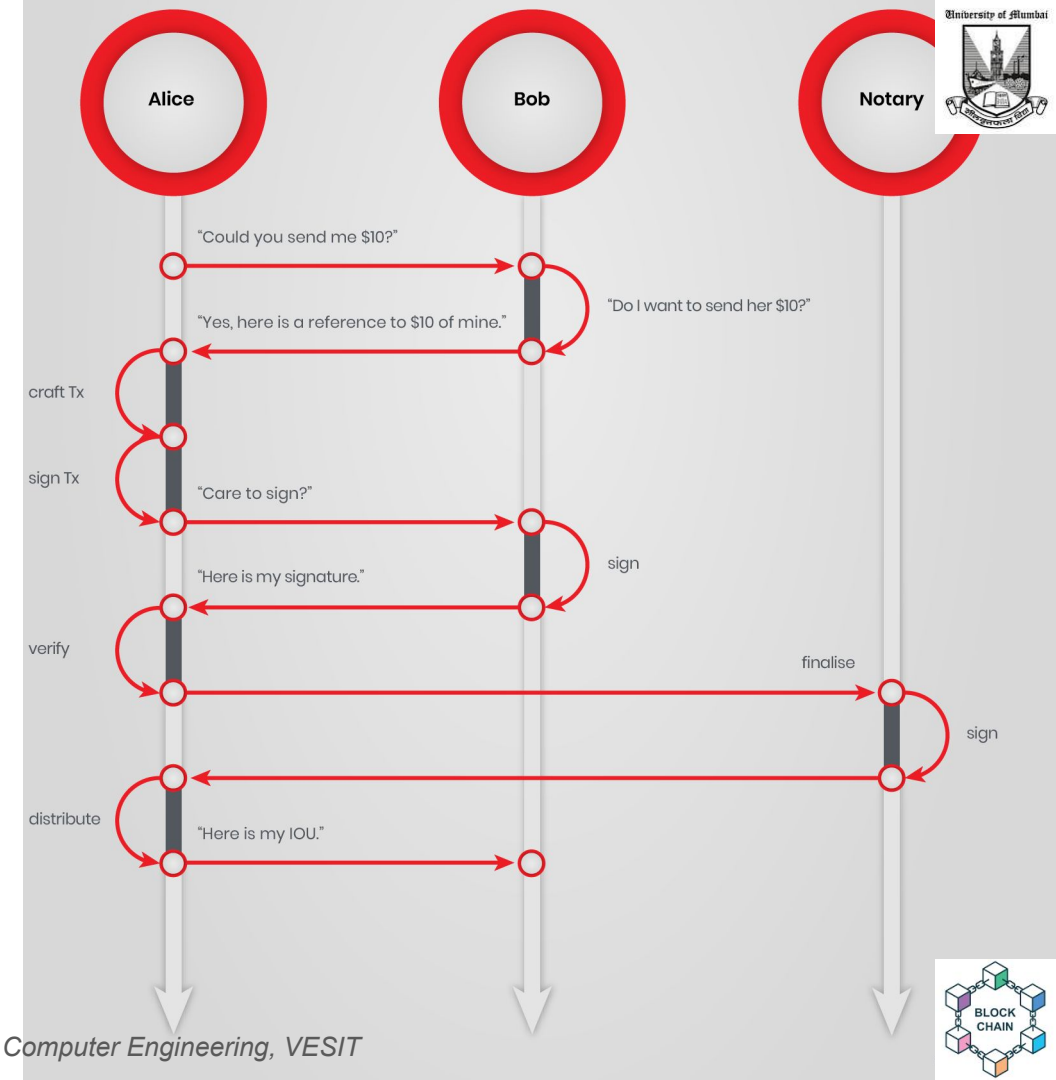
- Flows **model the business processes that oversee the evolution of transactions states.**
- Flows can be **created in order to be widely reused or created ad hoc.** Nothing prevents Bob from agreeing on and preparing a different flow in order to issue an IOU to Carol. In the case that Alice defaults, or a court of law orders the parties to agree on an unforeseen transaction, a flow made for the situation can be created then and there, as long as the contracts allow the proposed state changes.
- Flows can be **more complex and can include more than two parties**. For example, if regulatory oversight is required, then the flow would indicate a third step to inform the regulator.
- Flows are **light-weight processes that coordinate multi-step business processes that help peers reach consensus about shared states - their mutual understanding of what happened.**

Courtesy : R3 Training

Key Concepts in Corda

Consider the **simple example of Alice's IOU.**

Alice first proposes the IOU which includes reference to the IOU contract that informs the contract state to be created, as well as a proposal that Bob will credit Alice's account with some currency. There is a **process of gathering approvals before this proposed transaction is final.**



Courtesy: R3 Training

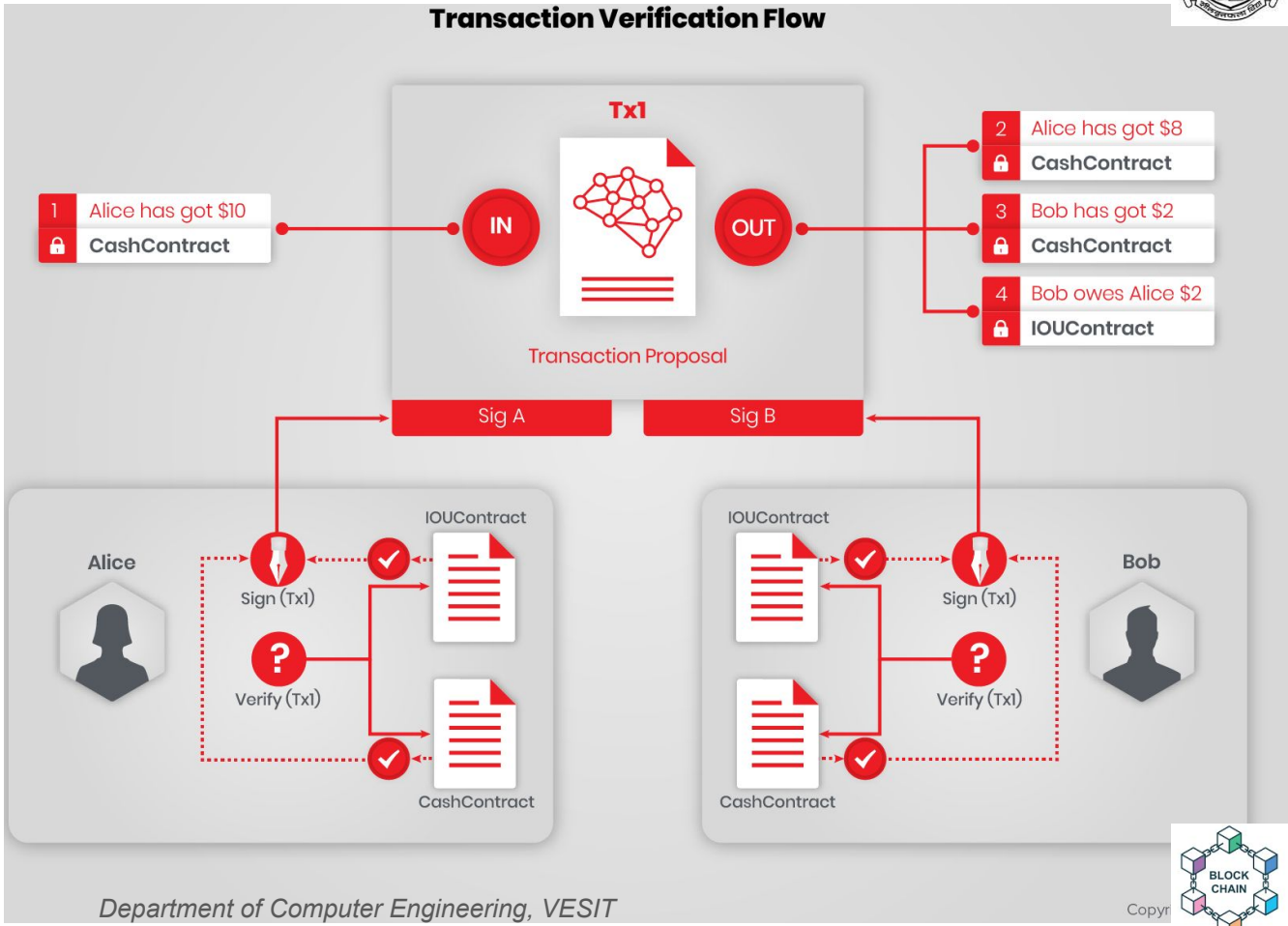
Key Concepts in Corda

Consensus

- Consensus is the **process by which all parties achieve certainty about the shared states.**
- Corda applies two types of consensus:
  1. **Validation Consensus :** when all peers achieve certainty that a transaction is signed by all peers listed in the commands and satisfies all constraints defined by the contracts referred to by the input and output states. To say that another way, **it *can be* done, and everyone agrees it *should be* done.**
  2. **Uniqueness Consensus** is when peers reach certainty that the output states generated by a transaction are the *unique successors* to the input states referenced by said transaction. This is **how Corda prevents double spending.**

Courtesy : R3 Training

Key Concepts in Corda



Courtesy : R3 Training

Key Concepts in Corda

Suppose Alice has $10. More precisely, suppose Alice has a checking account contract with the Bank, and the current unspent state of the contract says Alice has $10. When Alice spends $1 this transaction will consume the existing state of Alice's checking account and generate a new one that indicates Alice has $9. There would, of course, be other related details such as what happened to the dollar she spent. That concern is set aside to keep the focus on Alice's balance.

Suppose Alice was able to reference the input state of $10 again. Alice would be able to spend more money than she has. But, the original state has a successor state that indicates that Alice has $9. The $9 state is a successor to the historical $10 state in the history of Alice's account. The rule is that a state can only be consumed one time. Or, said another way, that the output states created are the unique successors of the input states.

Notaries help confirm uniqueness.

Courtesy: R3 Training

Key Concepts in Corda

Notary Services

- Notaries are **comparable to traditional Notary Public services that provide reliable witness to events.** In simple terms, Notaries maintain a key map of input states and the transactions that consumed them. They need not know the content of states and transactions; **they only need a reliable way to uniquely identify them.**

- In Corda, **every transaction is Notarized along with the peer that requested Notarisation and the transaction that marked the input state as historic.**

- In the simplest implementation, the Notary Service is implemented as a **fault-tolerant service and all the peers will use the same Notary for every transaction.**

Courtesy: R3 Training

Key Concepts in Corda

Notary Services

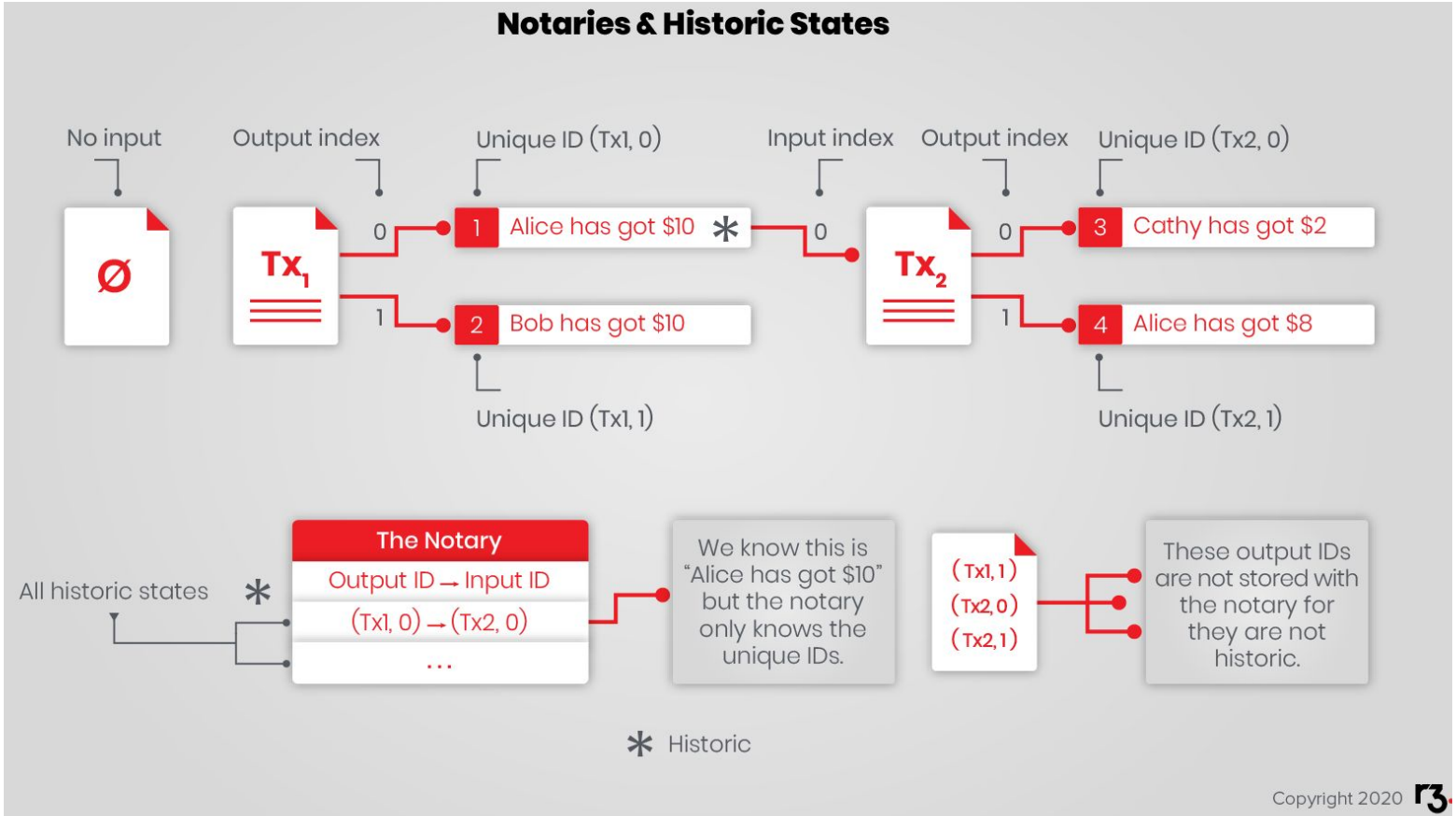
When a peer sends a transaction to the Notary Service, one of two things will happen. If any of the input states are already referenced in the Notary's map, then the Notary will throw an exception. If none of the input states are known to have been previously consumed, then the Notary adds each input state to the map and signs the proposed transaction. This process provides uniqueness consensus.

Together with verification consensus, uniqueness consensus provides certainty about transaction finality to all parties to the transaction.

To protect themselves against a potential DoS attack, whereby an attacker submits bogus transactions, a notary can be made to *verify* transactions. In this case, the notary will act as a regular node, and accept the transaction only if its smart contracts approve. This possibility implies that the Notary will gather more information than a map of state keys that have been consumed and this hints at reasons why a Corda network may need more than one Notary. There is a trade-off between contract-level validation at the Notary level and data-leakage.

Courtesy: R3 Training

Key Concepts in Corda



Courtesy: R3 Training

Key Concepts in Corda

Oracles

- Oracles provide **authoritative information about the external world.**

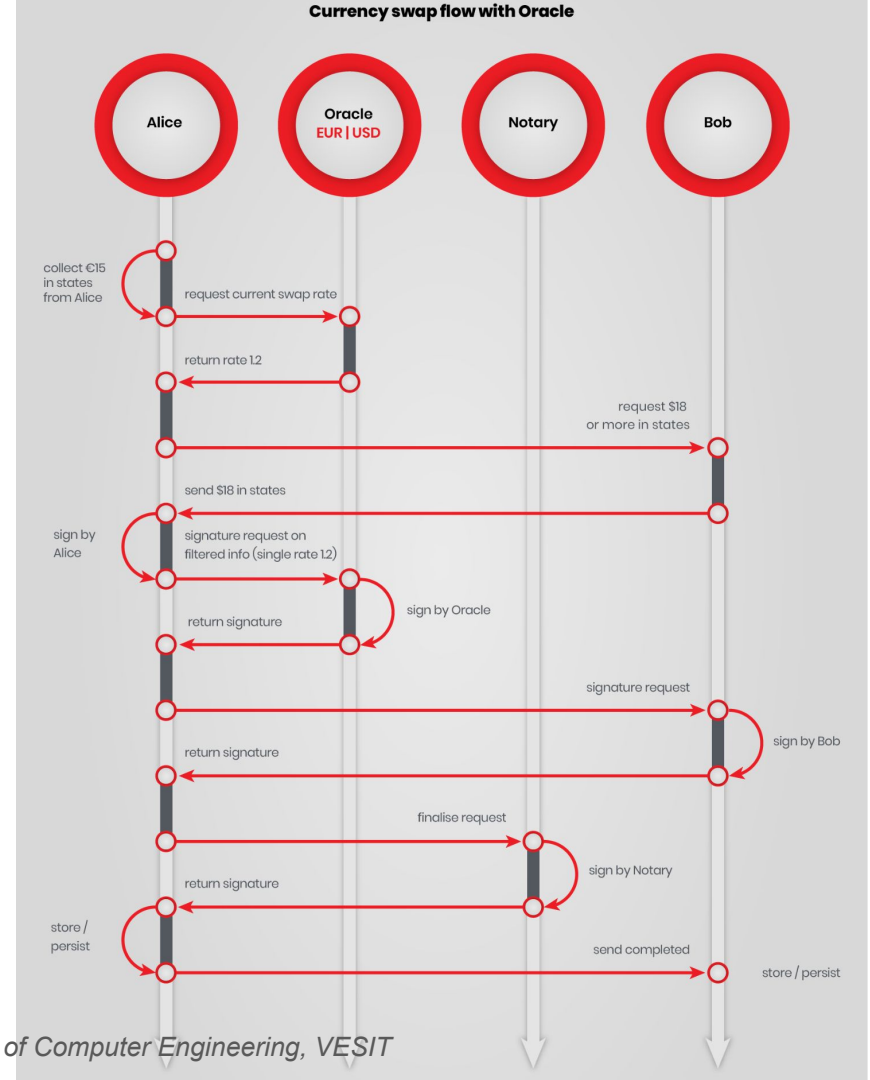
- For example, consider a currency swap. We'll say Alice wishes to trade with Bob 15 Euros for 18 Dollars. Is that fair?

