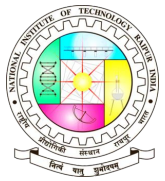
Distributed Peer-to-Peer(P2P) network

Dr. Preeti Chandrakar
Assistant Professor

Department of Computer Science and Engineering
National Institute of Technology, Raipur
September 2021

Outline

- Distributed network
- Distributed Hashtable
- Working of Distributed network
- Difference between centralized, decentralized and distributed network



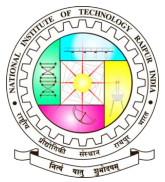
Distributed network

- A network configuration where every peer can communicate with one another without going through any centralized point or centralized server. Since there are multiple pathways for communication, the loss of any participant will not prevent communication.

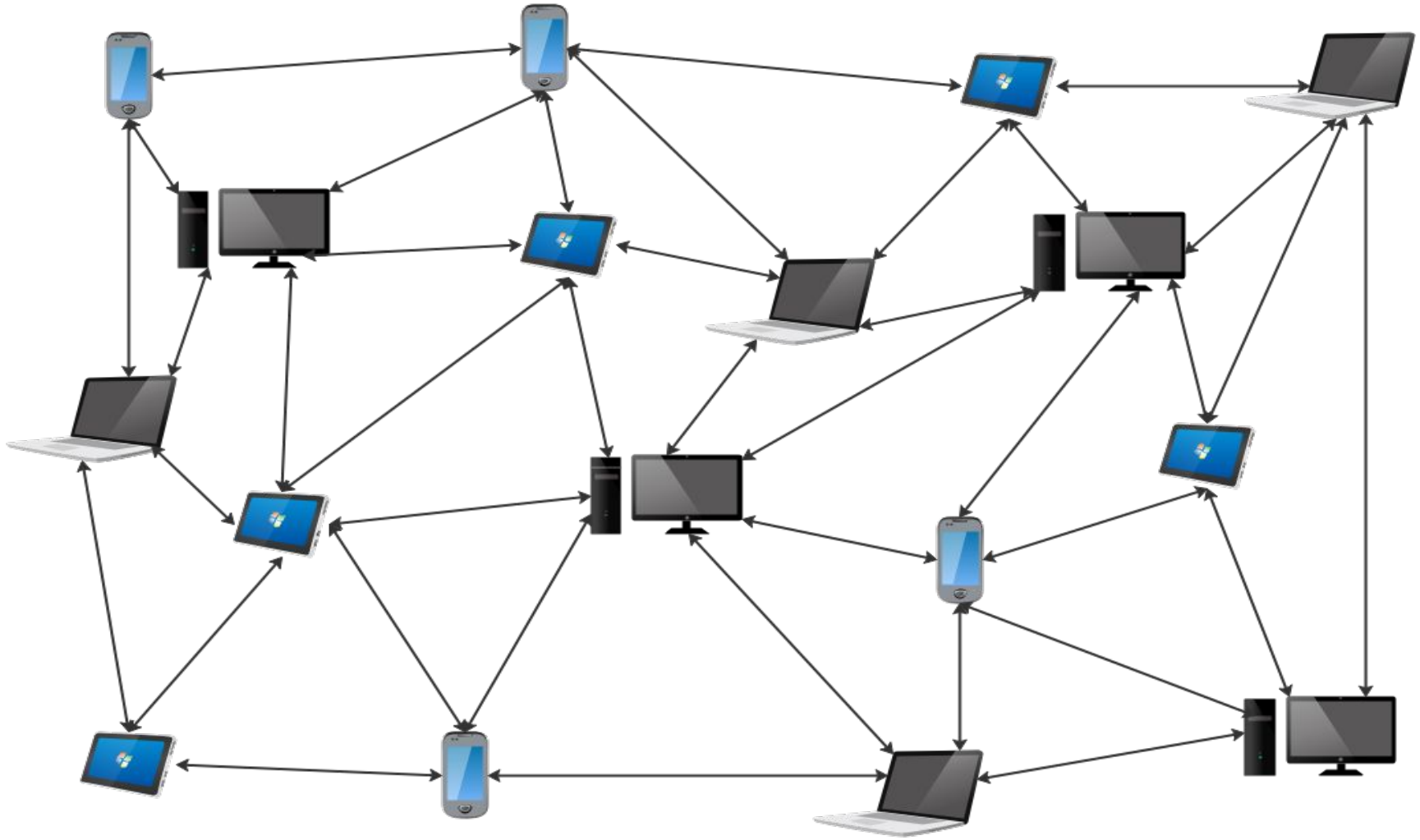
--NIST.IR.8202

Or

- A network of interconnected peers, that are able to communicate with each other without going through any central server or central authority. This network is prevented from single point of failure as there is no centralized server.

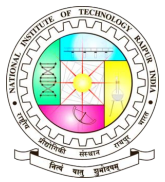


Distributed network



Distributed network

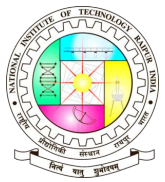
- Properties of the distributed networks:
  - No servers, means no supernode or central authority
  - Fault tolerance: if any node is disconnected at any time it dose not affect the whole network
  - Scalable : unlimited peers can join the network
  - High performance then centralized and decentralized networks
  - Peers are responsible to handle routing table or lookup on their own system
  - If any peer get defected or compromised, it is very hard to detect
  - Peers can join or leave network at any time without affecting the network
  - To maintain the network, it required high complex algorithms
  - It require high cost to implement the network
  - Secure in privacy point of view



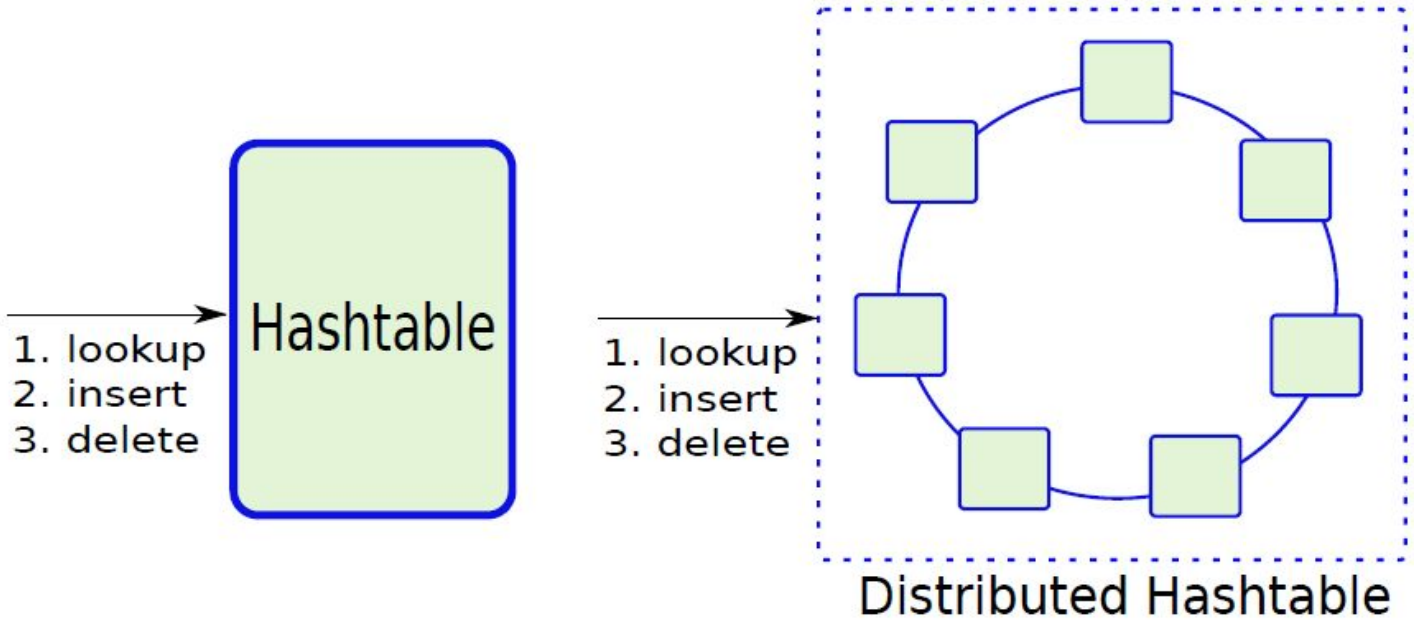
Distributed Hashtable

- **Hashtable:** Contains a set of key-value pairs. When the user provide the key, the hashtable returns the value which is associate with it.

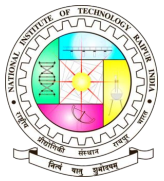
- Basic operations of hashtables are.
  - insert(key,value): Inserts the key,value pair into the hashtable.
  - lookup(key): Returns the value, or null if there is no value.
  - delete(key): Deletes the key.
- A hash is collision resistant, means that two different file can not have same hash value.



Distributed Hashtable

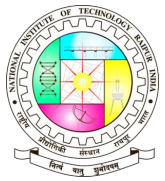


- From the above image we can see that, in distributed hash table number of multiple hashtables are connected with each other.
- This distributed hash table is used by the peers which are connected with each other in the distributed manner for file sharing in distributed network.



Properties of distributed Hashtable

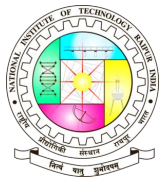
- They can store more data than centralized databases
- DHTs scale, and are ideal candidates for web scale storage
- They are more immune to node failures. They use extensive data replication
- DHTs also scale in terms of the number of users. Different users are redirected to different nodes based on their keys. (better load balancing)
- In the case of Torrent applications: they reduce the legal liability since there is no dedicated central server



Working of distributed P2P network

- Each node maintains its local data store in the system
- Dynamic routing table: address of other nodes and the keys that they (might) hold

  a node knows only about its immediate neighbors (not others)
- Queries are sent to a node that can pass it to its neighbors
- Each query has a TTL (time-to-live field) that is decremented at every hop



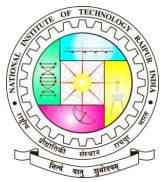
Routing table

Key	Address	Content ?
Key	Address	Content ?

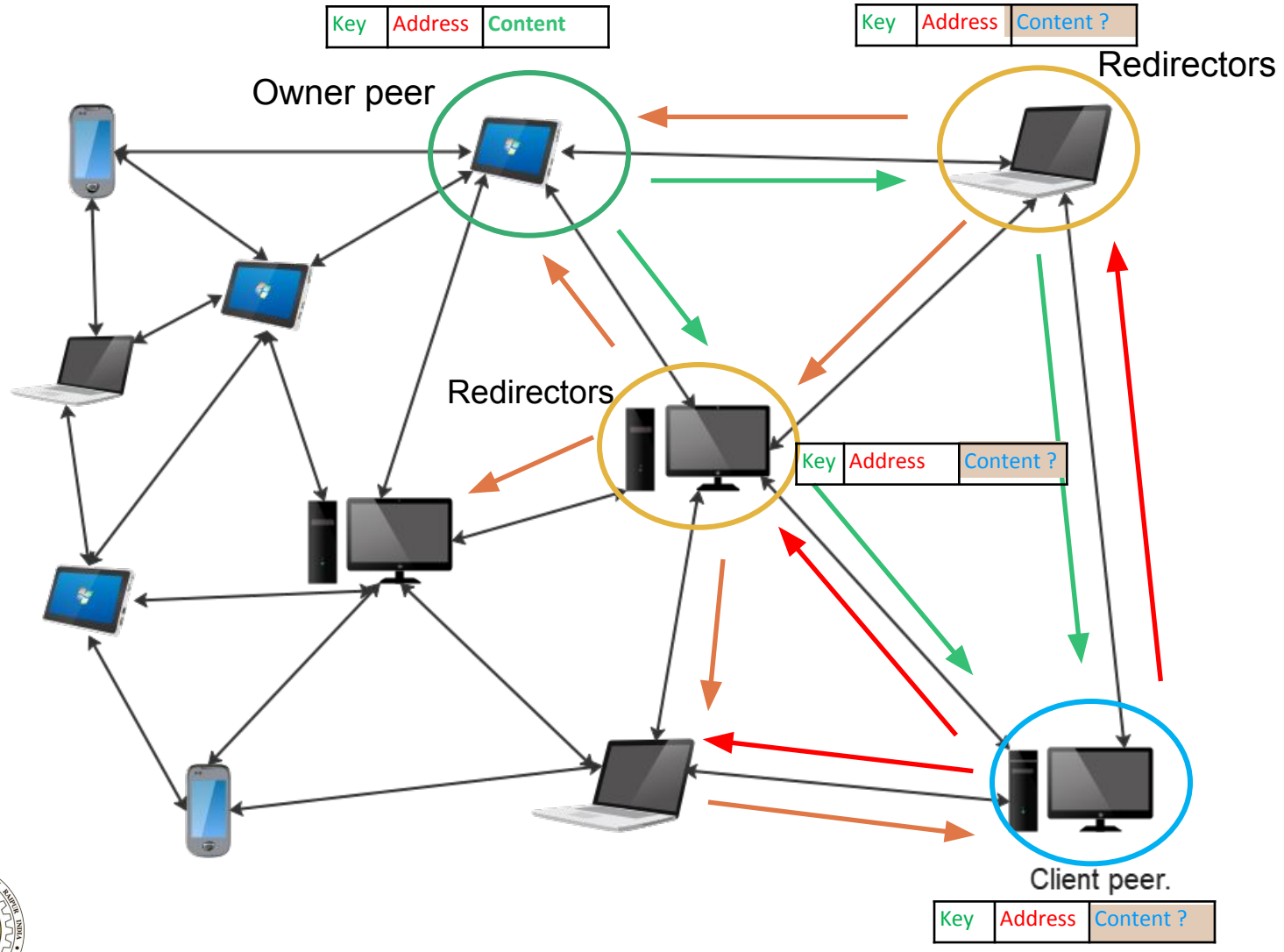
- **Key**: Hash of File name
- Example : Spider-Man: No Way Home.avi >

  ac8639c9ae9ad276363a8312c517a99b8235724c0faa6d94b4f805abfa44494a
- **Address**: IP of the peer neighbor peers stored in hash format.
- Example: 162.182.52.19:1006 >

  0025a3023fdefef329b725de81abdf6b35558c6cf4eec682ca823bfa87a5d909
- **Content**: The actual file or data



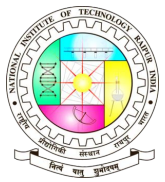
Working of distributed P2P network



1. Hashes the name of the file and and store in routing table, use this key to find the file.
2. Request file to owner with hash key.
3. Check key in it's routing table, if not found then, redirecting to the nearest node.
4. Redirecting.
5. Search the key in routing table and if find the file then respond.
6. Getting the requested file in response.
7. Strore the file and key in its routing table, and send to client peer, preetended as owner of the file.
8. Client store the file against it key in routing table.

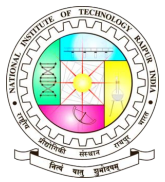
Working of distributed P2P network

1. The client peer hashes the name of the file. This is the key in its routing table
2. It looks up the key that is closest to the key, and passes the request to its owner
3. If a peer finds the file, it returns the contents, along with its address (saying that it is the owner of the data)
4. Otherwise, it finds the nearest key in its routing table, and forwards the request to that peer
5. If the next peer have the file then response to the same path from the request tis come
6. If the request is ultimately successful, then the peers on the way will:
    I. Send the file to the client peer and,
    II. Create an entry in their routing tables, and record the original source of request



Working of distributed P2P network

7. If a node cannot forward the request to another node:
    I. Creates a cycle that redirect the request to client node or
    II. Failure
8. Try the key with the second closest distance
9. At every hop decrease the TTL till it reaches 0
    II. To reduce the network load, the TTL can be dynamically decreased
    III. Nodes can decided to process which request next based on the TTL



Difference between centralized, decentralized and distributed network

S.No.	Centralized	Decentralized	Distributed
1.	Having one centralized server for communication between the peers	Having multiple decentralized servers for communication between the peers	This is server less network protocol.
2.	The peers are controlled by the centralized server for indexing the filename and IP addresses	The peers are controlled by the decentralized multiple servers for indexing the metadata of the file.	The nodes are maintain their routing table by self for connected neighbor nodes.
3.	If the centralized server is down then the whole network is affected.	If any server is down, it dose not affect the whole network.	If any node is disconnected it dose not effect the whole network.
4.	The Ip address is known by the centralized server and the peer only.	The Ip address is known by the multiple servers and also to the connected peers.	The security is high as the IP of peer and the file is in the hash format.
5.	If any peer is compromised by any attacker then the whole network is compromised, because it leads to centralized server.	If any peer is compromised by any attacker then only that peer and connected server will affected and the other part of network is running normal.	If any node compromised, It dose not affect the whole network,

Difference between centralized, decentralized and distributed network

S.No.	Centralized	Decentralized	Distributed
6	Its implementation is taking less cost because of having single server.	Its implementation is taking huge cost because of having multiple servers.	The cost of Its implementation is high then decentralized.
7	The communication process is faster because of short path to create communication between server to client peers.	The communication process is slower because of going through multiple path to create communication between source to client peers.	The communication process is slower then centralized and faster then decentralized network.
8	Scalability issues to connect more peer as centralized server is having limited size of storage.	No Scalability issues to connect more peer as it's having multiple servers and can add more when it required.	No Scalability issues to connect more peer .
9.	Load on the network is very high because of every peers are have to go through the centralized server for further communication.	Load on the network is normal because of any peers have to go through the only server connected to it, for further communication.	Load on the network is very low because of the nodes are maintain their own routing table and only directed to its neighbor nodes.
10	Example: Napster	Example: BitTorrent, Kaza, Grouckster.	Example : Blockchain(bitcoin)