- The parties can agree to use a third party quote to establish a rate of exchange. This third party is known as an Oracle. The Oracle would provide the rate of exchange to Alice, then Alice would use the "swap" command to parameterize a transaction proposing that she sends 15 Euros to Bob, Bob sends 18 Dollars to Alice, *and* the current rate is 1:1.2. Alice, Bob *and the Oracle* are listed as required signers in the swap command.

- In effect, the oracle, through its signature, vouches for the information that is included in the transaction. It may also provide valid information on demand, such that a node has an easier time building a transaction that the oracle will approve. Additionally, with a goal of additional privacy, it is possible, via the use of a Merkle tree and of a Merkle proof, to send only the minimum required information to the oracle. In Corda, this is referred to as a "transaction tear-off."

Courtesy: R3 Training

Key Concepts in Corda



Courtesy : R3 Training

Key Concepts in Corda

Corda Node and CorDapps

Corda nodes implement a number of services needed to participate in a Corda Network.

CorDapps

CorDapps are the distributed applications built on Corda. Notice that these are implemented as extensions to the Node. The Corda Node is constructed in Kotlin and the source code of the node is open to inspection and extension. Kotlin itself is based on Java. CorDapps can be coded in Java or Kotlin.

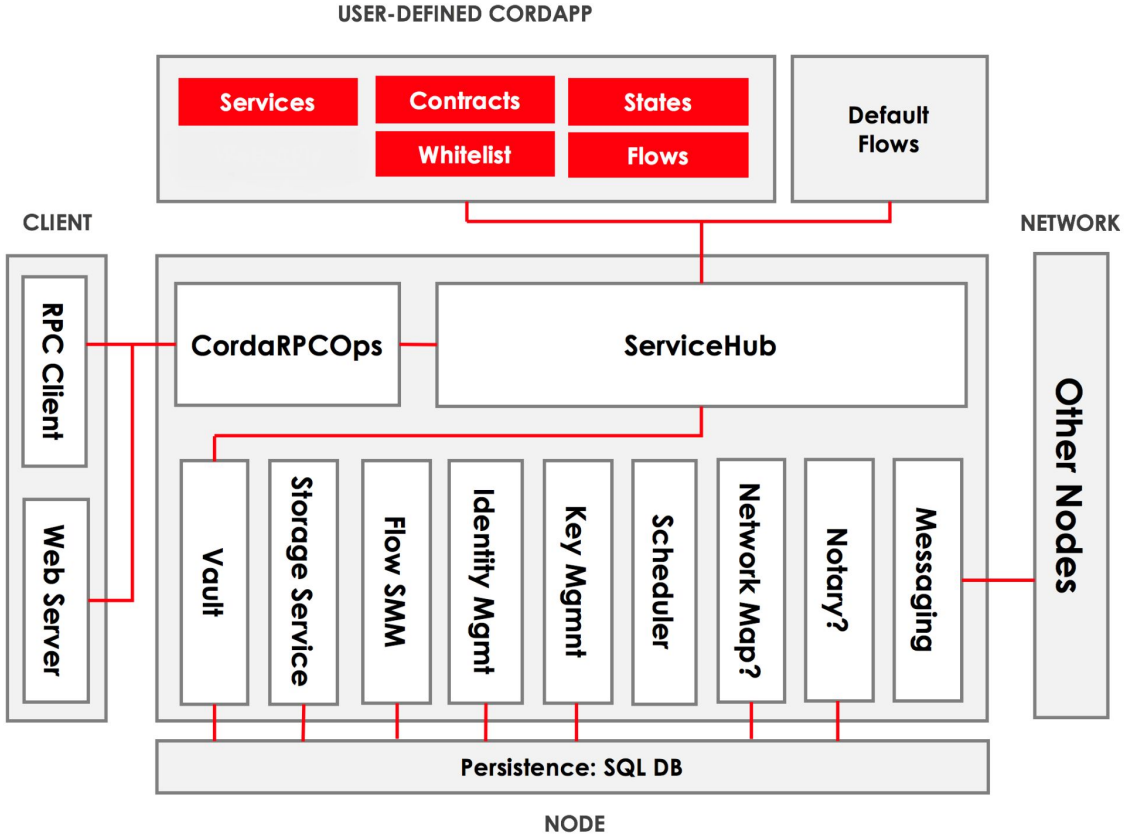
CorDapps consist of **Contracts, State definitions, Flows, and Services**.

Node Services

Clients connect to the Node via RPC. CorDapps connect via a service interface. The Node includes a messaging service for connecting with other Peers as well as Storage, Identity, and the Vault. The Vault is the storage repository where the node records states as key/value pairs. Below all this is the persistence layer, which is a SQL database.

Courtesy: R3 Training

Key Concepts in Corda



Courtesy: R3 Training

Key Concepts in Corda

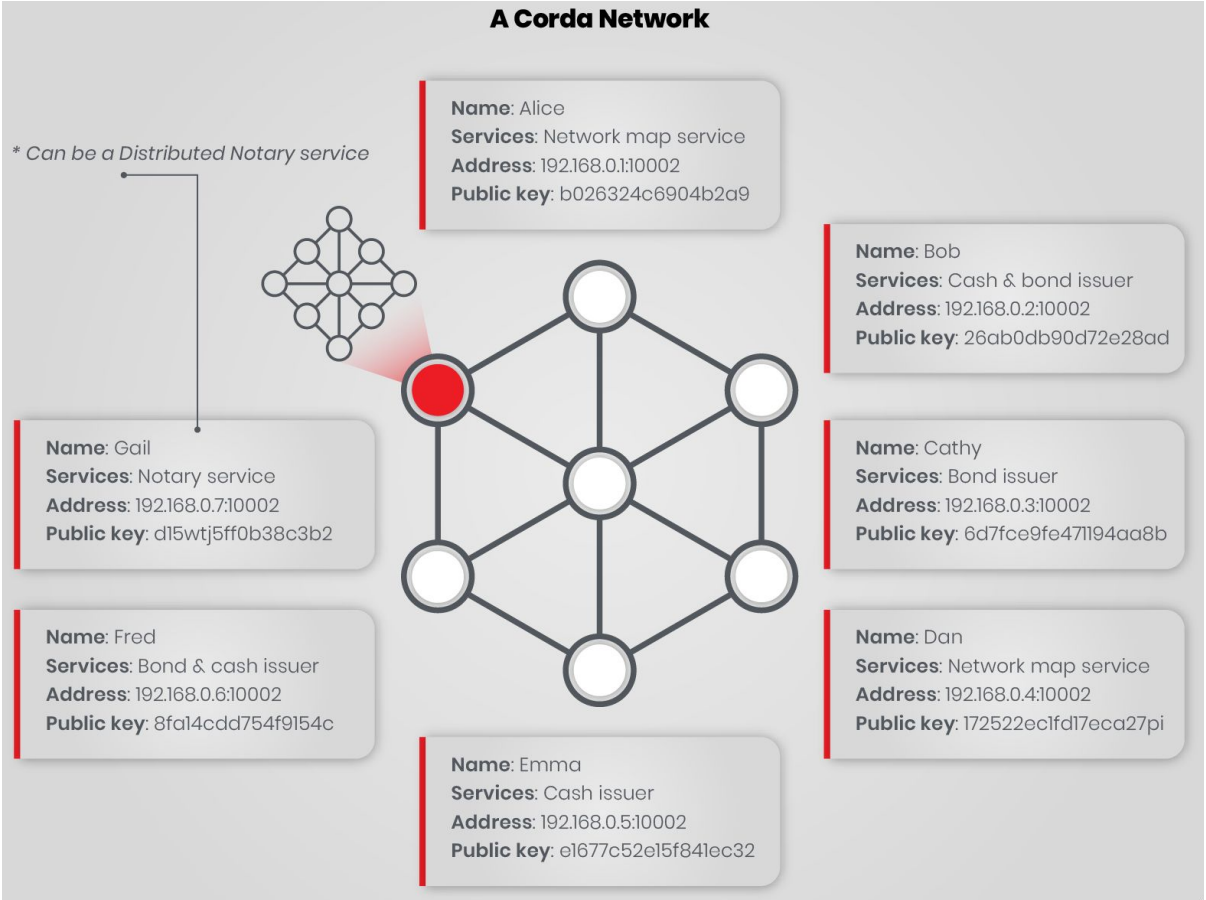
Corda Network

A Corda network is a **fully connected graph**. As well as two or more peer participants there are important special roles that form a complete implementation.

- One doorman which provides permissioning and certificate signing for the permissioned network
- Zero or more Oracles
- One network Map Service
- One or more Notary Services

Courtesy : R3 Training

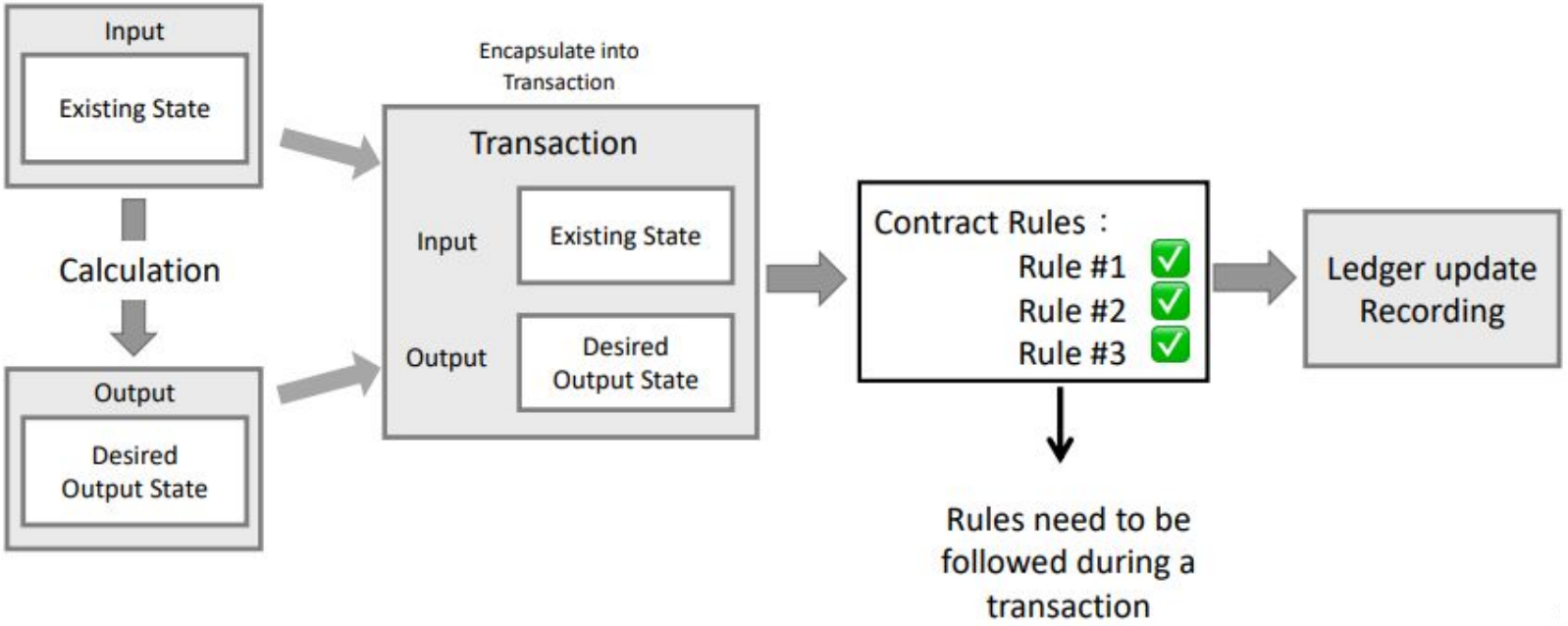
Key Concepts in Corda



Courtesy: R3 Training

Corda Contracts in CorDapp

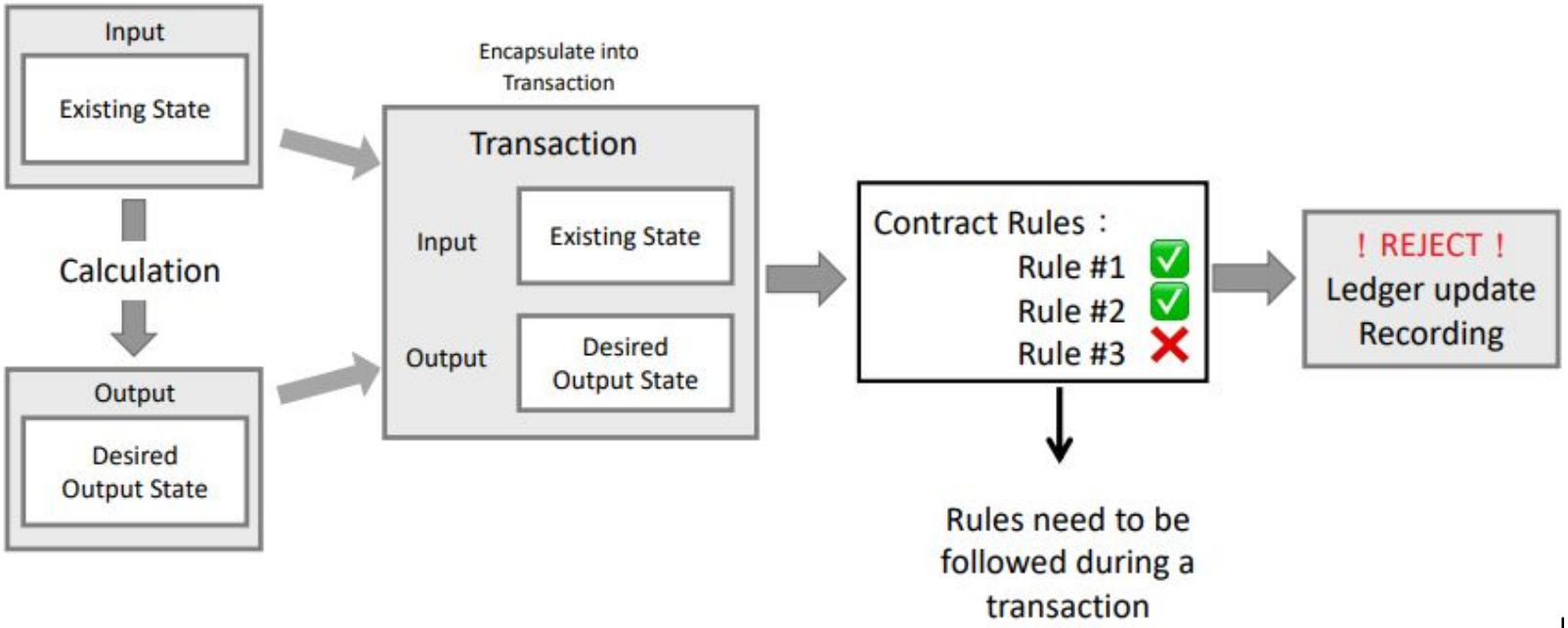
- Ledger update is done through transactions in the flows
- Contracts verify the validity of a transaction: SUCCESS ✅



Courtesy : R3 Training

Corda Contracts in CorDapp

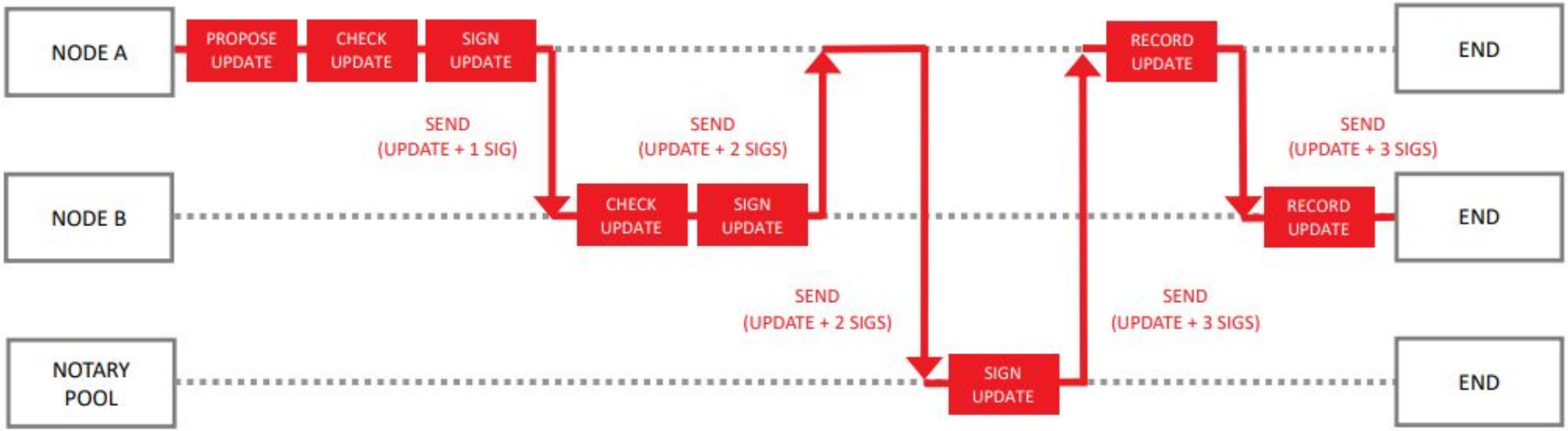
- Ledger update is done through transactions in the flows
- Contracts verify the validity of a transaction: FAILURE ✗



Courtesy: R3 Training

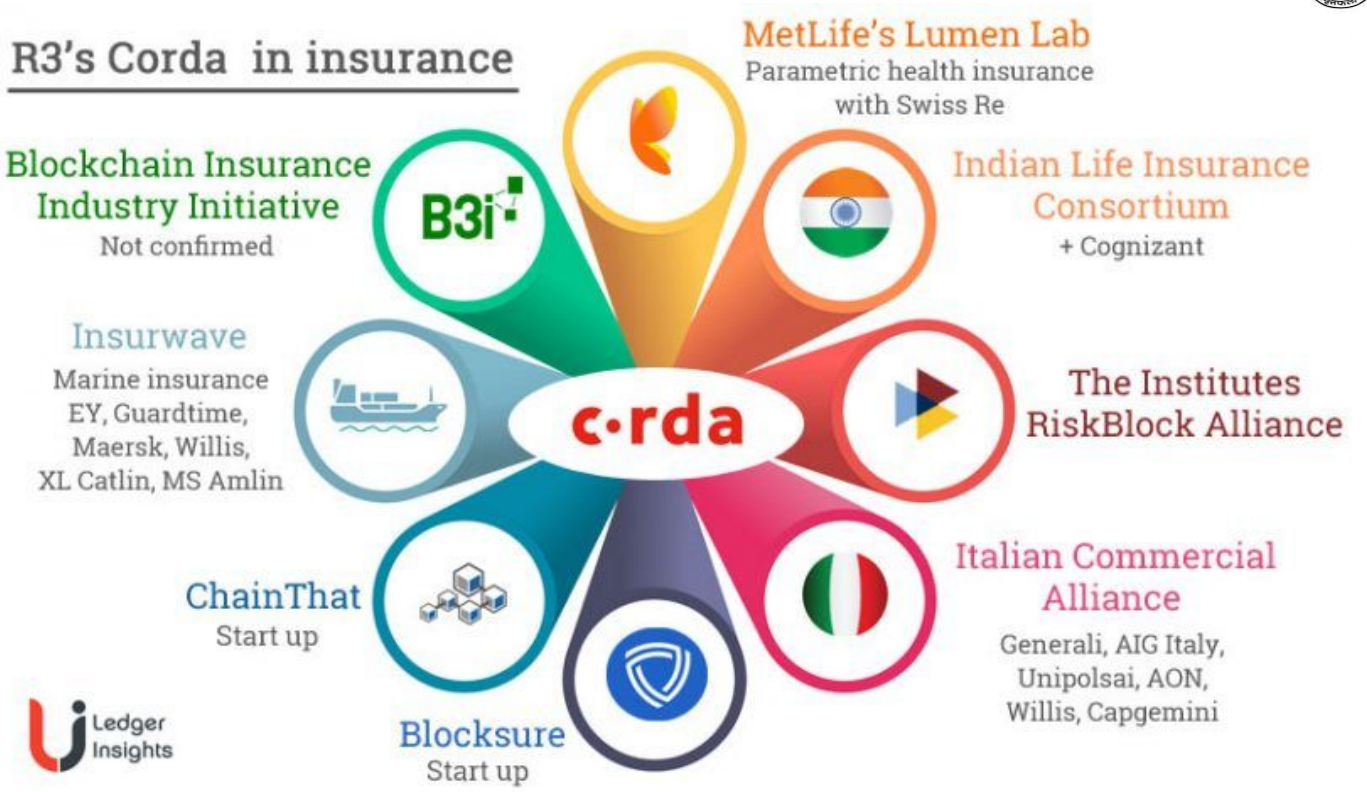
Corda Flows in CorDapp

- Flows execute the business logic
- Flows consist of two classes (Initiator & Responder)



Courtesy: R3 Training

Corda



Courtesy : Ledger Insights

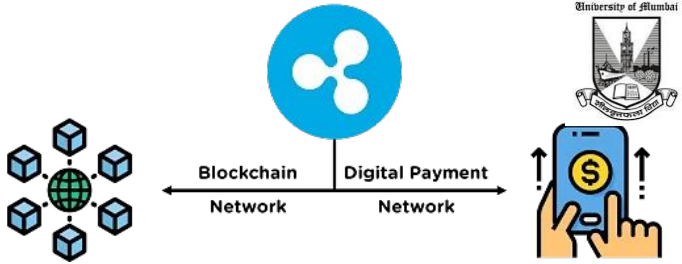
References - Corda

- https://www.corda.net/get-corda/
- https://training.corda.net/
- Slack CordaLedger: slack.corda.net
- Corda docs: docs.corda.net
- Free Training Site: training.corda.net
- Github Repository: github.com/corda
- https://expolab.org/ecs189f-fall-2020/handouts/Principle%20Foundations%20of%20R3%20Corda.pdf
- https://www.youtube.com/watch?v=FHfZ5X7qn1I&list=PLsyeobzWxl7pGh8x5C2hsu3My4ei-eX1Y
- https://www.corda.net/videos/introduction-to-corda-blockchain-for-developers/
- https://www.youtube.com/watch?v=tm06GCD0XJI
- Corda Introductory Whitepaper.
- Corda Technical Whitepaper.
- Kotlin, a massive leap forward.
- Continuations on Wikipedia.
- An essay on Javascript and continuation style programming.

Agenda

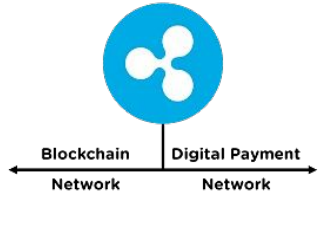
- Corda
- **Ripple**
- Quorum
- other Emerging Blockchain Platforms
- Blockchain in DeFi: Case Study on any of the Blockchain Platforms
- Comparison Tables

What is Ripple ?



- Ripple is a **payment settlement system and currency exchange network that can handle transactions worldwide.**
- Ripple was built to be a **SWIFT (a leading money transfer network)** successor or to replace the settlement layer between large financial institutions.
- Because of the network's rapid ability to confirm that it successfully completed a transaction, it **serves as a trusted middleman between two parties in a transaction.**
- When users use the network to make a transaction, the network takes a **modest charge** in the form of **XRP, a cryptocurrency.**

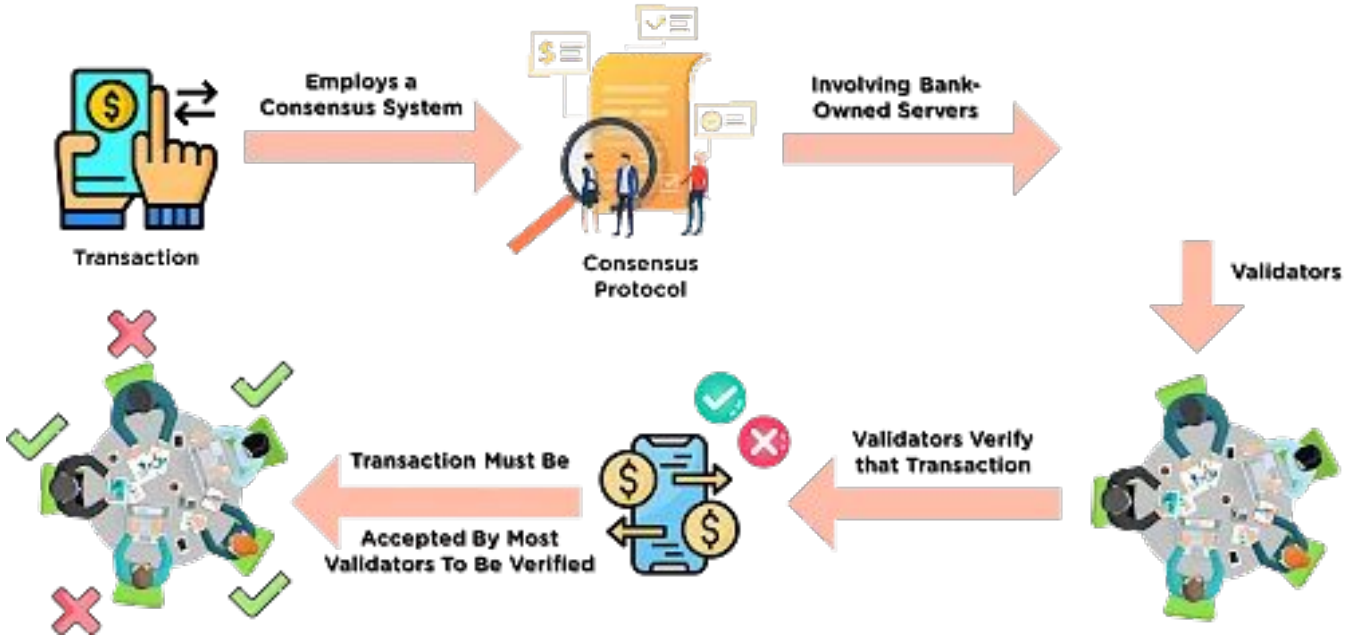
What Is XRP?



- **McCaleb and Britto** would form Ripple, employing XRP to enable Ripple network transactions.
- You can buy XRP to use as a form of investment, as a medium of exchange for other cryptocurrencies, or to finance Ripple network transactions.
- XRP's blockchain, in particular, differs from that of most other cryptocurrencies. For example,
  1. Other cryptocurrencies allow access to their transaction ledgers and verification procedures to anyone who can swiftly solve challenging equations.
  2. On the other hand, since most ledger holders must consent to the verification before a transaction can be added, ripple transactions are secure.
- Additionally, the consensus protocol used by the Ripple network, which uses XRP, is fairly centralized.
- Anyone can download its validation software, but its node lists let users validate transactions based on the parties they think they can trust.

Working of Ripple

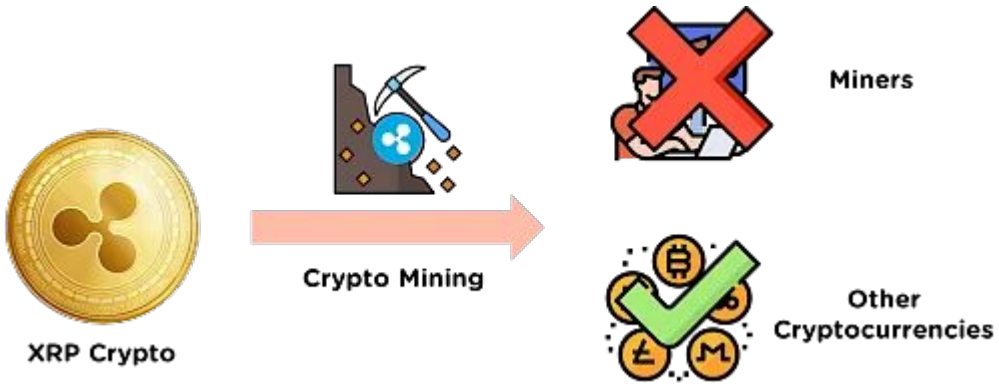
- The XRP crypto uses a consensus protocol to confirm transactions.
- Validators compare proposed transactions to the most recent version of the XRP ledger to determine whether they are valid.
- The majority of validators must accept a transaction to be verified.



How Is Ripple Mined?

Ripple (XRP) is created using a crypto ledger akin to blockchain technology and federated by financial institutions and payment processor networks.

- While it is true that miners cannot mine Ripple (XRP), it is technically viable to do it using other cryptocurrencies.
- Mining Bitcoin (BTC) and Ethereum (ETH) and then exchanging the mined coins for Ripple (XRP) through exchanges is one of the most effective methods for mining XRP.



Ripple	Bitcoin
The consensus protocol used by XRP confirms transactions.	Bitcoin uses mining to certify transactions and distribute new coins.
Its transactions are instantly confirmed and have very cheap transaction fees also, the user is charged a small amount of XRP each time a transaction is completed on the Ripple network.	Considering how difficult and expensive mining is used in the cryptocurrency, its transaction confirmations can take several minutes and are connected with substantial transaction fees.

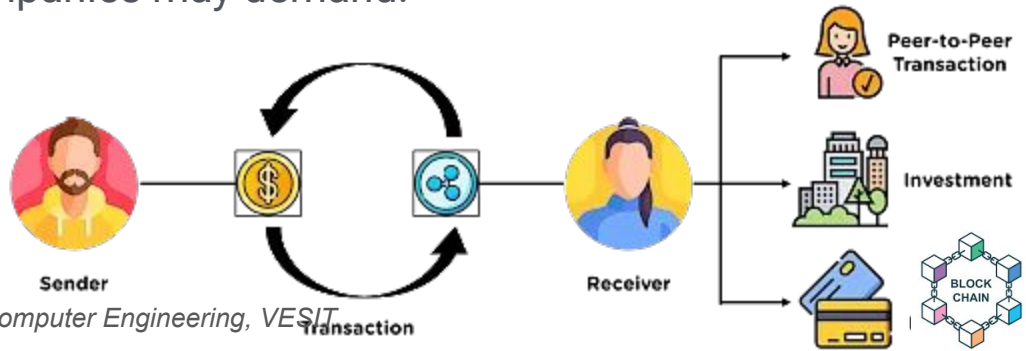
Ripple	Bitcoin
At its launch time, one billion XRP was pre-mined and progressively distributed onto the market by the company's major investors.	The supply of Bitcoin is limited to 21 million, which means there will never be more than 21 million Bitcoin.
A smart contract controls the release of XRP. According to an in-built smart contract, Ripple is expected to distribute a maximum of 1 billion XRP tokens per month; the current circulation is 55 billion.	The network receives newly released bitcoins. Network speeds and the complexity of the algorithm used to manufacture coins mostly influenced their supply, and they have no fixed release date.

Pros and Cons of Ripple

Pros	Cons
XRP transactions are inexpensive and quick.	The Ripple consensus methodology is perhaps less safe than other cryptographic transaction processing systems.
Financial institutions already use the payment network of Ripple.	Ripple's financial partners primarily use RippleNet rather than the XRP cryptocurrency.
Small company owners and individuals can utilize XRP to make safe money transactions.	Because a private corporation administers it and because of the SEC litigation, Ripple has sparked debate.
It may be used as a bridge currency for transactions between different currencies.	In the United States, buying XRP is quite challenging.

How to Use Ripple or XRP?

- XRP can be used in the same way as any other digital currency, for transactions or as a possible investment.
- Other transactions, such as currency exchange, could be implemented using the Ripple network. For example,
- You can first exchange your US dollars for XRP on the Ripple network and then use those funds to purchase euros rather than dealing with the currency conversion directly through a bank or money-changing exchange.
- This can be a much quicker and less expensive choice than paying the high fees that banks and money transmission companies may demand.