Blockchain Technology

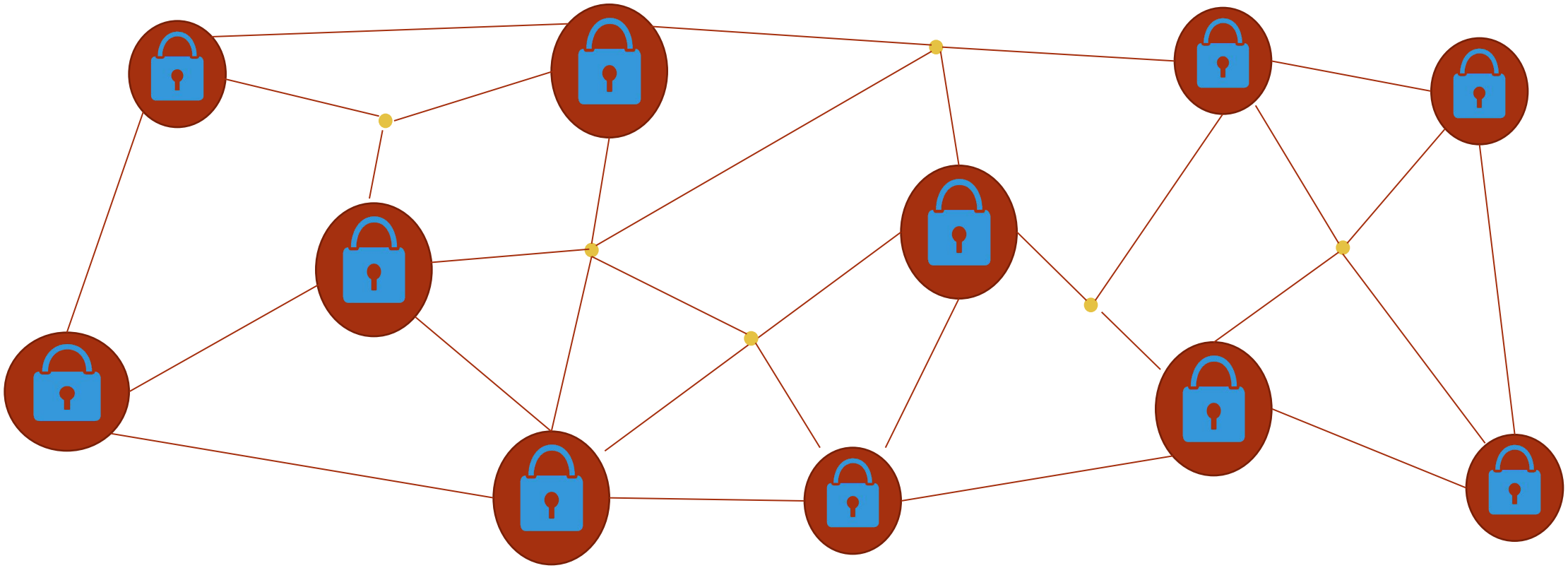
Dr. Preeti Chandrakar

NIT- Raipur

(C.G)

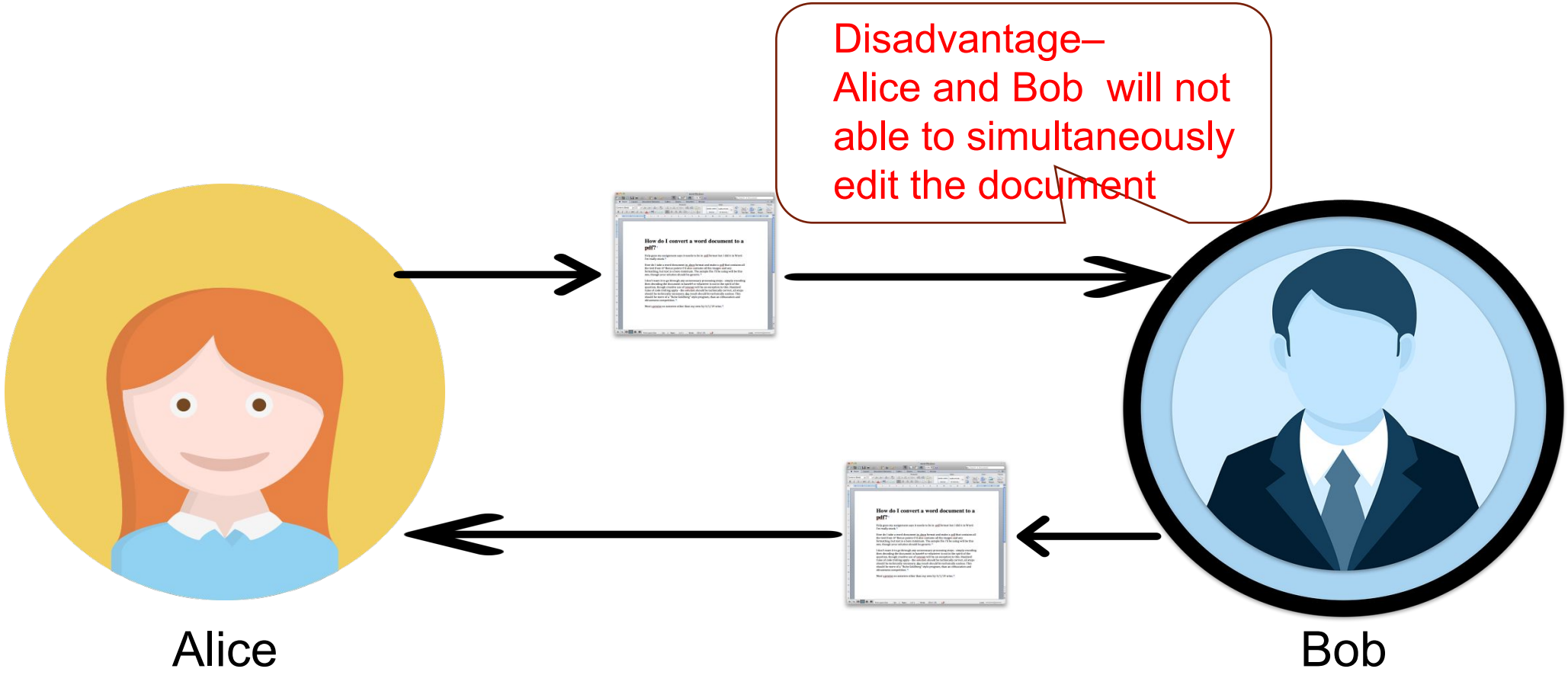
What is Blockchain

- A decentralized computational and information sharing platform that enables multiple authoritative domains, who do not trust each other, to cooperate, coordinate and collaborate in a rational decision making process



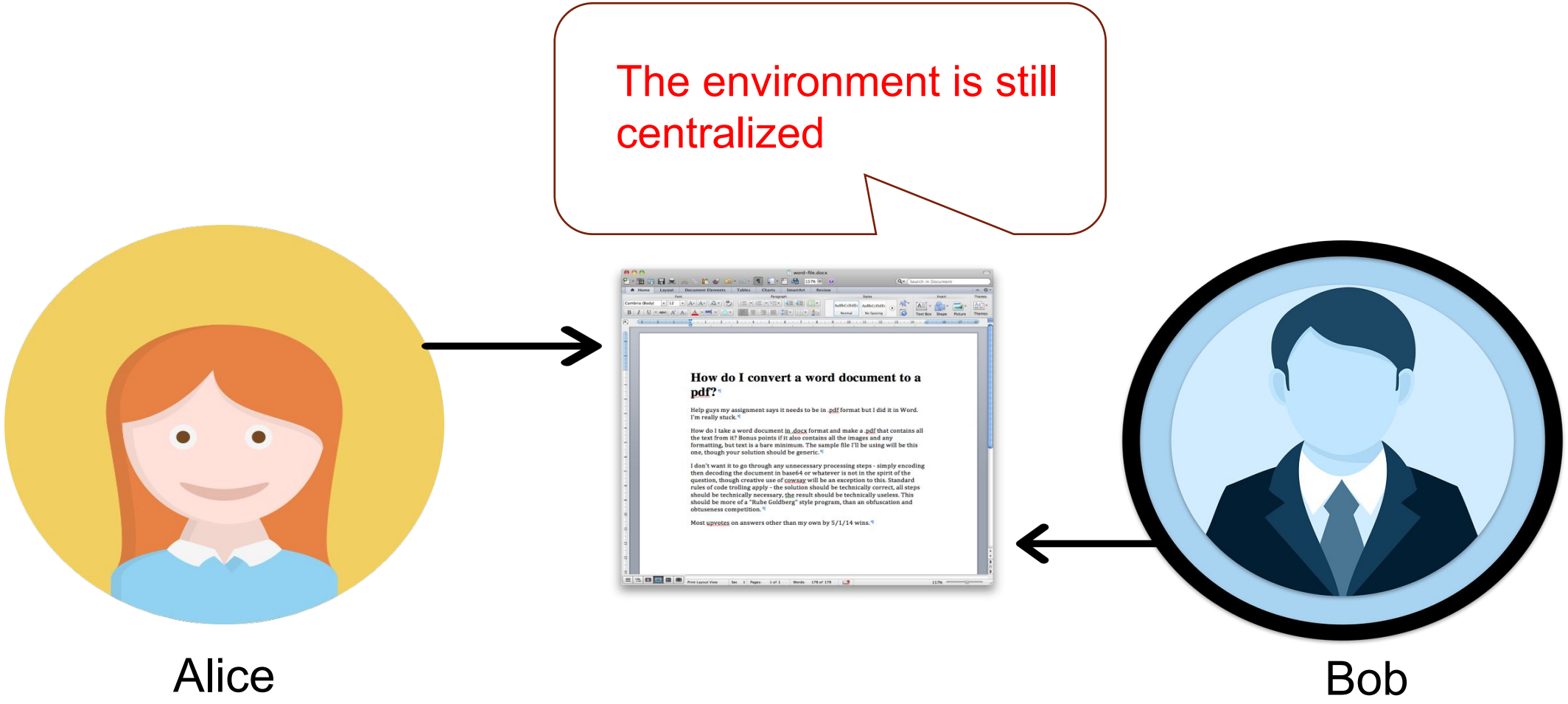
Microsoft Word to Google Doc- Sharing Information

- Traditional way of sharing documents- using Microsoft word



Microsoft Word to Google Doc- Sharing Information

- Shared Google doc- both the users can edit simultaneously



Problems with a Centralized System

- A single point of failure
  - If we do not have sufficient bandwidth to load Google doc, we will not be able to edit
  - Complete reliance on single point (centralized) is not safe

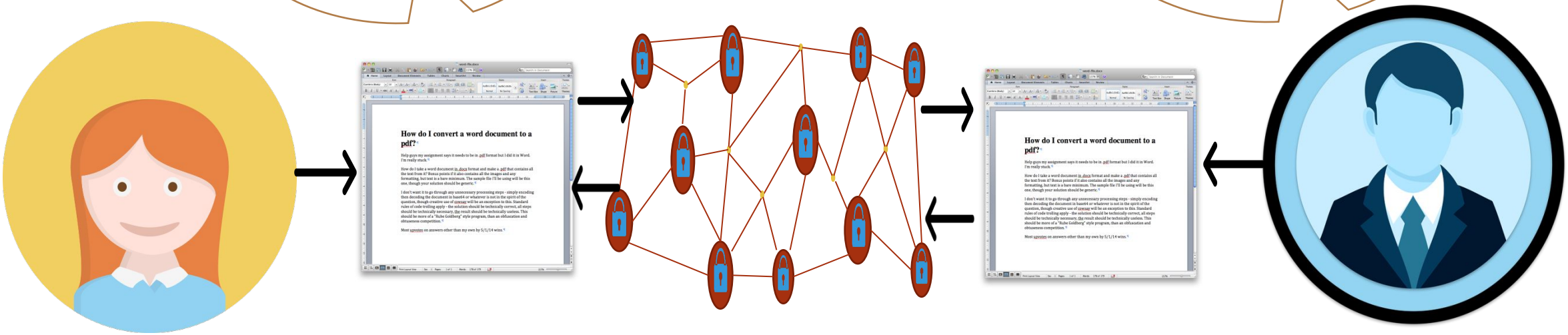
- Solutions
  - Decentralized: Multiple points of coordination
  - Distributed: Everyone collectively execute the job, tolerate multiple number of failures, until the network becomes disconnected

Ideal Solution Using Blockchain

Both users have their own copy of document

Decentralized database with storage consistency support

Every one edits on their local copy of the document



Alice

Blockchain

Bob

A Very Simplified Look of the Blockchain

- Every node maintain a local copy of the global data-sheet
- The system ensure consistency among the local copies
  - The local copies at every node is identical
  - If these node want to enter some information to this blockchain , so this information will get updated to all the copy of the blockchain that every node passes
  - The local copies are always updated based on the global information

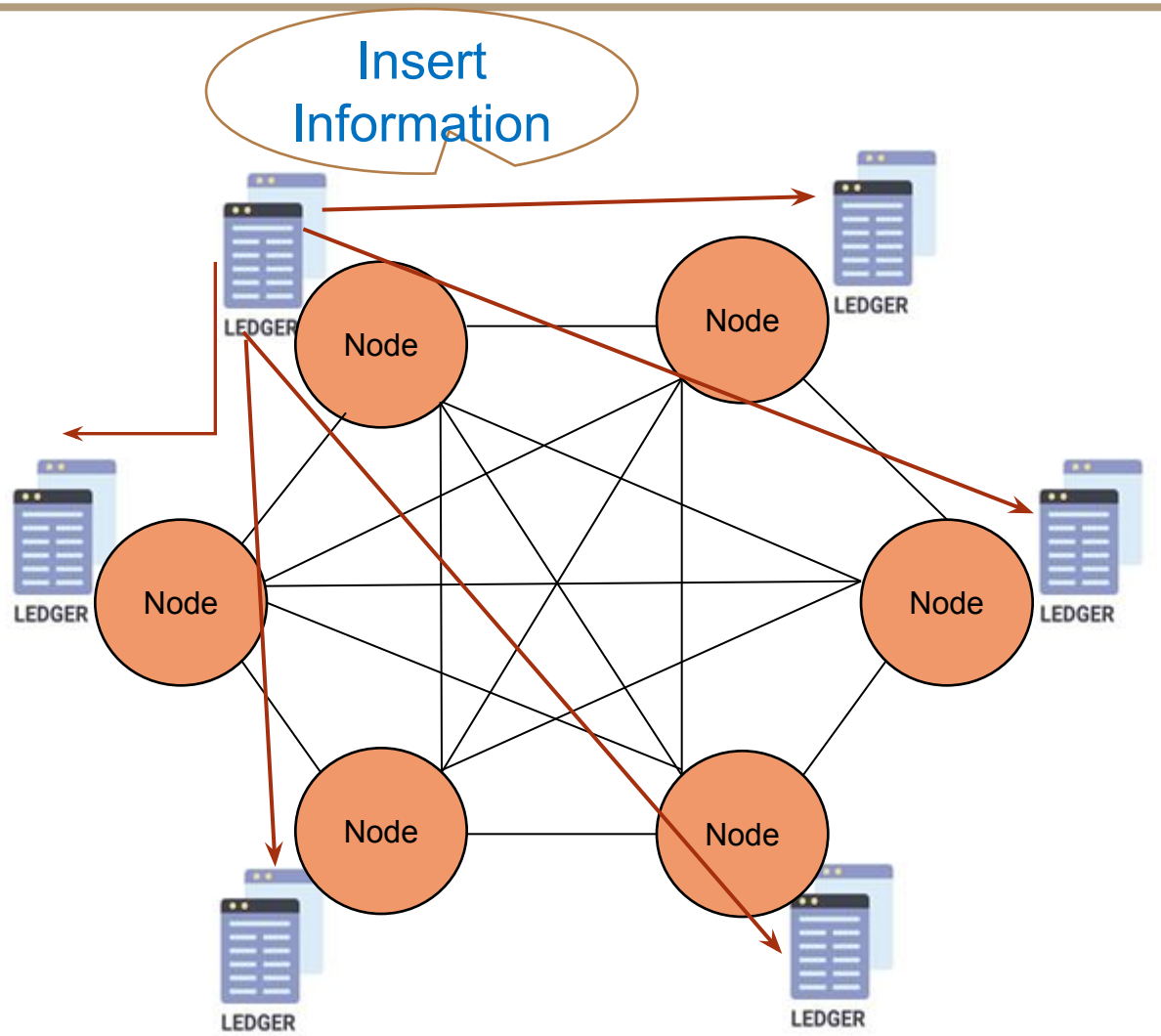


Fig 1: Blockchain network

A Very Simplified Look of the Blockchain

- We call this a Public Ledger
  - A database of "historical information" available to everyone
  - The "historical information", may be utilized for future computation
- An Example
  - Say, the historical information are the banking transactions
  - The old transactions are used to validate the new transactions