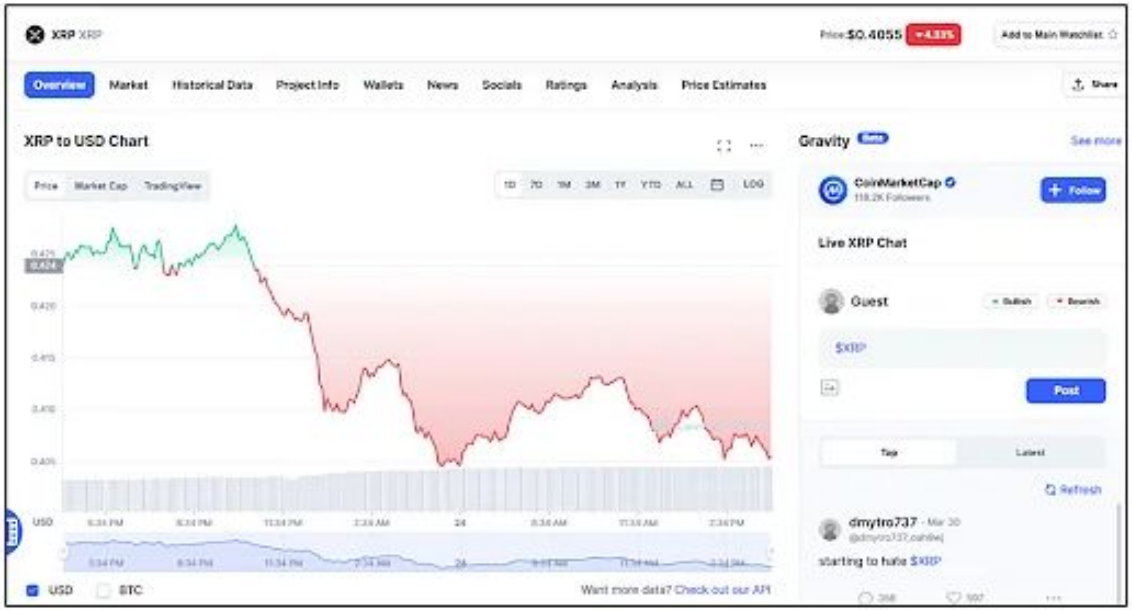
How to Buy Ripple?

To purchase Ripple's XRP cryptocurrency, create an account on an exchange that offers it for sale. Buy it next using one of the accepted payment methods on the exchange. Here are a few platforms to get XRP:

- KuCoin
- Uphold
- Coinmama

Is Ripple a Good Investment?

XRP is a risky investment that should not be undertaken lightly.



However, it would be wise to buy in XRP if you think Ripple will be a successful payment system. Be careful not to gamble with funds you cannot afford to lose.

References - Ripple

- https://ripple.com/
- https://ripple.com/insights/
- https://ripple.com/reports/Navigating-Crypto-Whitepaper-2022.pdf
- https://www.simplilearn.com/tutorials/blockchain-tutorial/what-is-ripple
- https://en.wikipedia.org/wiki/Ripple_(payment_protocol)
- https://ripple.com/developer-resources/

Agenda

- Corda
- **Ripple**
- Quorum
- other Emerging Blockchain Platforms
- Blockchain in DeFi: Case Study on any of the Blockchain Platforms
- Comparison Tables

What is Quorum?

Quorum

Quorum is a fork of the Go Ethereum (geth) client, which is the official GoLang implementation of the Ethereum protocol. Quorum is developed and maintained by J.P. Morgan.

Enterprise-ready, open-source blockchain platform, based on Ethereum:

- Designed for processing of private transactions within a **permissioned group of known participants**
- **Addresses specific challenges** to blockchain adoption within and beyond the financial services industry, e.g., **privacy, speed, throughput**

Quorum Key Features

- ✓ **Confidentiality** - details of transactions are private and never broadcast
- ✓ **Secure** – uses advanced encryption techniques, ensures only permissioned entities can access the network
- ✓ **Decentralized** – no dependency on a central service or party

Community – leverages world's largest pool of blockchain developers, supported by 150+ Enterprises through EEA

Proven – Ethereum has been in production since 2015, proving Quorum's underlying protocol in normal & stressed environments

Quorum Key Features

Quorum

PRIVACY

- Private Contracts & Transactions
- Zero Knowledge Layer

PERFORMANCE

- Very high TPS rate
- Blocks every ~ 50 ms

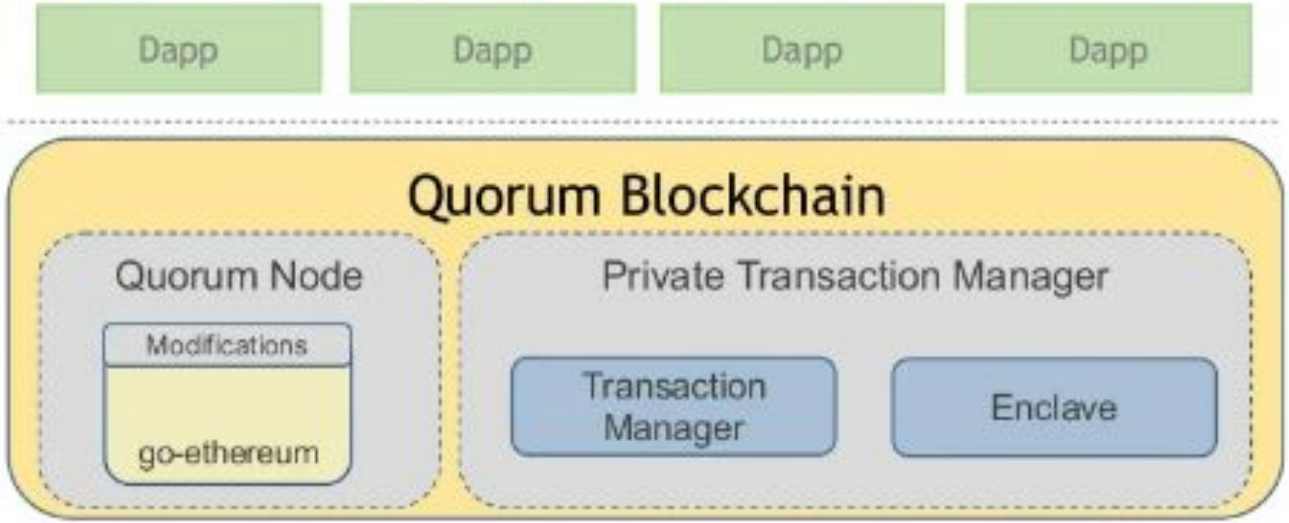
FINALITY

- No chain forking
- No transaction reversal

PERMISSIONING

- Known Peers Only
- Built into the protocol

Quorum High Level Design



Quorum works with standard Ethereum tools

Quorum Consensus Algorithm – Raft

- Well known consensus algorithm for distributed databases
- Useful for closed membership/ consortium settings
- At the start of the network, a leader is elected
- The leader proposes the blocks and other node validate the same
- A new leader is elected when the current leader goes down or term ends
- Leader election is completely **random**

Pros

- Faster block time (25 – 50 millisecs)
- Settlement finality

Cons

- Is not byzantine fault tolerance
- Requires interconnected network

Quorum Consensus Algorithm – Istanbul BFT

- In a network of N nodes can withstand F of Byzantine nodes where N = 3F + 1
- The algorithm has 4 phases – Propose, Pre-Prepare, Prepare, Commit
- The proposer multicasts the block proposal to the validators
- Validators agree on the block and broadcast their decision to others
- Each validator waits for 2F + 1 commits from different validators with the same result before inserting the block into blockchain

Pros

- Byzantine fault tolerant
- Settlement finality
- High throughput

Cons

- Complex and on-chain: blocks are always minted

Quorum In Depth: Privacy

What makes Quorum's privacy solution unique?

Quorum offers **decentralised privacy**:

- No dependency on a central node/service to ensure privacy
- No dependency on an external application to ensure privacy
- Single-chain architecture: chain contains both public & private transactions → guarantees privacy whilst ensuring better security
- Designed to meet regulatory requirements around in-country data & is compatible with next gen crypto primitives such as ZKP

Privacy implementation

Privacy achieved through Ethereum modifications & off-chain application: Tessera

- Ethereum modifications:
  - state trie split into public state trie & private state trie
  - 'v' value of private transactions set to 37 or 38
  - 'privateFor' added to transaction parameters in order to specify an array of recipients that will receive transaction's details
  - EVM prevented from executing private to public writes
- Tessera: A stateless application responsible for encryption/decryption of private transaction data and off-chain private messaging

How it works (also see https://github.com/jpmorganchase/quorum/wiki/Transaction-Processing)

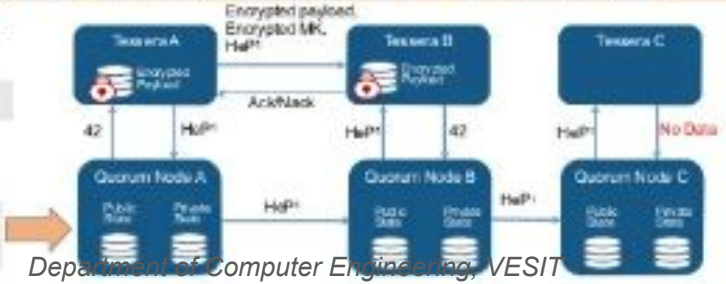
- Simple to use, same familiar Ethereum Transaction:

Public Transaction (no change from geth):

```
myContract.new(42, {from:$PartyA, data:
$CODE})
```

Private Transaction (from Party A):

```
myContract.new(42, {from:$PartyA,
data:$CODE, privateFor:[$PartyB]})
```



Quorum Roadmap 2019+

Privacy

- Organization-level privacy addressing
- Asset Transfer with Privacy (ZKP)
- Private Contract extensibility
- Private State consensus

Permissioning

- Authenticated RPC API access

Performance

- Transaction parallelization R&D
- EVM optimizations
- eWASM support

Resiliency

- Fault-tolerant Transaction Manager
- Transaction Manager clusters
- Data archiving

Scalability & Interoperability

- Inter-quorum asset transfers
- Public chain relay/bridge

Tooling

- One-click network deployments
- Database adapters for better querying
- Monitoring/Logging tools
- Identity management tools

General

- Ensure EEA Specification compliance
- Dedicated Developer Advocacy & Community team
- Hiring across engineering, product & community globally

What can I build with Quorum?

- Private blockchains are useful when you have an organization or problems that consist of
  - Geographically distributed members
  - Spotty trust between members
  - No reason or need for central control
- With above conditions met, some of the things that we can build with Quorum are
  - Asset or supply chain systems
  - Multi-party and organizational decision making via multi-signature contracts (distributed trust)
  - Real-time auditing, operational transparency, and data integrity uses

Who is using Quorum today?

Quorum

Just a small sampling. To see more, Google "Quorum Blockchain"!

- Central Bank Projects
  - Project Khokha – South Africa's Central Bank experiment for payments and settlements on blockchain
  - Project Ubin – Singapore Money Authority cash-on-ledger research platform
- Royalty Platform from Microsoft and EY
- Synaptic Health Alliance – platform for healthcare data exchange across healthcare sector
- JPMorgan Blockchain projects built on Quorum
  - Interbank Information Network – cross-bank blockchain for data exchange with 200+ banks across the globe
  - Project Dromaius – Debt issuance platform
  - JPM Coin – stable coin for value exchange

The easy way

Quorum

https://github.com/jpmorganchase/quorum-examples

- ✔ Official Quorum starting point
- ✔ An easy guest environment to get started with on any platform
  - Vagrant (**works on all machines**)
  - Dockerized setup
- ✔ Comes with examples for public and private smart contracts
  - configures an operational and fully configured 7 node cluster
  - has a real world example from finance industry: 5NodeRTGS

The advanced way

Quorum

For more advanced users with access to AWS and other environments

- ✔ Quorum Cloud – the official way of deploying Quorum networks on AWS using ECS Fargate, S3, and EC2 via automated Terraform configuration
  - https://github.com/jpmorganchase/quorum-cloud
- ✔ Quorum Maker – an open source tool made by Synechron Labs for guided Quorum network kick-start and management. This tool is provider agnostic
  - https://github.com/synechron-finlabs/quorum-maker
- ✔ Cakeshop – is an integrated development environment and SDK for Quorum
  - https://github.com/jpmorganchase/cakeshop

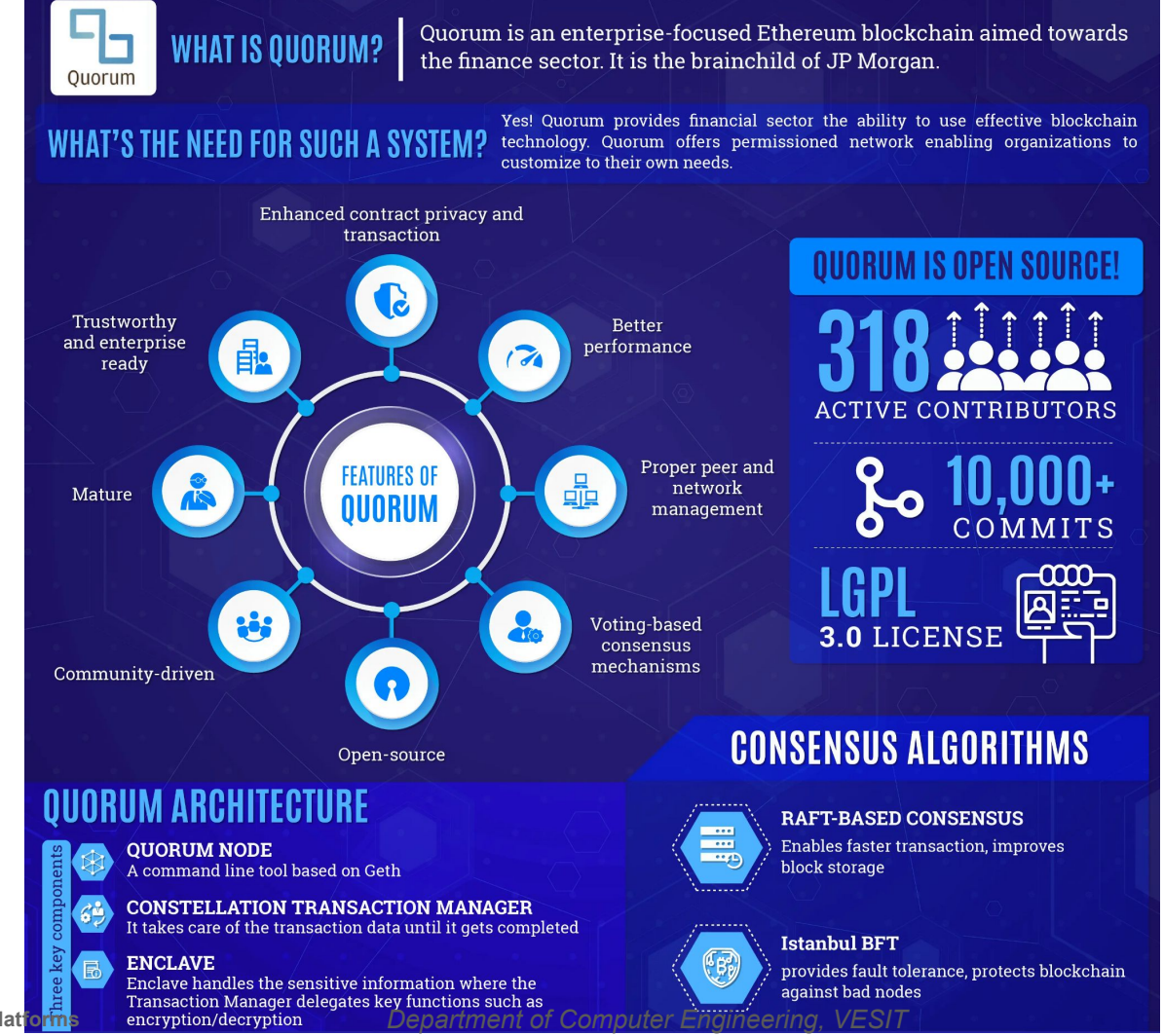
Reach out

Quorum

www.jpmorgan.com/quorum

www.github.com/jpmorganchase/quorum

slack http://bit.ly/quorum-slack


Quorum
WHAT IS QUORUM?
Quorum is an enterprise-focused Ethereum blockchain aimed towards the finance sector. It is the brainchild of JP Morgan.
WHAT'S THE NEED FOR SUCH A SYSTEM?
Yes! Quorum provides financial sector the ability to use effective blockchain technology. Quorum offers permissioned network enabling organizations to customize to their own needs.
Enhanced contract privacy and transaction
Better performance
Proper peer and network management
Voting-based consensus mechanisms
Open-source
Community-driven
Mature
Trustworthy and enterprise ready
FEATURES OF QUORUM
QUORUM IS OPEN SOURCE!
318
ACTIVE CONTRIBUTORS
10,000+
COMMITS
LGPL
3.0 LICENSE
QUORUM ARCHITECTURE
three key components
QUORUM NODE
A command line tool based on Geth
CONSTELLATION TRANSACTION MANAGER
It takes care of the transaction data until it gets completed
ENCLAVE
Enclave handles the sensitive information where the Transaction Manager delegates key functions such as encryption/decryption
CONSENSUS ALGORITHMS
RAFT-BASED CONSENSUS
Enables faster transaction, improves block storage
Istanbul BFT
provides fault tolerance, protects blockchain against bad nodes