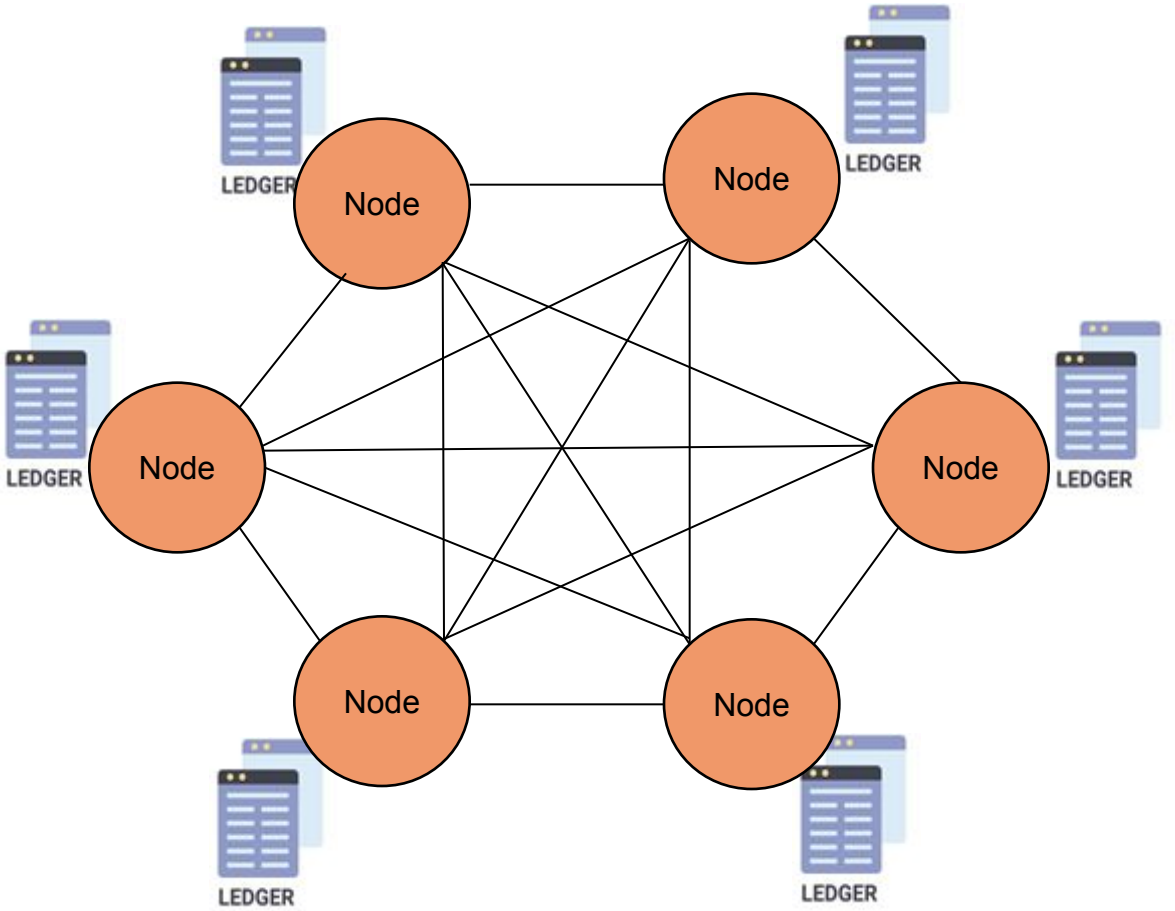


Fig 2: Blockchain network

An Example of Public Ledger from Banking Sectors

Public Ledger of Alice

Alice: Rs 100

Alice

Rs 100

Bob

Alice: Rs 100

Public Ledger of Bob

Public Ledger of Eve

Alice: Rs 100

Eve

Jane

Alice: Rs 100

Public Ledger of Jane

An Example of Public Ledger from Banking Sectors

Public Ledger of Alice

Alice: Rs 100
Alice - Bob Rs 50

Alice

Rs 50

Bob

Alice: Rs 100
Alice - Bob Rs 50

Public Ledger of Bob

Public Ledger of Eve

Alice: Rs 100
Alice - Bob Rs 50

Eve

Jane

Alice: Rs 100
Alice - Bob Rs 50

Public Ledger of Jane

An Example of Public Ledger from Banking Sectors

Public Ledger of Alice

Alice: Rs 100
Alice -□ Bob Rs 50
Bob-□ Eve Rs 30

Alice

Bob

Alice: Rs 100
Alice -□ Bob Rs 50
Bob-□ Eve Rs 30

Public Ledger of Bob

Rs 30

Public Ledger of Eve

Alice: Rs 100
Alice -□ Bob Rs 50
Bob-□ Eve Rs 30

Eve

Jane

Alice: Rs 100
Alice -□ Bob Rs 50
Bob-□ Eve Rs 30

Public Ledger of Jane

An Example of Public Ledger from Banking Sectors

Public Ledger of Alice

Alice: Rs 100
Alice -□ Bob Rs 50
Bob -□ Eve Rs 30

Alice has Rs 50

Alice

Rs 80

Bob

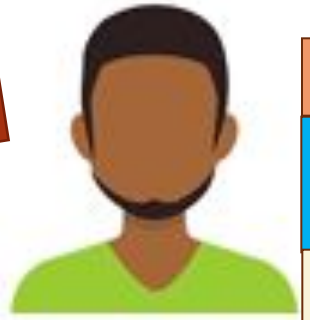
Public Ledger of Bob

Alice: Rs 100
Alice -□ Bob Rs 50
Bob -□ Eve Rs 30

Public Ledger of Eve

Alice: Rs 100
Alice -□ Bob Rs 50
Bob -□ Eve Rs 30

Eve



Jane

Public Ledger of Jane

Alice: Rs 100
Alice -□ Bob Rs 50
Bob -□ Eve Rs 30

Blockchains and Public Ledgers

- Blockchains work like a public ledger
- However, we need to ensure a number of different aspects
  - **Protocols for Commitment**
    - Ensure that every valid transaction from the clients are committed and included in the blockchain within a finite time
  - **Consensus**
    - Ensure that the local copies are consistent and updated
  - **Security**
    - The data needs to be tamper proof. If client may act maliciously or can be compromised
  - **Privacy and Authenticity**
    - The transactions belong to various clients; privacy and authenticity needs to be ensured

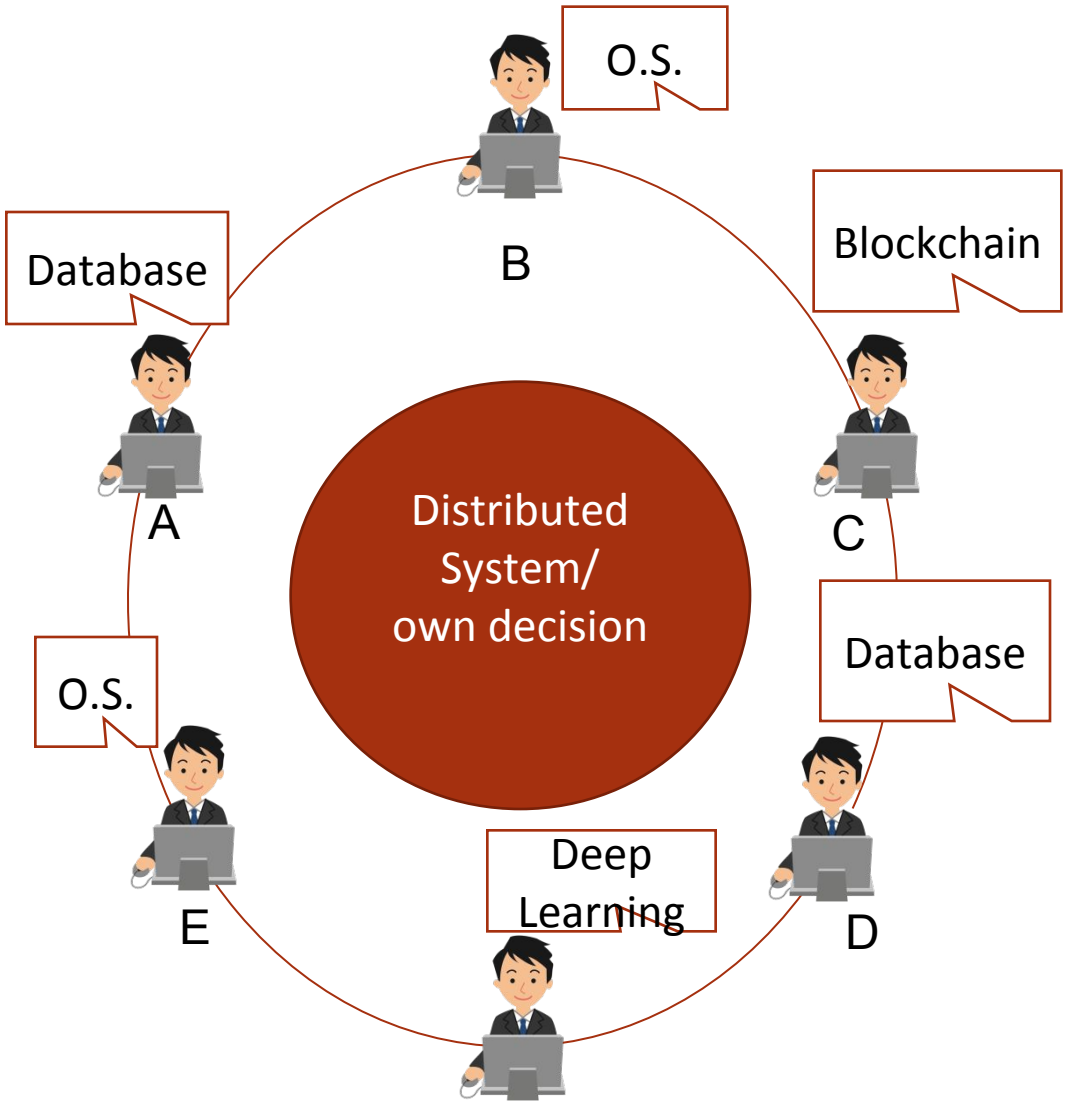
Formal Definition of a Blockchain

- A blockchain is "an open distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way"
- Keywords
  - **Open**- Accessible to all
  - **Distributed or Decentralized** – No single party control
  - **Efficient-** Fast and scalable
  - **Verifiable**- everyone can check the validity of information
  - **Permanent**- The information is persistent

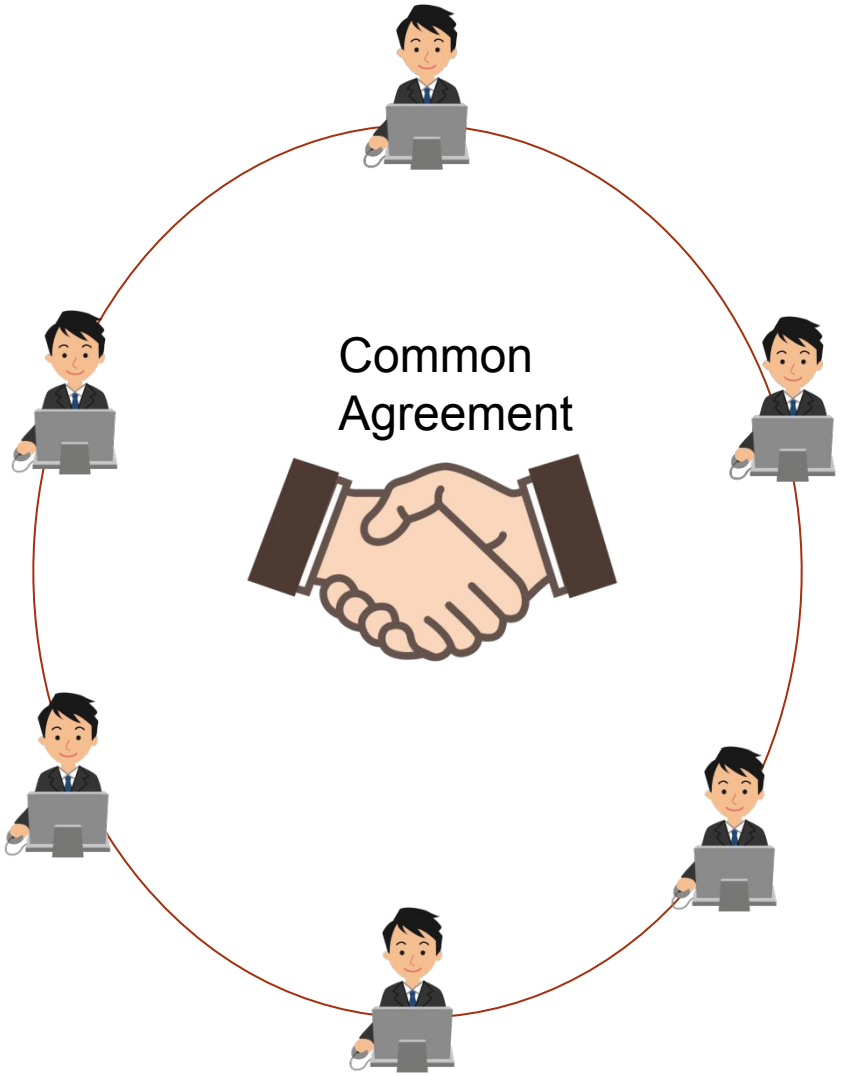
History of Blockchain

- Satoshi Nakamoto is inventor of bitcoin (decentralized money) and also generator of first blockchain
- In 2008 Satoshi Nakamoto in his white paper “Bitcoin: A Peer-to-Peer Electronic Cash System”
- First bitcoin transaction is take place in 3rd Jan 2009
- Bitcoin not a blockchain
- Bitcoin is the digital token and blockchain is the ledger that keep of who owns the digital tokens
- We can’t have bitcoin without blockchain but we can have blockchain without bitcoin

Consensus in Distributed System



Need a common decision the entire distributed system may look like a single system to an outsider



Consensus

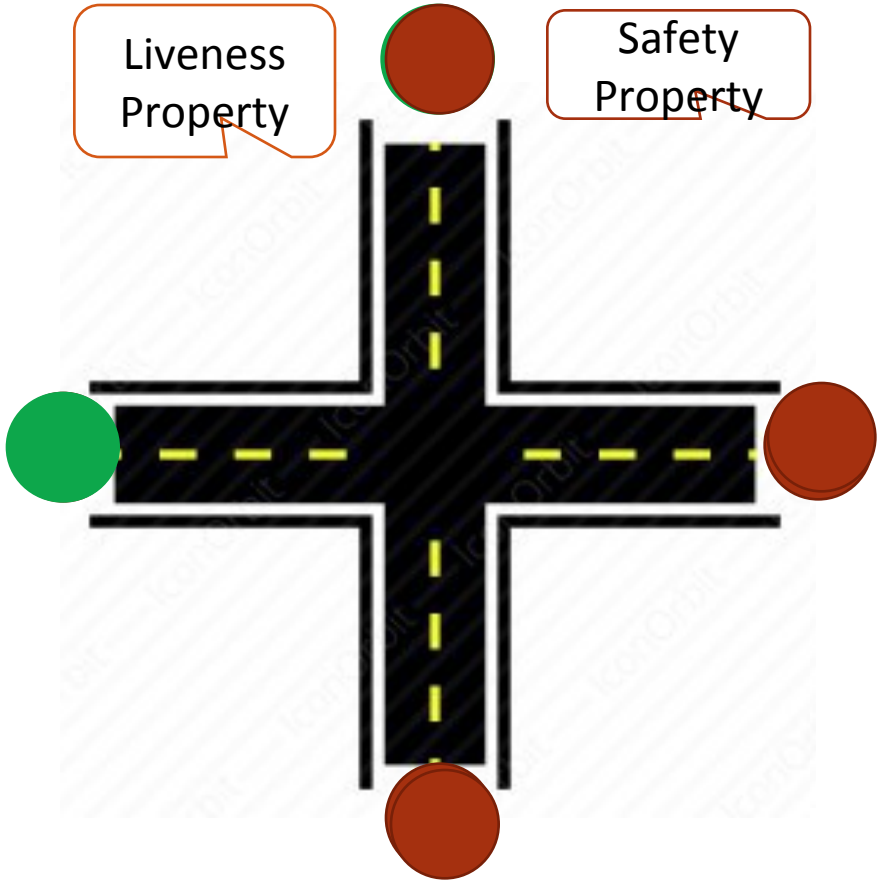
- A procedure to reach in a common agreement in a distributed or decentralized multi-agent platform
- Consensus algorithm is important for a message passing environment in a distributed system
- In distributed system each process might propose a value
- Processes co-ordinate to agree upon one value, V
- V needs to be proposed by at least one process
- All processes need to agree upon V
- It uses to solve the distributed commit problem
  - Each process can proposed two values- commit or abort
  - All process agree to either commit or abort
- Run multiple copies of a computation, and use voting to decide the result

Why Consensus

- In a traditional or conventional distributed system we apply consensuses to ensure reliability and fault tolerance
- In distributed or decentralized environment, we have multiple individual parties and they can take their own decision, then it may happened that some nodes are working as maliciously or faulty
- So, in those cases it is important to come to a common decision or a common view point
- Under this kind of distributed environment our objectives is to ensure reliability, that means, to ensure correct operations in the presence of faulty individuals
- Consensus algorithm is important in distributed system because the entire distributed system may look like a single system to an outsider

Why Consensus

- **Safety Property:** Something wrong does not happen
  - If the traffic light on one road is green, then the traffic light on the particular road is red
- **Liveness Property:** Somethings good always happens
  - The red light ultimately turn green
- So we need protocols that never violate the safety constraint
- Liveness is a secondary criteria



Example of a Consensus Mechanism

- 4 generals, generals are taking the decisions in a decentralized on a distributed way
- They have their own policies to take the decision
- They can either make an attack or can reflect from the attack
- From their individual opinion by applying some kind of choice function, which can be the majority decision in this case
- Applying such choice function, 3 generals they are making a choice towards attack
- So, with the majority principle the system can come to a consensus that they should make a attack collectively