ALASTRIA
A multi-sector Spanish consortium
IHS MARKIT
Loan trading(syndicated) on the blockchain
IHS Markit
ETHHOTELS
Part of PoC multi-blockchain network
ETH HOTELS
VAKT
Digital ecosystem for physical post-trade Processing
VAKT
BLOCKONE IQ -QUORUM
Data provider for Quorum blockchain
B1IQ
PROJECTS / NETWORKS BASED ON QUORUM
KOMGO
Financial network integration with other users and networks
komgo
FogChain
FOGCHAIN
10T based location service and record management
Kimlic
iin
KIMLIC
KYC on the blockchain
IIN
Interbank Information Network(IIN) payments platform

Table 1: Hyperledger Fabric Vs. Corda Vs. Quorum

Show 6 entries Search:

Feature/Metrics	Hyperledger Fabric	Corda	Quorum
Consensus algorithm	Kafka	RBFT	Pluggable
	RBFT	Pluggable	Istanbul BFT
	Sumeragi	Raft	Raft Consensus
	PoET		
	Pluggable		
Throughput	>2000 tps	170 tps	A few 100s
Tokens	FabToken(not yet released)	-	Ether
Zero Knowledge Proof	Yes	No	Yes
Smart contract language	Java, Golang, NodeJS	Java, Kotlin	Solidity

ghlights

Built on Ethereum

- First mover advantage. In production since July, 2015.
- 50,000+ unit tests, Security Audits, Bounty Program
- Largest Ecosystem of Developers, Tools, DApp's
- Public Ethereum blockchain protects over $1B+ Ether[1]

Simple Privacy Design

- Supports both private and public transactions and smart contracts

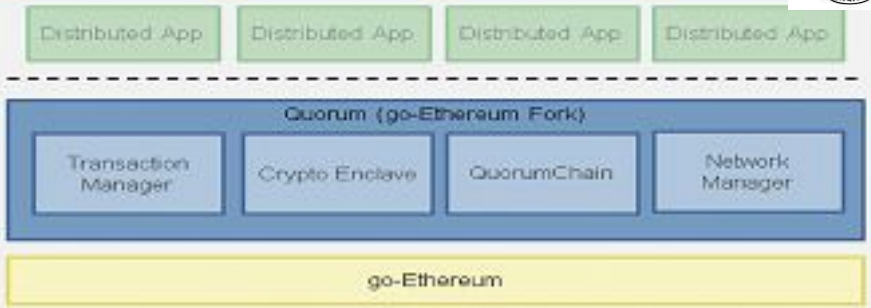
Single Blockchain Architecture

- All public and private smart contracts and state derived from a single, common, complete blockchain of transactions validated by every node in the network
- Private smart contract state validated by parties to contract only
- Best of both worlds… every node validating the list of transactions while only exposing details of private transactions and contracts to relevant parties

High Performance

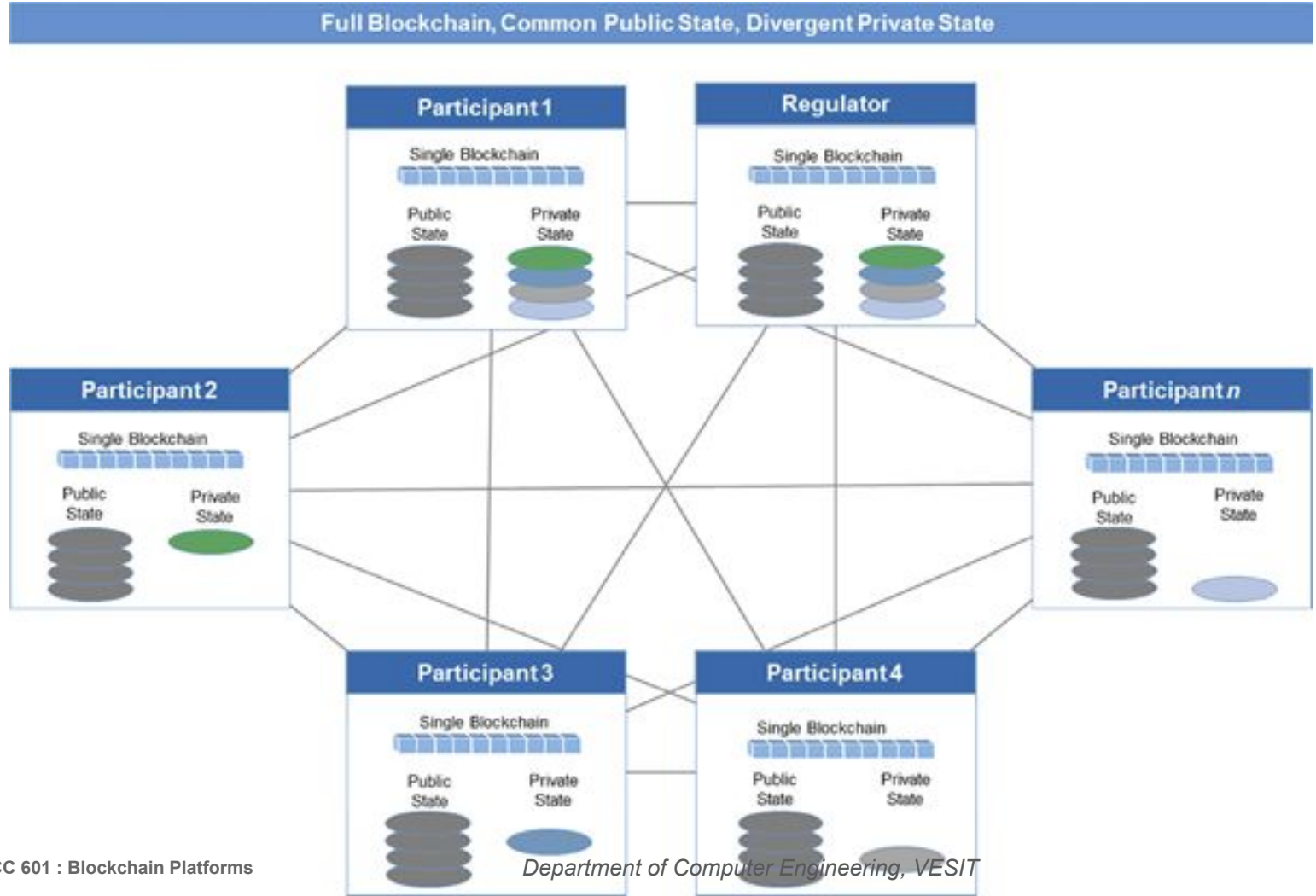
- Able to process **dozens to hundreds of transactions per second**, depending on system configuration; enough to support institutional volumes

Architecture

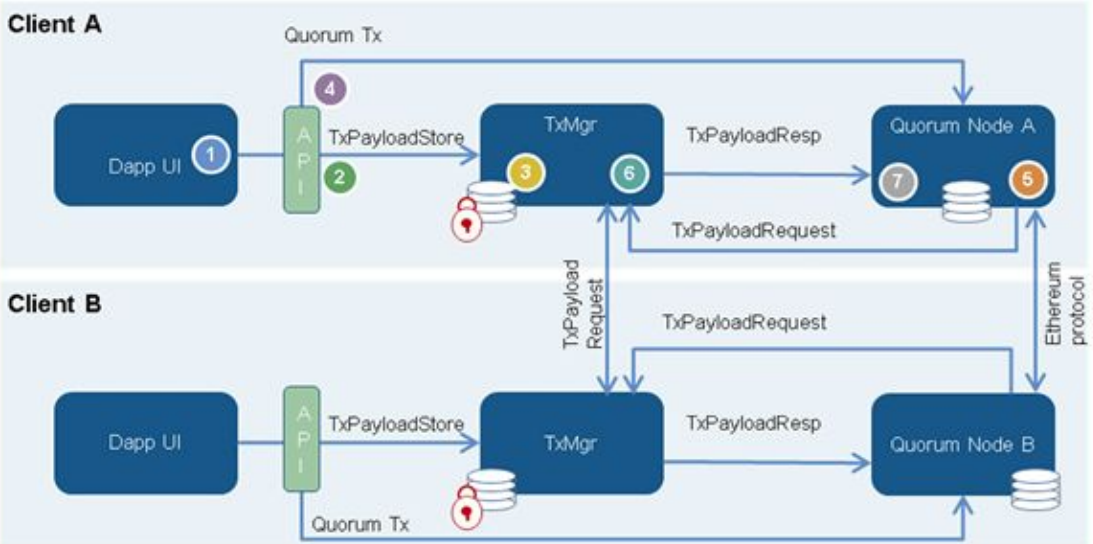


Components

- **Transaction Manager** – allows access to encrypted transaction data for private transactions, manages local data store and communication with other Transaction Managers
- **Crypto Enclave** – responsible for private key management and encryption and decryption of private transaction data
- **QuorumChain** – voting-based, BFT-hardened consensus mechanism that utilises core Ethereum features to verify and propagate votes through the network
- **Network Manager** – controls access to the network, enabling a permissioned network to be created


Full Blockchain, Common Public State, Divergent Private State
Participant 1
Single Blockchain
Public State
Private State
Regulator
Single Blockchain
Public State
Private State
Participant 2
Single Blockchain
Public State
Private State
Participant n
Single Blockchain
Public State
Private State
Participant 3
Single Blockchain
Public State
Private State
Participant 4
Single Blockchain
Public State
Private State

Simple Privacy Design



1. Dapp sends transaction to Quorum Node, specifying recipient and transaction payload
2. **Prepare Tx Payload Record** by generating a symmetric key, encrypt the payload with symmetric key, hash the encrypted payload, encrypt the symmetric key with the public keys of the parties to the Tx, then send to the TxMgr for storage.
3. **TxMgr** validates the sending signature and stores the TxPayload message
4. **Tx sent** to the Quorum node containing only the hash of the encrypted payload generated in step 2.
5. **Quorum Node** receives a new block for validation containing the private Tx. It requests the payload data from the TxMgr (passing its Pubkey, TxHash, Sig)
6. **TxMgr** validates the signature, looks up the TxHash and if the requester is party to the Tx, return the encrypted payload and encrypted Symmetric key.
7. **Quorum Node** decrypts the symmetric key, decrypts the Tx Payload and sends to the EVM for contract code execution.

TxPayload includes:
- Encrypted Tx payload
- Hash of encrypted Tx payload (TxHash)
- Party 1 Public Key encrypted Symmetric Key
- Party 2 Public Key encrypted Symmetric Key
- Party *n* Public Key encrypted Symmetric Key

ghlights

Built on Ethereum

- First mover advantage. In production since July, 2015.
- 50,000+ unit tests, Security Audits, Bounty Program
- Largest Ecosystem of Developers, Tools, DApp's
- Public Ethereum blockchain protects over $1B+ Ether[1]

Simple Privacy Design

- Supports both private and public transactions and smart contracts

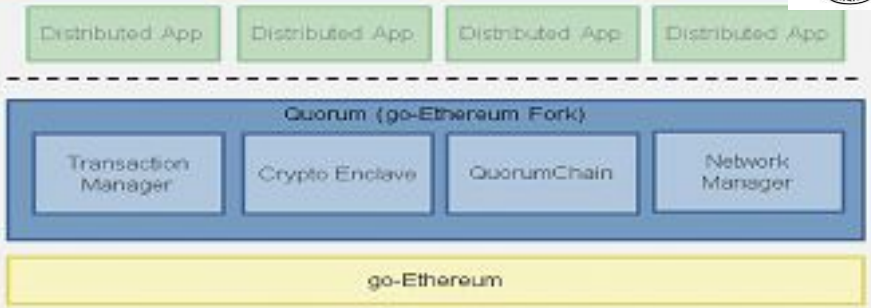
Single Blockchain Architecture

- All public and private smart contracts and state derived from a single, common, complete blockchain of transactions validated by every node in the network
- Private smart contract state validated by parties to contract only
- Best of both worlds… every node validating the list of transactions while only exposing details of private transactions and contracts to relevant parties

High Performance

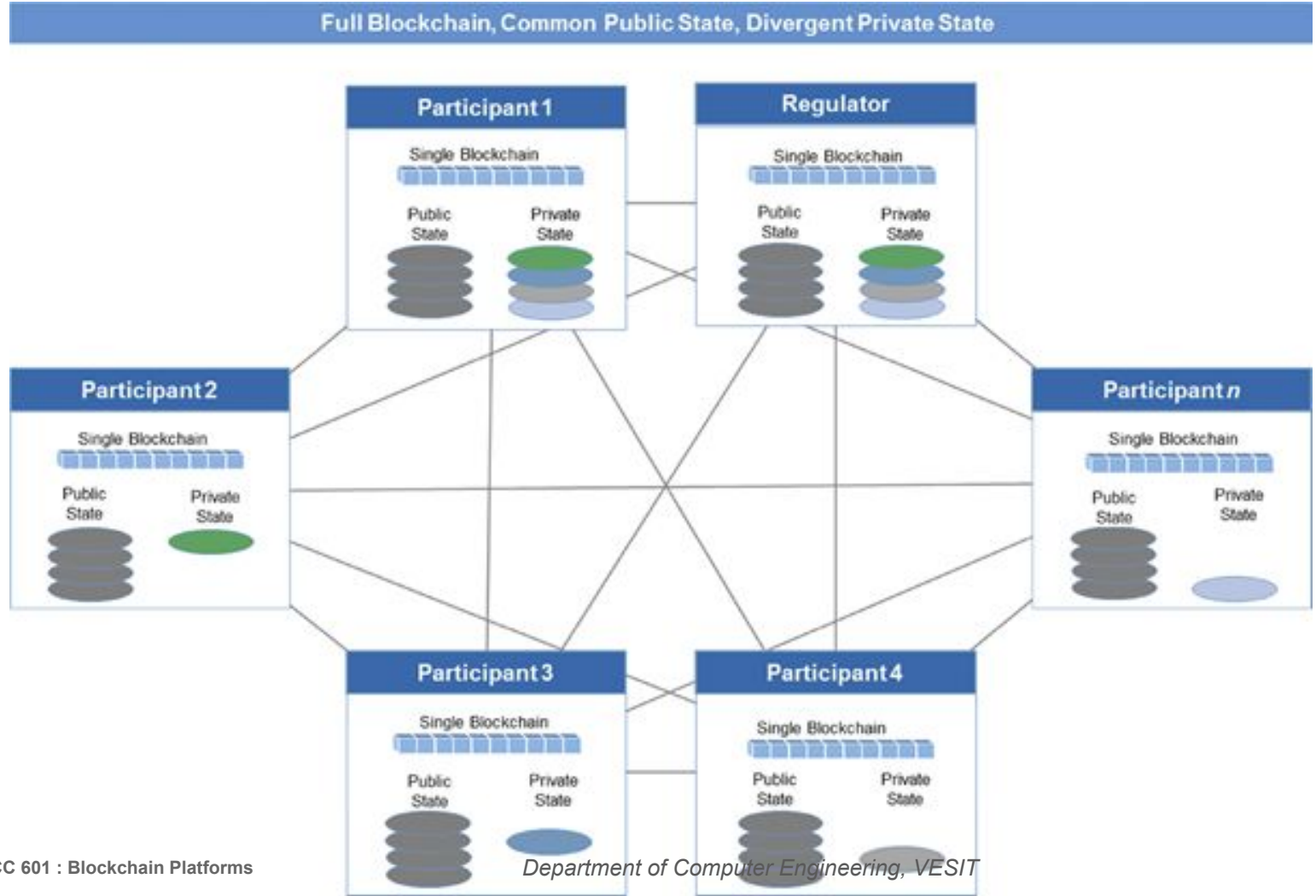
- Able to process **dozens to hundreds of transactions per second**, depending on system configuration; enough to support institutional volumes

Architecture

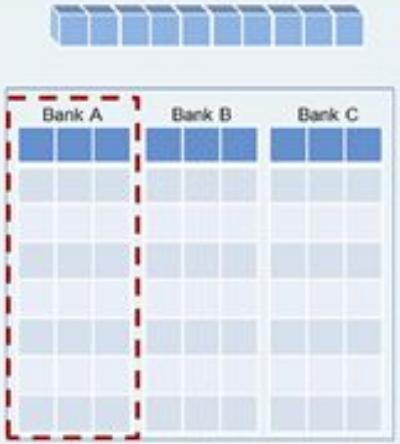


Components

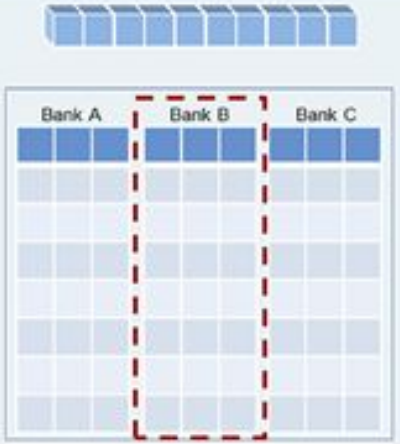
- **Transaction Manager** – allows access to encrypted transaction data for private transactions, manages local data store and communication with other Transaction Managers
- **Crypto Enclave** – responsible for private key management and encryption and decryption of private transaction data
- **QuorumChain** – voting-based, BFT-hardened consensus mechanism that utilises core Ethereum features to verify and propagate votes through the network
- **Network Manager** – controls access to the network, enabling a permissioned network to be created


Full Blockchain, Common Public State, Divergent Private State
Participant 1
Single Blockchain
Public State
Private State
Regulator
Single Blockchain
Public State
Private State
Participant 2
Single Blockchain
Public State
Private State
Participant n
Single Blockchain
Public State
Private State
Participant 3
Single Blockchain
Public State
Private State
Participant 4
Single Blockchain
Public State
Private State

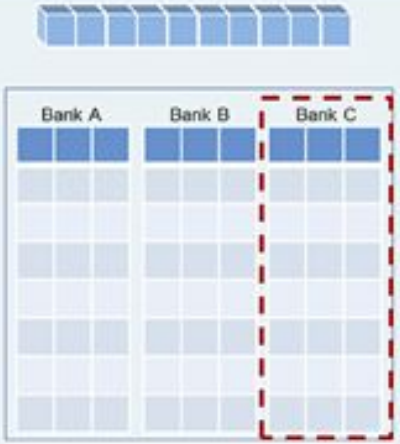
Bank A node



Bank B node



Bank C node



One contract per bank architecture

- All transactions sent to individual Bank smart contracts, atomic payment (debit/credit) still intact. Each node has all bank contracts.
- Advantage of this design is for contract state validation with regulator.
- Quorum "Private Transactions" enforce privacy requirements
- Only the local bank contract is 100% complete, other bank ledgers represent (payment history) from the local nodes perspective
- Regulator will/can have a complete view by being party to every transaction

Regulator

References - Quorum

- https://www.kaleido.io/blockchain-platform/quorum
- https://101blockchains.com/quorum-blockchain-tutorial/
- https://101blockchains.com/quorum-blockchain-use-cases/
- https://consensys.net/docs/goquorum/en/latest/concepts/architecture/
- https://abhibvp003.medium.com/quorum-a52631b13018

Agenda

- Corda
- Ripple
- Quorum
- **Other Emerging Blockchain Platforms**
- Blockchain in DeFi: Case Study on any of the Blockchain Platforms
- Comparison Tables

Binance

- Binance is an online exchange where users can trade cryptocurrencies.
- Crypto wallet and exchange services.
- Safe blockchain
- Higher Value for money score compared to Blockchain.

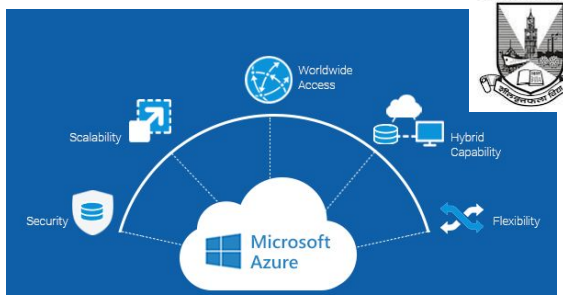
Cardano

- PoS
- Internal cryptocurrency ADA
- 100% decentralization in 2015 and has 1500 validator pools
- Faster and cheaper transactions than Ethereum
- Also supports NFTs
- The Cardano protocol called Ouroboros has mathematical proof of the persistence.
- Also known as academic blockchain because based on lots of research and tests.
- So trustworthy but slow development

IBM

- It is connected to the free IBM cloud Kubernetes clust
- It is an open-source and community-based platform.
- It offers a permission network.
- This platform supports Go and Java programming languages.
- Open-source protocol is built to run in any computing infrastructure, on-premises or in the cloud.
- It provides a 30 day free trial for users.

MICROSOFT AZURE



- Blockstream has a cluster of blockchain-related products.
- But its main product is Liquid Network, a sidechain network for faster and private transactions.
- Blockstream also has a hot wallet (Aqua) and a hardware wallet (Jade) for liquid and bitcoin networks.

BLOCKSTREAM

They propose a 3-year cloud mining for bitcoin networks as well.

Blockstream assures its users to deploy cold storage (like hardware wallet) to prevent any hacking attempt by cybercriminals.

AVALANCHE

- Native currency is AVAX
- It's one of the rival of ethereum which beats in scalability, speed
- Its transaction speed is 4500 transactions/ sec
- Claims thats its safe against 51% attacks

BIGCHAINDB

- Database with Blockchain characteristics
- Compatible with python, c++, golang
- Its BF tolerant
- Check transactions, assets, etc

Agenda

- Corda
- Ripple
- Quorum
- Other Emerging Blockchain Platforms
- **Blockchain in DeFi: Case Study on any of the Blockchain Platforms**
- Comparison Tables

Case Study of Trade Finance in R3 Corda Blockchain

- Enterprise DeFi is a term used to describe major global corporations that incorporate Decentralized Finance (DeFi) technology into their services, products, and operations.
- It enables institutional investors to easily access new lending markets based on blockchain-powered financial infrastructure, which was previously impossible.
- The financial sector is trying to take lead in blockchain adoption by investing heavily in the technology.
- We need advanced solutions that incorporate existing legacy technologies, workflows, and regulatory systems.
- There are three types of networks : public, private, and federated.
- A public blockchain network can be joined and used by anyone. Everyone has access to the ledger and can be involved in the consensus mechanism.

Case Study of Trade Finance in R3 Corda Blockchain

- In a private blockchain, only one organization has authority. In other words, the authorities decide who can participate in and access the network. Consider it a "centralized-decentralized" network.
- When you use a federated blockchain, you will gain access to a decentralized private environment..Federated blockchain is central to private blockchain. It is used in financial services, insurance payments, multi - party aggression, supply chains, and many other industries.
- Over the last three years, enormous efforts have been made to establish federated networks, with over a dozen projects, the majority of which use R3 Corda or Hyperledger.
- This is appropriate for businesses who want to use blockchain without going public. These authority nodes have already been chosen from among all the organizations connected by the network. A block is validated by this selected group of nodes. Furthermore, this suggests that the selected nodes have access to information and areas that are restricted.

Case Study of Trade Finance in R3 Corda Blockchain

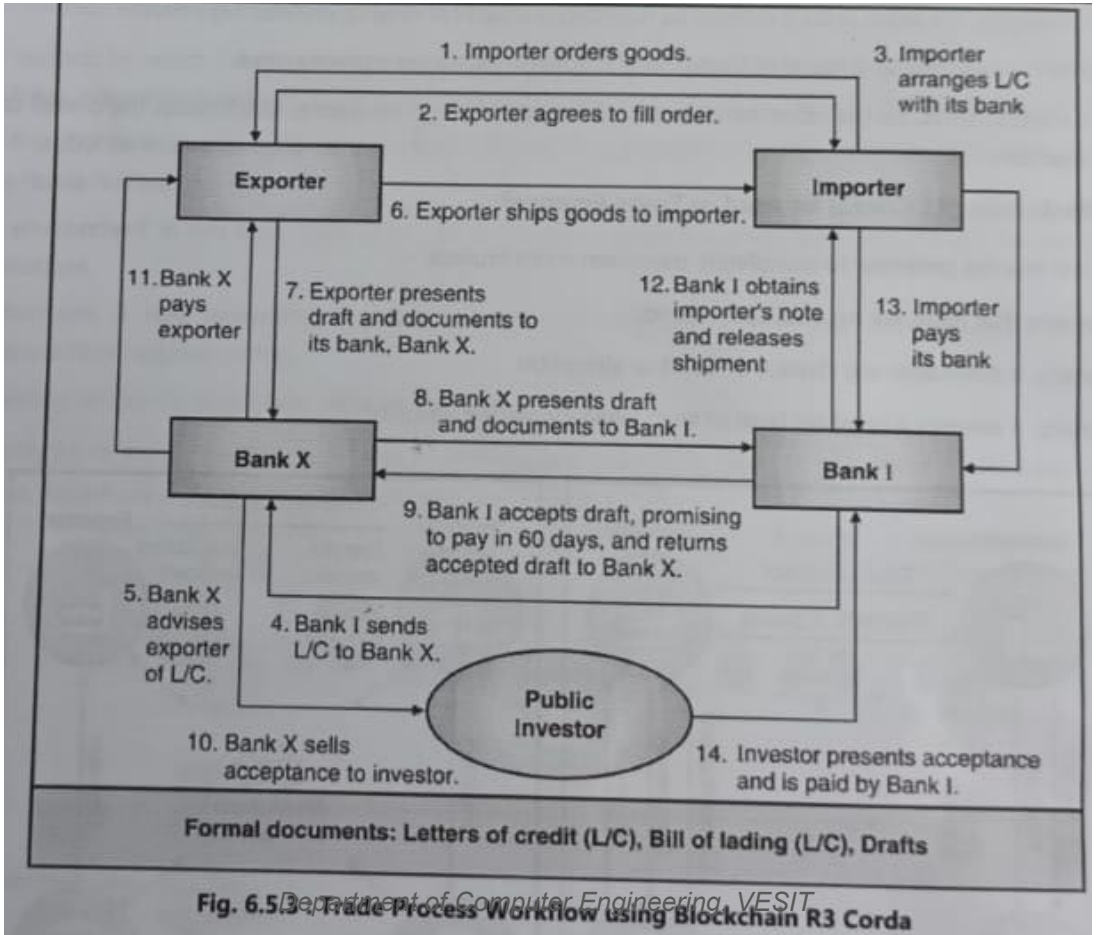


Fig. 6.5.3 : Trade Process Workflow using Blockchain R3 Corda

Case Study of Trade Finance in R3 Corda Blockchain

The Advantages of Blockchain for Trade Finance

- The blockchain trade finance platform's documents can be checked for authenticity at any time! It helps to save time for all parties involved. This also implies that no fraudulent activities take place.
- Transparency is also provided by blockchain in trade finance.
- There are no middlemen, which improves the overall system by reducing fraud. This increases the banks' ability to do trade financing without risk or dispute.
- Blockchain in trade finance can prevent double spending.
- Smart contracts could indeed help to automate and eliminate time spent on paperwork. Before smart contracts can be executed on the network, they must be finalized. Before the smart contract agreement is drafted, both partners must meet and finalize the details.
- Furthermore, the smart contract is only executed when a predefined condition or set of rules is met.
- Blockchain enables buyers and sellers to have proof of ownership, ensuring transparency and tracking the location of the shipment.
- Regulations could be managed in a single location. KYC/AML solutions are included. Additionally, it facilitates trade between nations by allowing for the management of regulatory differences between them.
- Besides this basic use case, blockchain is used in tons of trade finance use cases like, international trade-Letter of Credit, Maritime Trade, etc.

Case Study of Trade Finance in R3 Corda Blockchain

A Case Study of the Trade Finance System using R3 Corda

- The method by which financial firms grant credit to ensure the exchange of goods is known as "trade financing", and it has existed for hundreds of years with little change as the volume of international trade has increased.
- The financial services and equipment used by companies to promote domestic and international trade are referred to as "trade finance".
- The environment is not ideal right now. For instance, at present, transactions are carried out using third-party applications.
- Furthermore, it may necessitate a significant amount of documentation, including multiple copies and the involvement of multiple parties.
- Ultimately, simple tasks can take weeks to complete.
- The players in the environment of trade finance include importers, exporters, financial institutions, trade finance entities, export credit companies, and insurers.

Case Study of Trade Finance in R3 Corda Blockchain

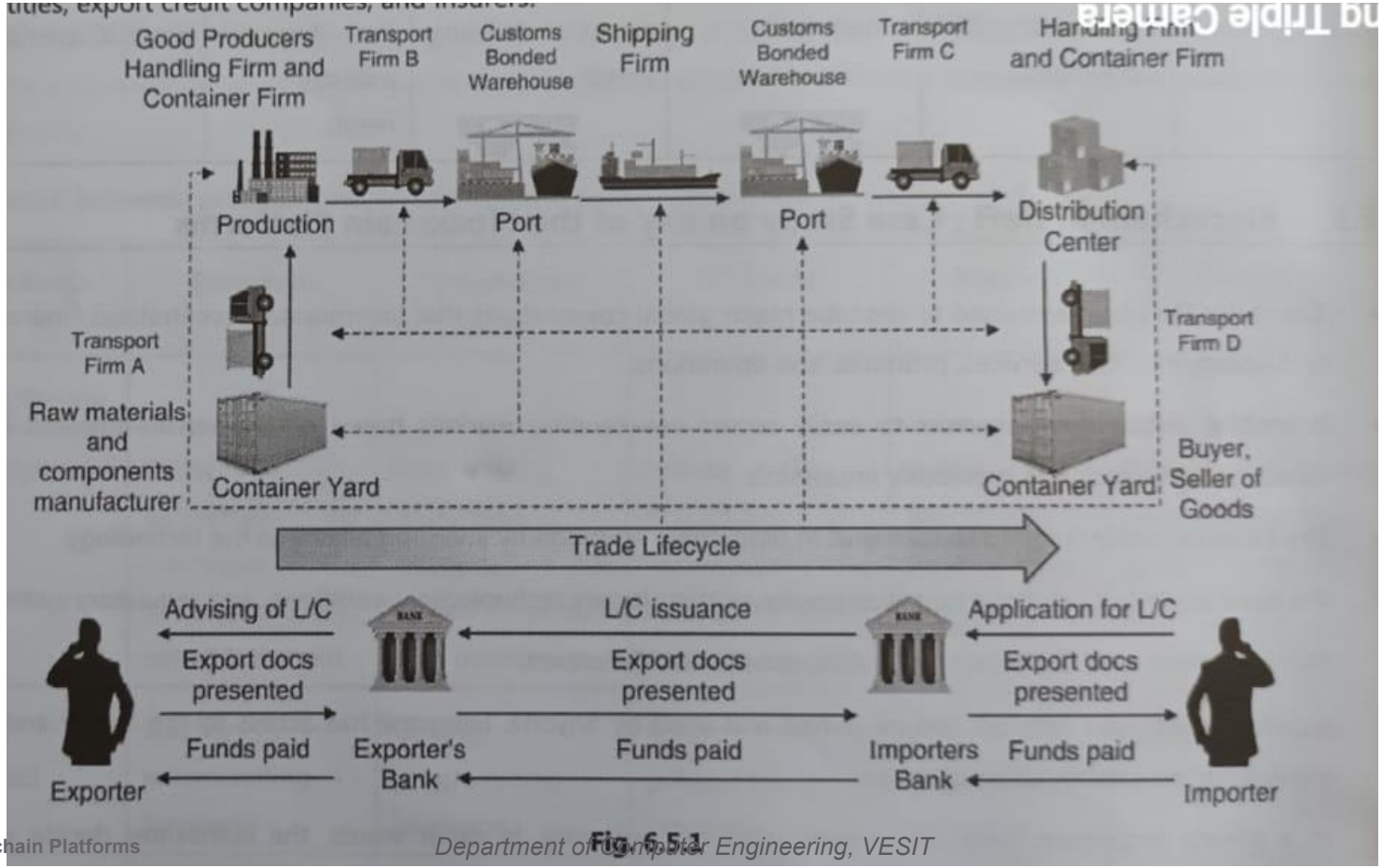


Fig. 6.5.1

Case Study of Trade Finance in R3 Corda Blockchain

- Current trade finance issues are :
  - Every good in the supply chain must be checked for quality on a regular basis.
  - Throughout the supply chain journey, each good must interact with a variety of players and people.
  - A product can take up to 30-35 days to travel from farm to retailer. Furthermore, the documents may take up to ten days to complete.
  - This leads to incorrect party tracking and fraud, such as fake deliveries.
  - Costly throughout, especially when it comes to writing the letter of credit, the process involves banks and clients. Additionally, disputes are managed improperly and take a long time to settle.
  - They face a significant regulatory burden because they must manage things, including fraud prevention, KYC protocols, geopolitical risks, etc.
  - Furthermore, the seller finds it difficult to maintain product conditions, introducing product risks.
  - Currency risks arise as a result of fluctuations in foreign exchange currency rates.
  - The transport risk, on the other hand, arises as a result of cargo insurance, which raises the overall cost of th transaction.

Case Study of Trade Finance in R3 Corda Blockchain

How Can Blockchain (R3 Corda) be used in Trade Finance?

- Blockchain has the potential to completely transform trade finance.
- It can ensure that there are no duplicate records.
- Additionally, it eliminates any chance of fraud or alteration.
- Additionally, it ensures a constant level of trust while increasing visibility.