Attack

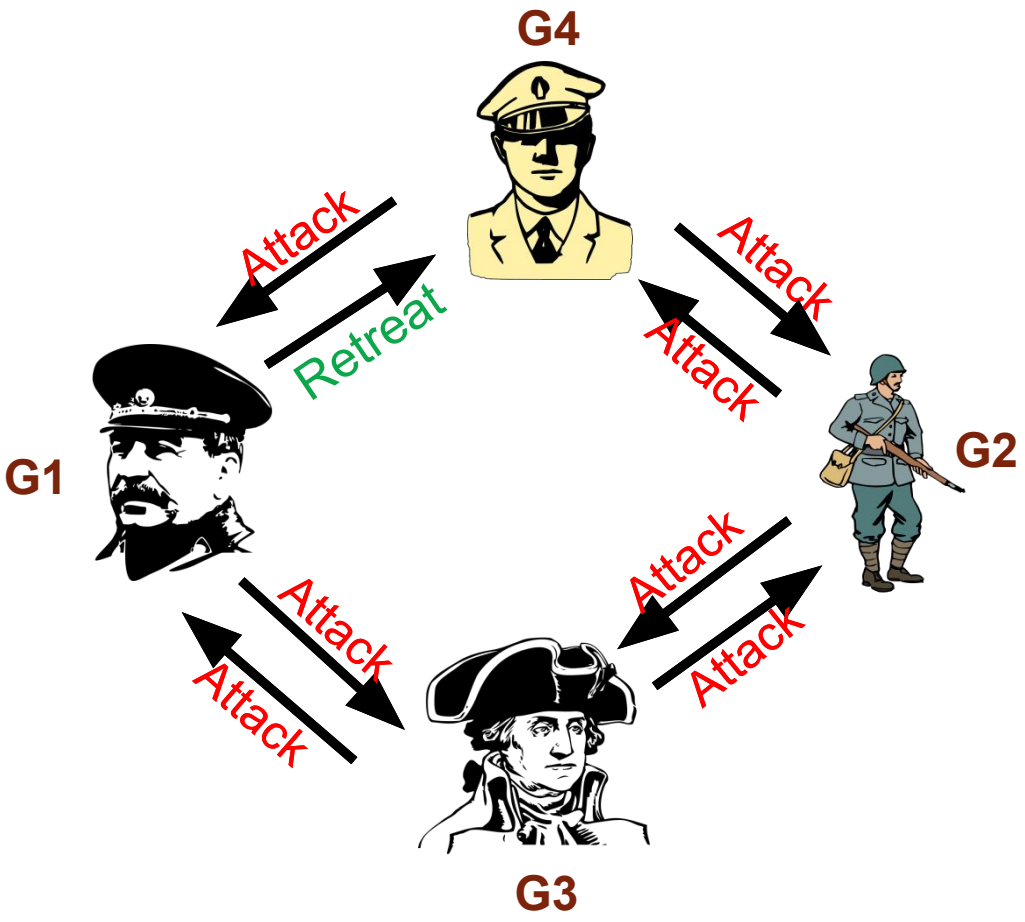

Retreat


Attack


Attack

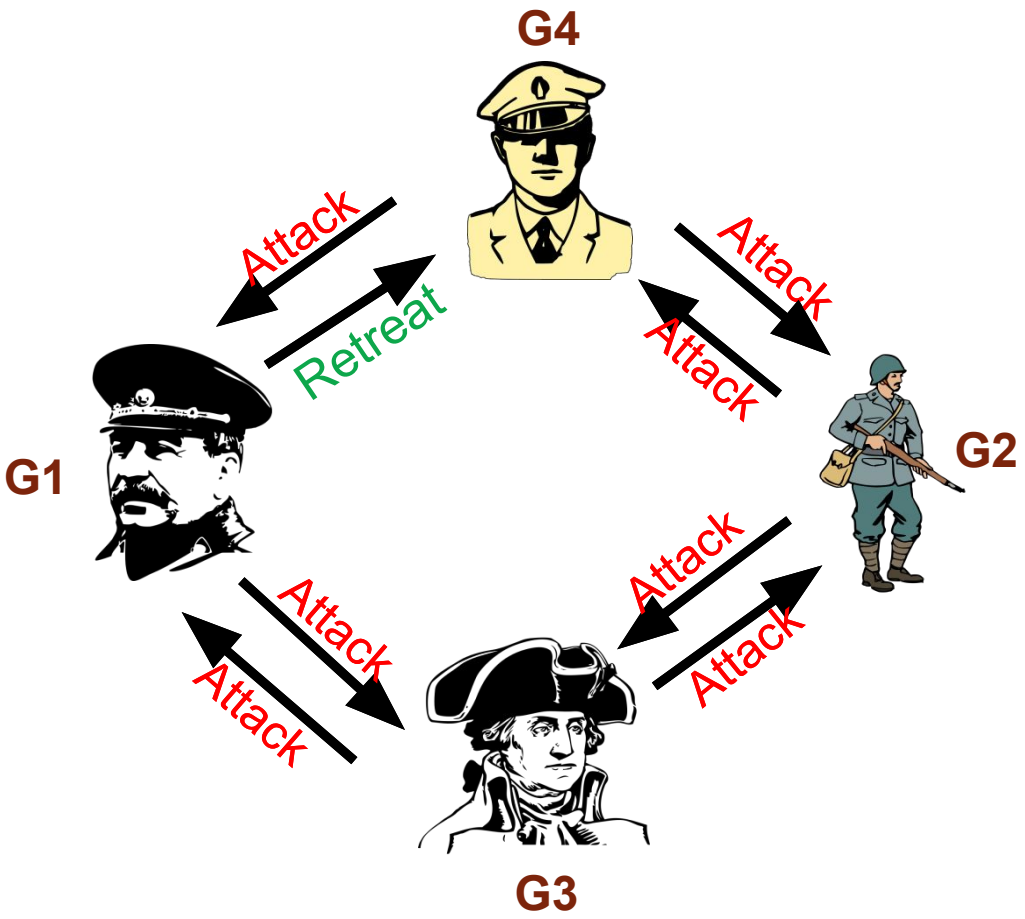
A Consensus is Difficult in Faulty Environment

- General G4 is sending the information the general makes a telephone call to other 2 general G1 and G2.
- G3 he again makes a phone call to G2 and G1 and send his decision that the soldiers should now make an attack
- G1 is a malicious general, he makes a call to G4 for retreat and call to G3 saying as attack
- A complete distributed or decentralized environment the generals may get confuse
- G4 get confused that whether or what he should do?



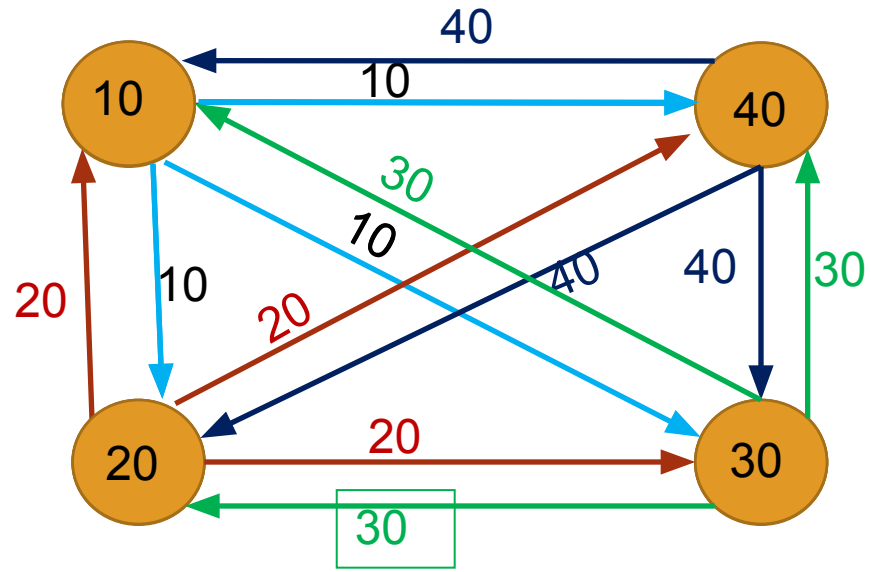
A Consensus is Difficult in Faulty Environment

- In a decentralized or a distributed platform achieving consensus over a message passing system can be difficult when a malicious node is present
- This malicious node called as the byzantine node
- This kind of failure is called as byzantine failures



Distributed Consensus

- If there is no failure, it is easy and trivial to reach in a consensus
- Broadcast the personal choice to all
- Apply a choice function(consensus), say the maximum of all the values
- Let 4 nodes make a choice of 10,20,30,40, they are individual choices to all other nodes
- Every node receives all the choices from all the neighbors
- They can apply on max function to find out that what is the maximum
- Every one will reach to the value 40



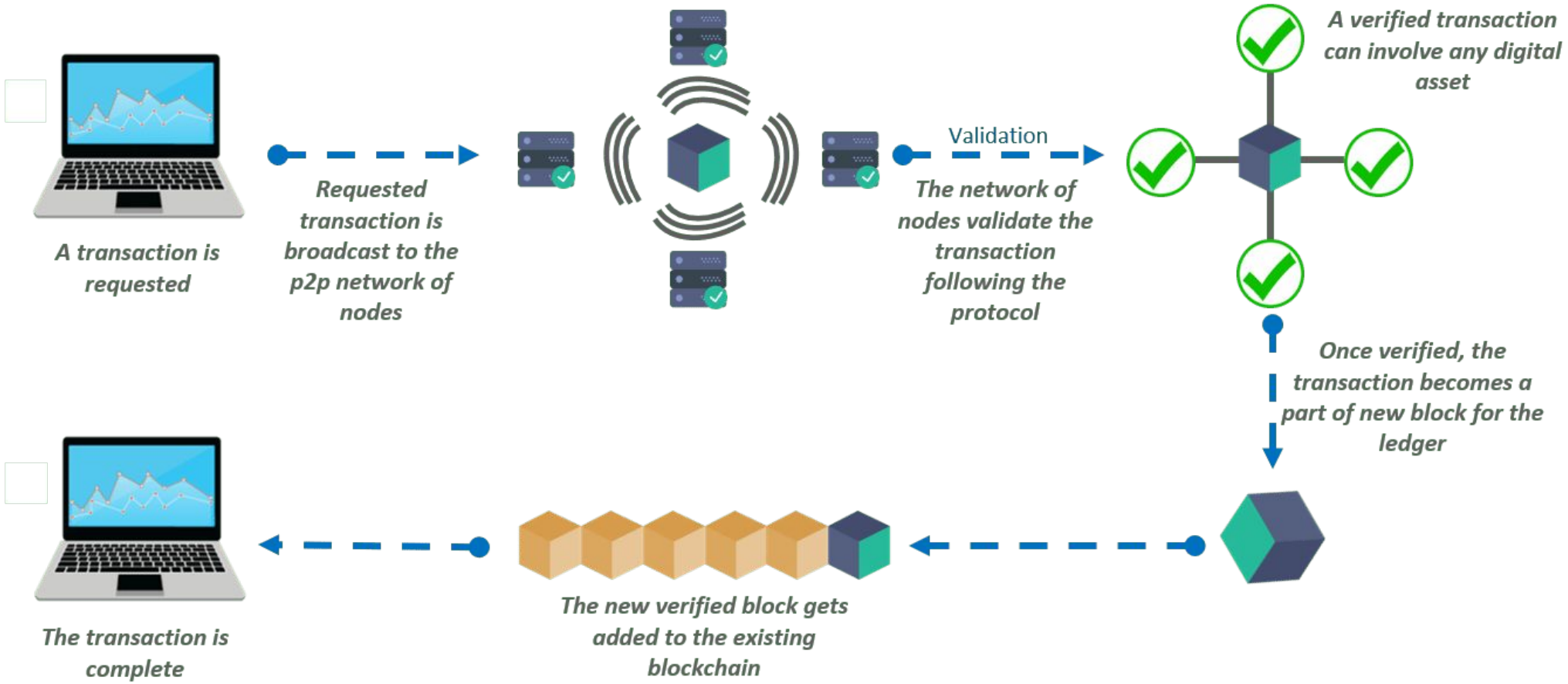
Distributed Consensus

- There can be various type of faults in a distributed system
  - **Crash Fault**
    - A node suddenly crashes or becomes unavailable in the middle of a communication
  - **Network or Partitioned Faults**
    - A network fault occurs(link failure) and the network gets partitioned
  - **Byzantine Faults**
    - A node starts behaving maliciously