Case Study of Trade Finance in R3 Corda Blockchain

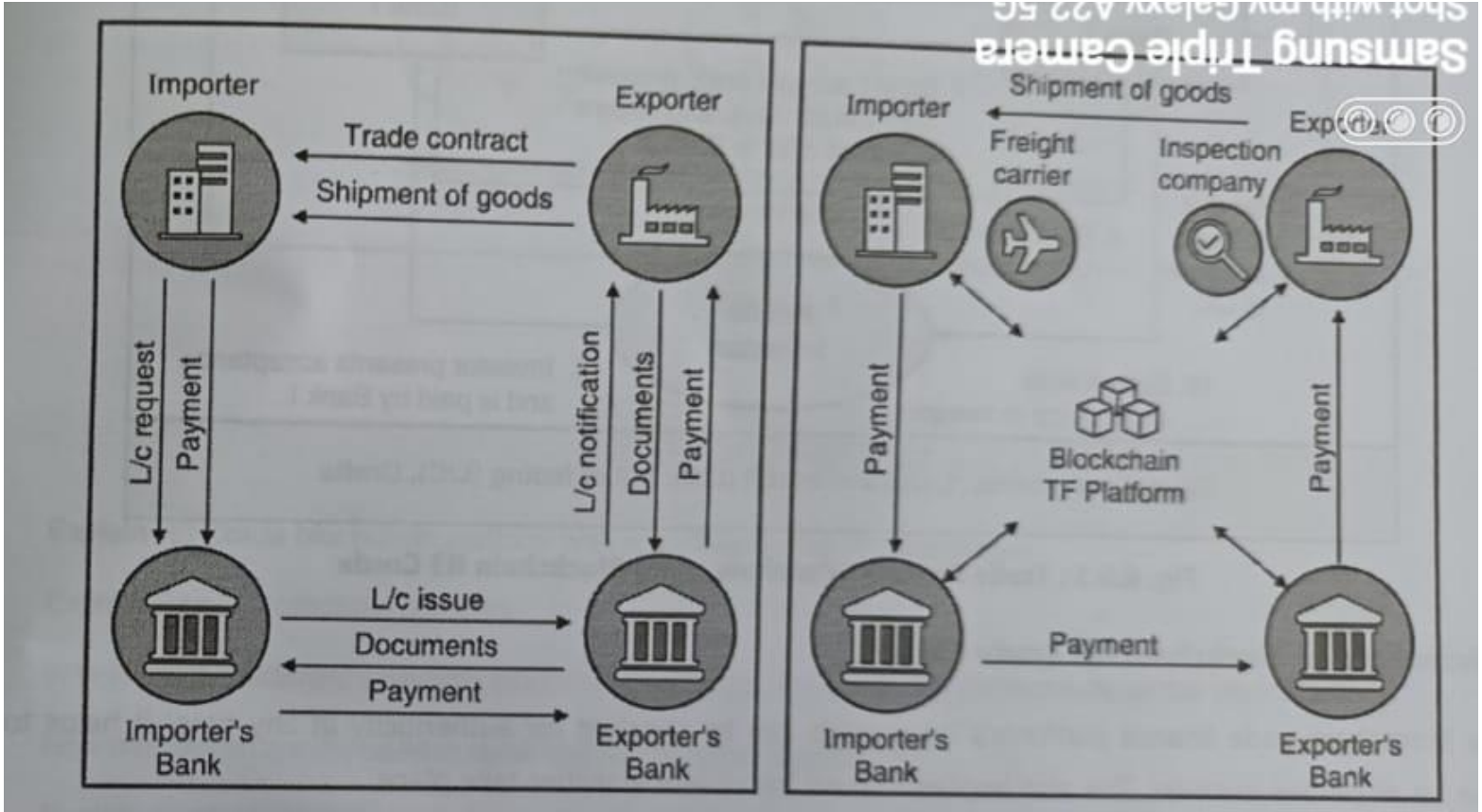


Fig. 6.5.2 : Trade Finance System Players and Workflow

Case Study of Trade Finance in R3 Corda Blockchain

How?

- A trustworthy trading partner can be found by buyers and sellers using blockchain.
- The terms are then agreed upon by both parties, and the smart contracts are initiated. The order is then initiated after the smart contract is created.
- In the event that the smart contract requirements are satisfied, the buyer bank assumes control at the following location and will automatically settle the amount.
- To expedite the settlement process, the seller could request financing from his bank.
- The seller is now shipping the goods. Both parties are keeping an eye on the package.
- The buyer confirms the trade, indicating that the condition on the smart contract has been met.
- Finally, the seller is compensated.
- The procedure is straightforward, and as you can see, the majority of the steps are automated in order to maintain transparency and trust.

Case Study of Blockchain in Trade Finance

- R3 Corda real-world use-cases are TMX, Payments Canada, Maersk, NatWest, CrowellMoring, and Swift.

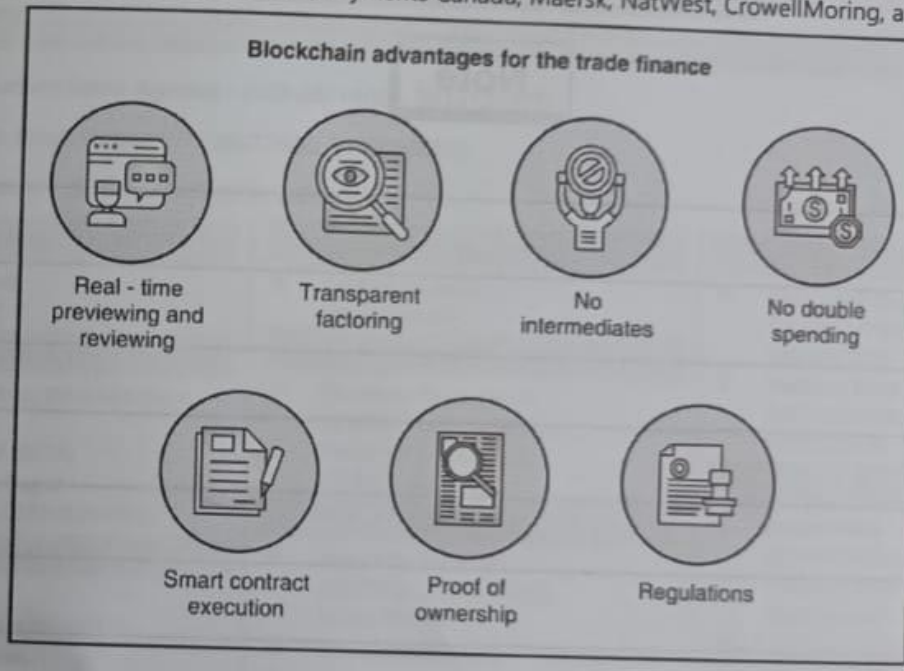
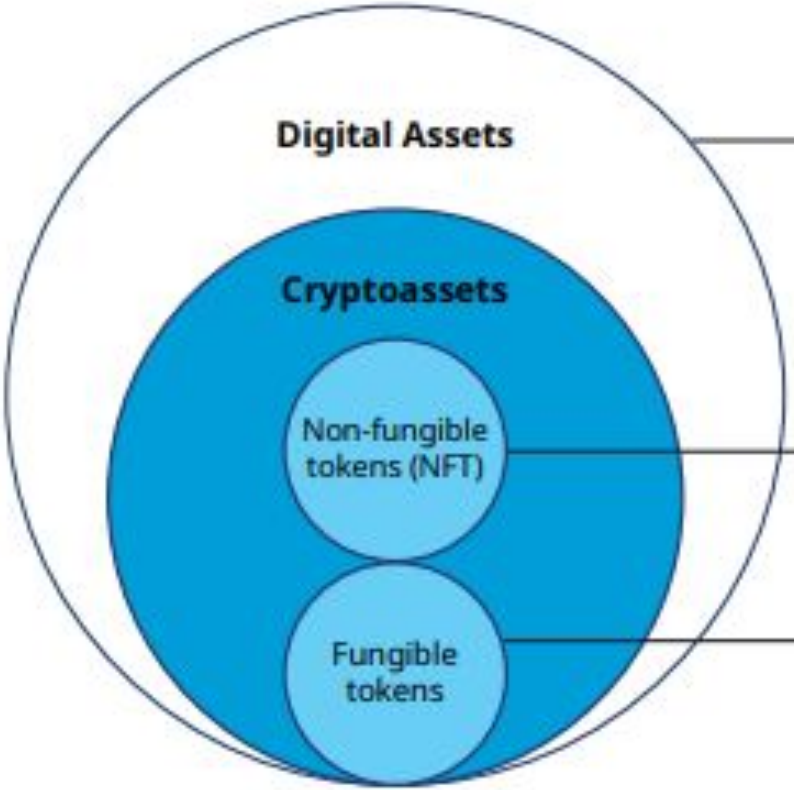


Fig. 6.5.4

Agenda

- Corda
- Ripple
- Quorum
- Other Emerging Blockchain Platforms
- Blockchain in DeFi: Case Study on any of the Blockchain Platforms
- **Comparison Tables**

Distinction between crypto and digital Assets



Assets based on blockchains (permissioned, permissionless) or designed to have some of their features

Digital asset examples:

- CBDCs (may or may not be blockchain-based)
- Tokenized securities

Assets built on public blockchains which include fungible and non-fungible tokens

NFT examples:

- Digital art
- Tickets

Fungible token examples:

- Cryptocurrencies (popular cryptocurrencies such as BTC, ETH, and more)
- Stablecoins (USDT, USDC)
- Governance tokens (DeFi protocols governance tokens)

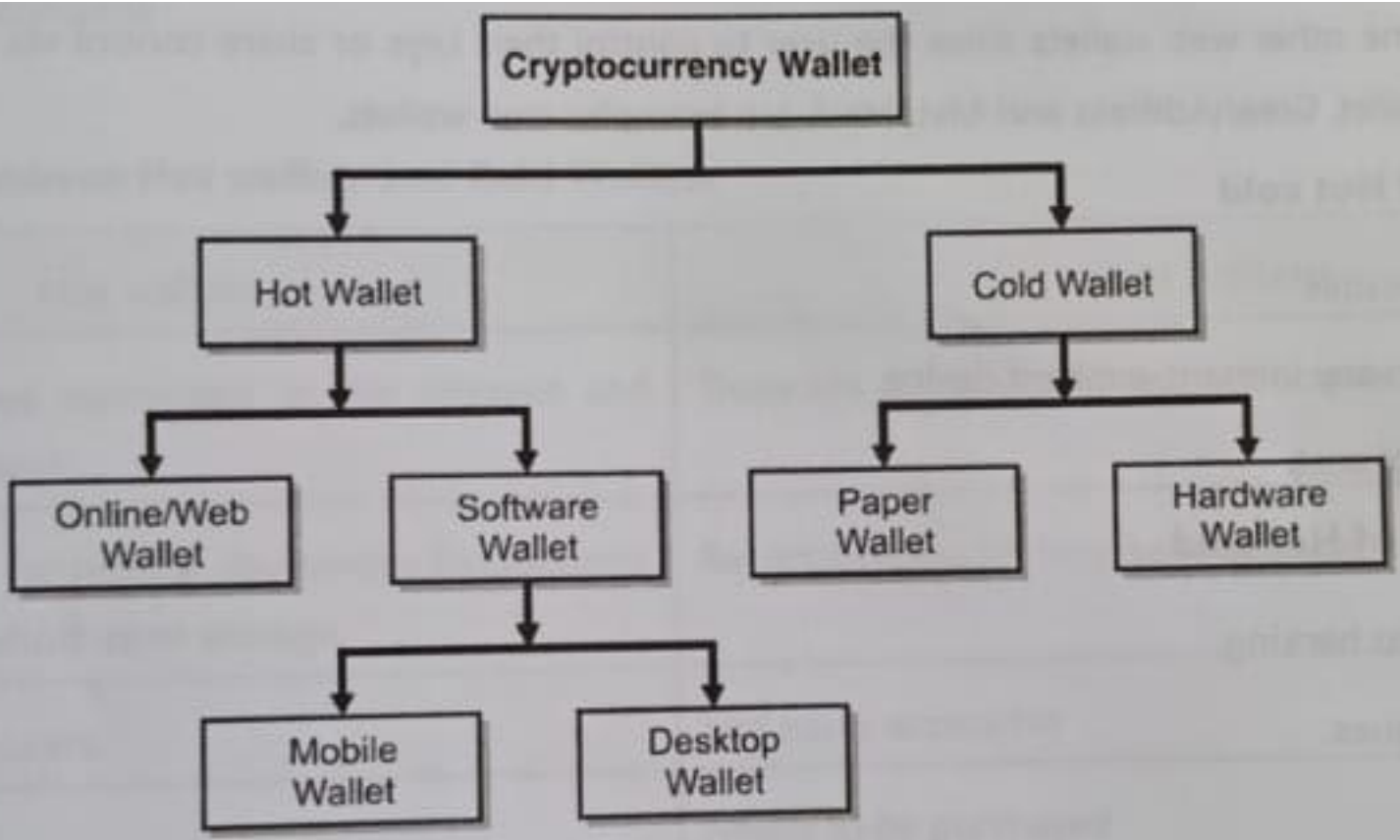


Fig. 2.4.1 : Types of wallets

Hot wallets	Cold Wallets
This wallet is always connected to the internet and cryptocurrency network.	These are offline wallets and not connected to internet
These better to use for regular day-to-day transactions. Recommended for short-term storage.	Recommended for long-term storage.
Easily accessible by users.	Not easily accessible
Usage is free	Needs to be purchased
Susceptible to malware or cyber attacks	Stealing the currency from cold wallet usually would need physical control of or access to the cold wallet. No risk of hack or malware.
Software wallets and online wallets are the examples.	Hardware wallets and paper wallets are the examples.

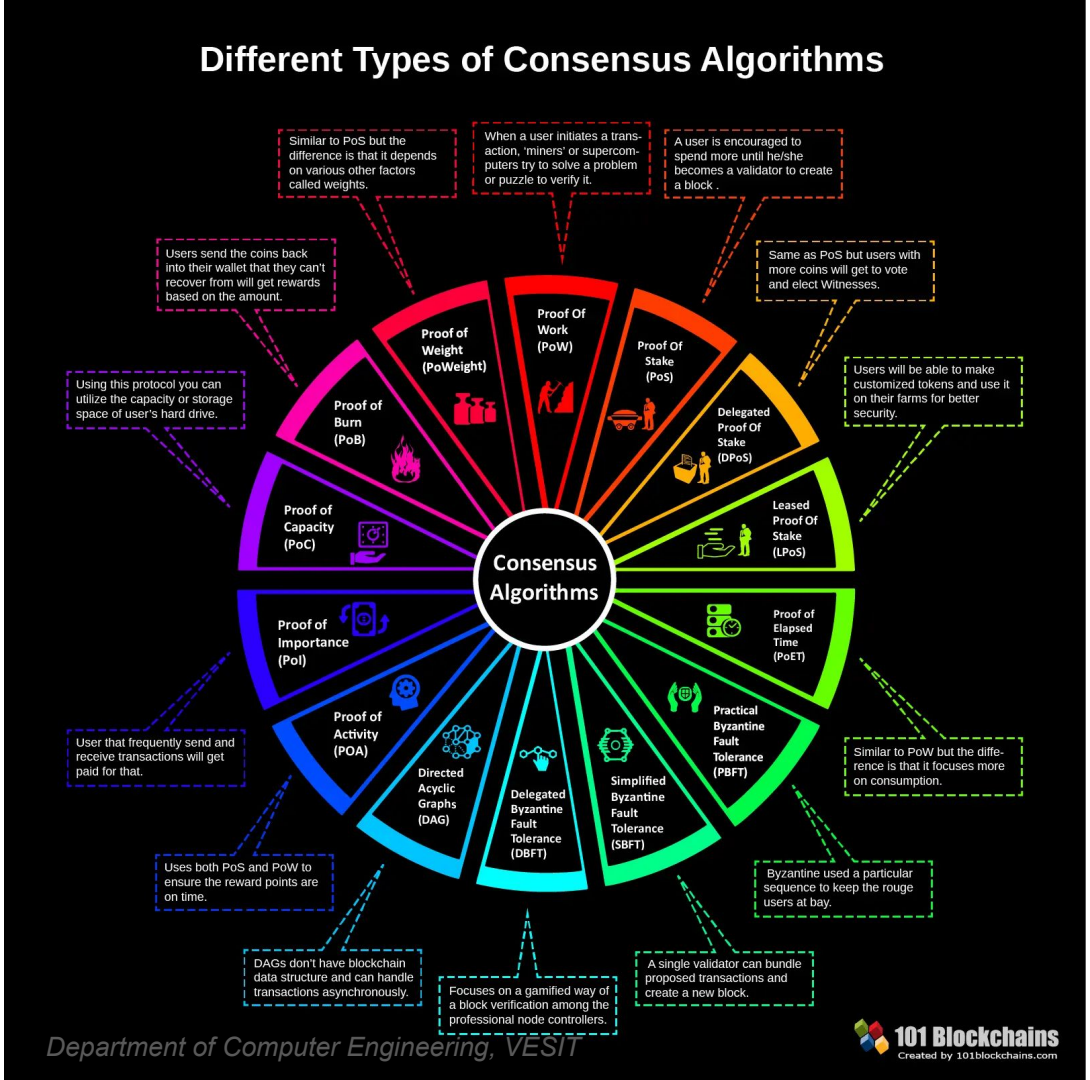
Comparison between Altcoin, Utility Tokens and Security tokens

Altcoin	Utility Tokens	Security tokens
These currencies are created using open-source software of an existing currency or as native currency of an original blockchain and protocol.	In order to create the tokens, standard templates from blockchain platform are used. They are created via smart contracts.	In order to create the tokens, standard templates from blockchain platform are used. They are created via smart contracts.
Altcoins are used for payment purpose.	Utility tokens are created for funding blockchain projects or companies. These tokens are given out during crowd sales as a project executes an Initial Coin Offering (ICO).	Utility tokens are created for funding blockchain projects or companies. Security tokens are offered during Security Token Offering (STO).
It is currency or cash	It represents right to use service or product in future	A security token signifies some kind of ownership, most commonly a share of the company that issues the token.
Value depends on market supply and demand	Used for utility. Users with utility tokens receive exclusive benefits like discounted transaction fees and early access to products and services on the platform. Users also gets voting rights etc.	Token holders gets part ownership. Value is directly linked to the valuation of company.
Security depends on strength of blockchain protocols.	Susceptible to ICO scams and frauds	Regulations prevents from scam and frauds.
These are not regulated or controlled by any authorities.	Typically unregulated.	Strictly Regulated by financial market authority.

Externally Owned Account	Contract Account
Owned by some external entity (person, corporation etc)	Owned by contract
This account does not have a code.	A code is assigned to each contract account.
A transaction can be initiated by an externally owned account with another externally owned account or a contract account.	But contract account cannot initiate transactions.
Externally owned accounts have a private key associated with them	Contract accounts do not have a private key
It is free to open an externally owned account.	It costs money to set up a contract account.
The only type of transaction that may occur between two EOAs is the transfer of ether.	Any type of transaction with a contract account is permitted, including deploying or invoking the functionality of other contracts.

PoW	PoS	PoET	PBFT	DBT	HoneyBadger-BFT	Tendermint	IoTA
Fintech	Fintech	Lack of consensus finality	Vulnerable to faulty nodes > (n-1)/3 n = total nodes	Vulnerable to faulty nodes > (n-1)/3 n = total nodes	Fintech	Fintech	Lack of consensus finality
High energy & computation cost	Lack of consensus finality	Prone to forks	High communication complexity	Vulnerable to DoS Attack	Vulnerable to Sybil Attack	Vulnerable to faulty nodes > (n-1)/3 n = total nodes	Prone to forks
Lack of consensus finality	Prone to forks	Trust is placed in the enclave that allocates wait time	Vulnerable to DoS Attack	Poor Scalability w.r.t No of Validating Nodes	Vulnerable to faulty nodes > (n-1)/3 n = total nodes	Poor Scalability w.r.t No of Validating Nodes	Prevents double spending
Prone to forks	Latency in TX confirmation due to forks	Require special hardware	Poor Scalability w.r.t No of Validating Nodes	Low communication complexity	Poor Scalability w.r.t No of Validating Nodes	Vulnerable to DoS Attack	Mitigates Sybil Attack
Mining require ASICs	51% Attack	Distributed Ledger	Distributed Ledger	Distributed Ledger	High computation cost, compared to other BFT protocols	Consensus finality & No forks	Low energy & computation cost
High latency in TX confirmation	Malicious Collusion of Rich Stakeholders	Prevents double spending	Consensus finality & No forks	Consensus finality & No forks	Consensus finality & No forks	Fast TX confirmation	Low Latency (No Fee, Parallelized Consensus)
51% Attack	Prevents double spending	Low energy cost	Fast TX confirmation	Fast TX confirmation	Fast TX confirmation	Low communication complexity	No voting required
Prevents Double Spending	Mitigates Sybil Attack	Low computation cost	High Throughput	High Throughput	Low communication complexity	Prevents double spending	Avoids Quantum Computing Attacks
Mitigates Sybil Attack	Nodes are not trusted	Nodes are known	Low energy & computation cost	Low energy & computation cost	Avoids DoS attack based on timing assumption	Low energy & computation cost	Low communication complexity
Nodes are not trusted	Low energy & computation cost		Prevents double spending	Prevents double spending	Prevents double spending	Punishment for validating nodes	Suitable for Asynchronous Networks
			Nodes are known	Nodes are known	Nodes are known		Addresses scalability issue concerning network size and TX throughput

Courtesy : ResearchGate



Courtesy : [101 Blockchains]

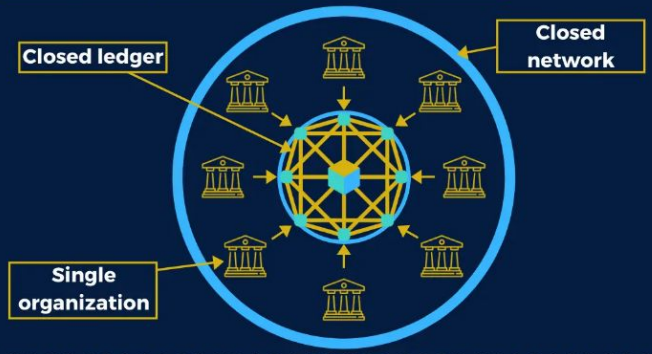
5.2 KEY CHARACTERISTICS OF PRIVATE BLOCKCHAIN

1. **Type of organization :** Single entity or organization
2. **Participants :** Known and trusted
3. **Access :** Fully restricted
4. **Type of network :** Centralized and single point of failure
5. **Operation :** Pre-approved participants can read/initiate transactions
6. **Verification :** A single validator node/central authority to create a block
7. **Immutability :** Secured by distributed consensus
8. **Consensus mechanism :** PoW or PoS consensus mechanism
9. **Security :** Depends on the blockchain architecture that is adopted
10. **Speed of transaction :** High. It takes a few seconds to create a block
11. **Energy consumption :** Low
12. **Scalability :** Better as high storage and computational power are not required.

PUBLIC BLOCKCHAIN

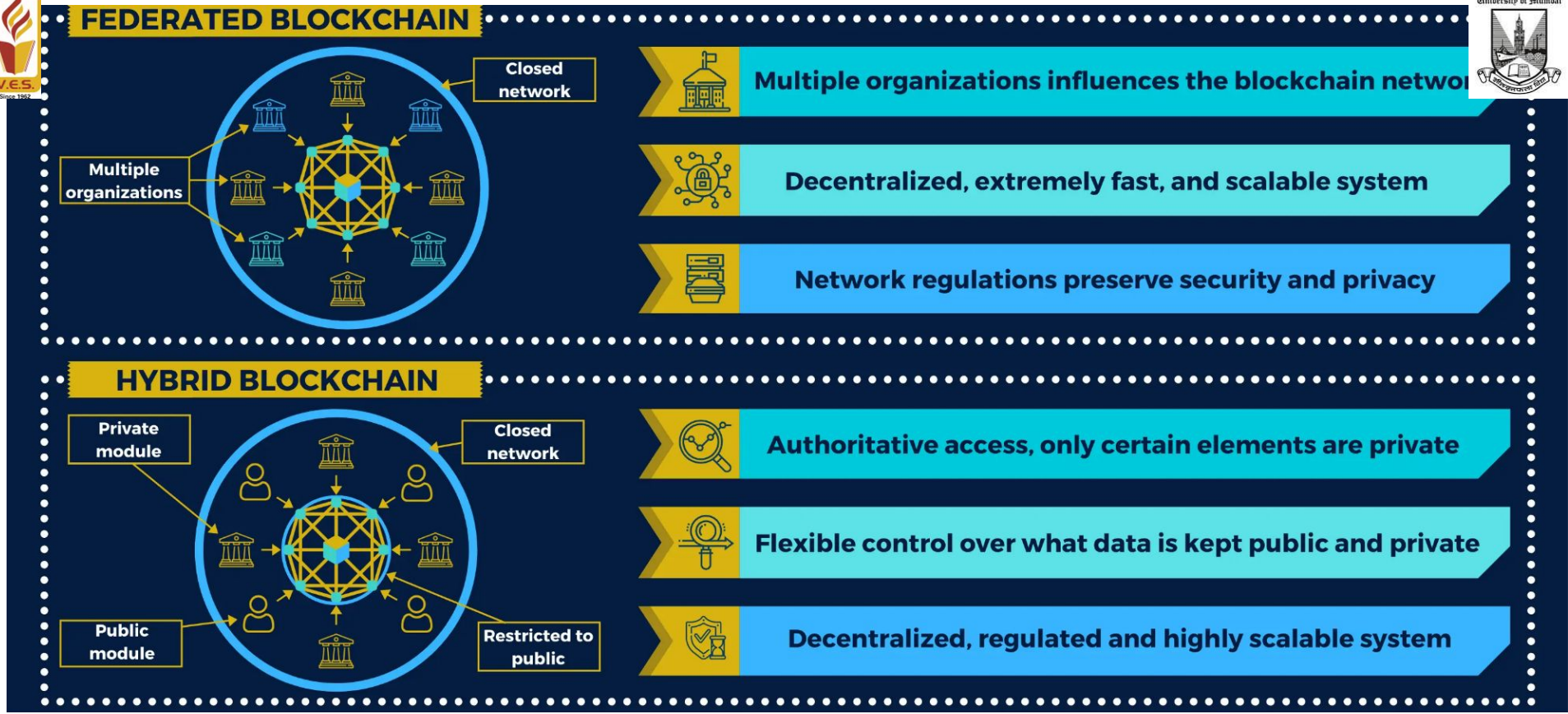
- Anyone is allowed to join and participate in the consensu
- Fully decentralized, secured and immutable ledge system
- Transactions are anonymous but transparent to everyone

PRIVATE BLOCKCHAIN



- A single organization will have authority over the network
- Faster output, power efficient, and offers privacy
- Simplified data handling process but not open to everyone

Courtesy : 101 Blockchain



Courtesy : 101 Blockchain

1.8.5 Comparison between Public, Private and Consortium Blockchain

Common Features	Public Blockchain	Private Blockchain	Consortium Blockchain
Network	Fully decentralized	Centralized	Partially decentralized
Accessibility	Open to anyone	Central Incharge or Single individual	More than one central Incharge
Failure	No single point of failure	Single point of failure	Multiple point of failure
Transaction speed	Slower	High	Very high
Participants	Everyone is anonymous	Known and trusted participants	Known and trusted participants
Energy Consumption	Very High	Low	Low

Common Features	Public Blockchain	Private Blockchain	Consortium Blockchain
Scalability	Limited	Better	Better
Computation power required	High	Low	Low
Trust	Trust-free	Trusted	Trusted
Efficiency	Low	High	High
Consensus determination	All miners	Single organization	Selected set of members
Immutability	Tampering is nearly impossible	Tampering can be possible	Tampering can be possible
Consensus Process	Permissionless	Permissioned	Permissioned
Time for Block creation	More than 10 minutes	In seconds	In seconds
Read Permission	Public	Public or restricted	Public or restricted
Security	Based on consensus protocols and hash functions.	Depends on blockchain architecture	Depends on blockchain architecture
Consensus Mechanism	PoW, PoS, etc.	Voting or different PoW/PoS consensus Algorithms	Voting or different PoW/PoS consensus Algorithms

VS

Founder	Vitalik Buterin	Satoshi Nakamoto
Release Date	30 July 2015	9 Jan 2008
Release Method	Presale	Genesis Block Mined
Total Coin Supply	Unlimited	21 Million
Blockchain	Proof of Work (Planning for Proof of Stake)	Proof of Work
Useage	Smart Contracts Digital Currency	Digital Currency
Scalable	Yes	Not Now
Mining	GPUs	ASIC Miners
Crypotocurrency Used	Ether	Bitcoin (Satoshi)
Intended Purpose	A platform for immutable, programmable smart contracts	To be store of value/medium of exchange
Algorithm	Ethash	SHA-256
Blocks Time	12 to 14 Seconds	At least 9 to 10 minutes

Fintra.co.in

Bitcoin	Ethereum
In Bitcoin SHA-256 is used in the algorithm of the Blockchain	In Ethereum, Keccak-256 or Keccak-512 are used. Keccak is a variant of SHA3.
Assets : Bitcoin	Assets : Ether
Bitcoin is exclusively used as an online currency.	Ethereum is often used as an online currency, however its blockchain is also utilized for decentralized applications such as voting systems, file storage, and so on.
Data related to Bitcoin network transactions is frequently utilized simply to keep track of transactions.	On the Ethereum network, for example, transactions may include executable code.

Bitcoin	Ethereum
Simple and robust	Complex and feature-rich
Bitcoin consumes much more energy than Ethereum.	Ethereum is certainly consuming an increasing amount of energy, but this will likely decrease significantly after it has transitioned to Proof-of-Stake.
Stack based primitive scripting language, not Turing-complete	Turing-complete scripting language
Key lengths Public key = 512 bits (compressed: 257 bits) Private key = 256 bits	Key lengths Public key = 512 bits Private key = 256 bits Address = 160 bits
Mining Proof-of-work	Mining Proof-of-Work to Proof-of-Stake
Signature size 512 bits (64 bytes)	Signature size 520 bits (65 bytes)
Energy consumption ≈ 3, 2 million households	Energy consumption ≈ 1, 1 million households
Transaction time Bitcoin is known for its slow and expensive transactions. It takes around 10 mins to complete a Bitcoin transactions.	Transaction time Ethereum transaction takes only 12 seconds.
Block time 10 minutes	Block time 12 seconds
Mining reward 12,5 Bitcoin per block	Mining reward 3 Ether plus some other rewards
Consensus time ≈ 1 hour	Consensus time ≈ 3 minutes

	XDC Network	Ethereum	Hyperledger Fabric	R3 Corda	Ripple	Quorum	Hyperledger Sawtooth	EOS	Hyperledger Iroha	Openchain	
Industry focus	Cross-Industry	Cross-Industry	Cross-Industry	Financial Services	Financial Services	Cross-Industry	Cross-Industry	Cross-Industry	Cross-Industry	Digital Asset Management	Financial Services
Ledger Type	Permissionless	Permissionless	Permissioned	Permissioned	Permissioned	Permissioned	Permissioned	Permissioned	Permissioned	Permissioned	Both Public & Private
Consensus Algorithm	XDC Delegated Proof-of-Stake	Proof of Work	Pluggable Framework	Pluggable Framework	Probabilistic Voting	Majority Voting	Pluggable Framework	Delegated Proof-of-Stake	Chain-based Byzantine Fault Tolerant	Partionned Consensus	Stellar Consensus Protocol
Smart Contract	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes
Governance	XDC Network	Ethereum Developers	Linux Foundation	R3 Consortium	Ripple Labs	Ethereum Developers and JP Morgan Chase	Linux Foundation	EOSIO Core Arbitration Forum(ECAF)	Linux Foundation	CoinPrism	Stellar Development Foundation

Platforms	Ethereum	Hyperledger Fabric	R3 Corda	Ripple	Quorum
Public/Private					
/Permissioned	Public Blockchain- No permission is needed to access network content.	Private Blockchain- Network is limited to people with permission.	Private Blockchain- Permission is needed to access network content.	Public Blockchain- Permissionless network content	Both Public, Private permissioned network content
Smart Contract	Solidity programming language	Golang programming language	Kotlin programming language	C++ programming language	Solidity and Vyper
Governance	Carried out by developers (DAO)	Linux foundation incharge	R3 company in charge	Ripple Labs	Etherum developers and JP Morgan Chase
Consensus	Relies on Proof of Stake for decision making	The consensus process does not require participation from every node in a network.	Making decisions is limited to those parties engaged in a transaction.	Participating nodes verify the authenticity of a transaction by conducting a poll.	Multiple voting based consensus mechanisms which do away with the current proof of work consensus mechanism used today by the public Ethereum blockchain among many others

Platforms	Ethereum	Hyperledger Fabric	R3 Corda	Ripple	Quorum
Consensus Protocol	PoW/ PoS	Puggable	Notary-based	Probabilistic voting	Pluggable
Currency	Ether	No native currency	No native cryptocurrency	XRP	ETH
Use Case	Popular with generalized applications and mostly used for P2P and B2C operations.	the preferred platform for B2B operations, mainly used in enterprise.	Runs on a specialized distributed platform for the financial industry needs.	Runs on a specialized distributed platform for the financial industry needs.	Popular with generalized applications and mostly used for P2P and B2C operations.