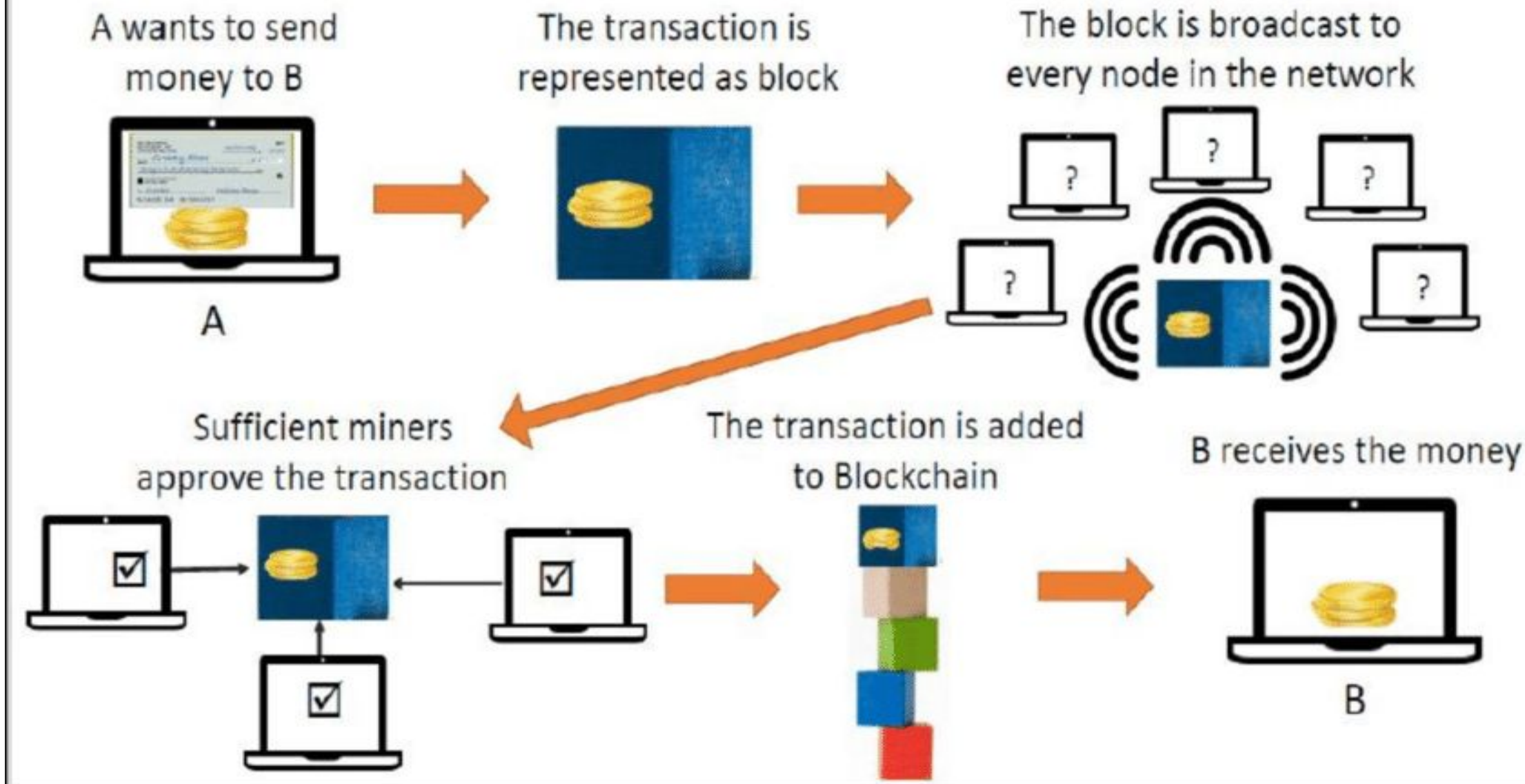
Distributed Consensus

- Properties
  - **Termination**
    - Every correct individual decides some value at the end of the consensus protocol
  - **Validity**
    - If all the individuals proposes the same value, then all correct individuals decide on that value
  - **Integrity**
    - Every correct individual decides at most one value, and the decided value must be proposed by some individuals
  - **Agreement**
    - Every correct individual must agree on the same value

How BLOCKCHAIN works



How Blockchain works?



Main Ingredients of Blockchain

Initiation and Broadcasting of Transaction	•Digital Signatures •Private/Public Keys
Validation of Transaction	•Proof of Work and certain alternatives
Chaining Block	•Hash Function

Key Characteristics of Blockchain

- Decentralization
- Immutable
- Public Ledger
- Transparency
- Anonymity
- Openness
- Validation(Consensus)

Immutability — The ability for a blockchain ledger to remain a permanent, indelible, and unalterable history of transactions

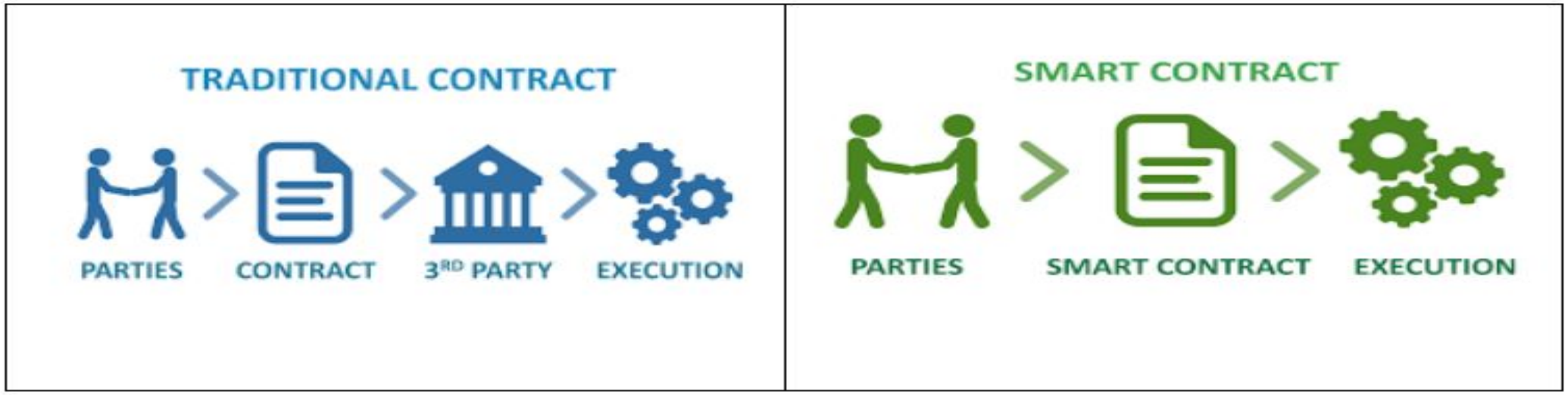
No anyone modify the data or transaction once they are recorded in the block chain data base.

Annoymity- suppose a person sends a sum of money, then the receiver will get to know that the sender is linked to a bitcoin address but will not know the actual address. Hence, we say that bitcoin or any other alt currencies are not entirely anonymous.

Bitcoin is considered pseudoanonymous, which means a person may be linked to a public Bitcoin address, but not to an actual name or home address.

Smart Contract

- A Smart Contract is an agreement between two people in the form of computer code. They run on the blockchain, so they are stored on a public database and can not be changed.
- The transactions that happen in a smart contract are processed by the blockchain which means they can be sent automatically without a third party.



Traditional Vs Smart Contract

	Traditional Contract	Smart Contract
Third Party	Government, Lawyers etc……	None
Execution Time	1-3 days	Minutes
Remittance	Manual Process	Automatic Process
Transparency	Unavailable	Available
Archiving	Difficulty	Easy
Security	Limited	Cryptographically Secure
Cost	Expensive	Cheap
Signature	Manual Process	Digital Signature

Condt…

- Smart contracts help to exchange “money", property, shares or any things of value in a transparent, conflict-free way while avoiding the services of a middleman.
- Pre-written logic in the form of computer code.
- Stored and replicated on the blockchain.
- Executed and run by the network of computers running the blockchain.
- Contract written in a specific programming languages are compiled into bytecode which an EVM can read and execute.

Condt…

- Smart contracts are self-executing contracts which contain the terms and conditions of an agreement between the peers.

The terms and conditions of an agreement is written in code

These agreements facilitate the exchange of money , shares ,properties etc.

It executes in blockchain’s decentralized platform

Condt..

- Let’s consider an example where Rachel is at airport and her flight is delayed.

Cont….

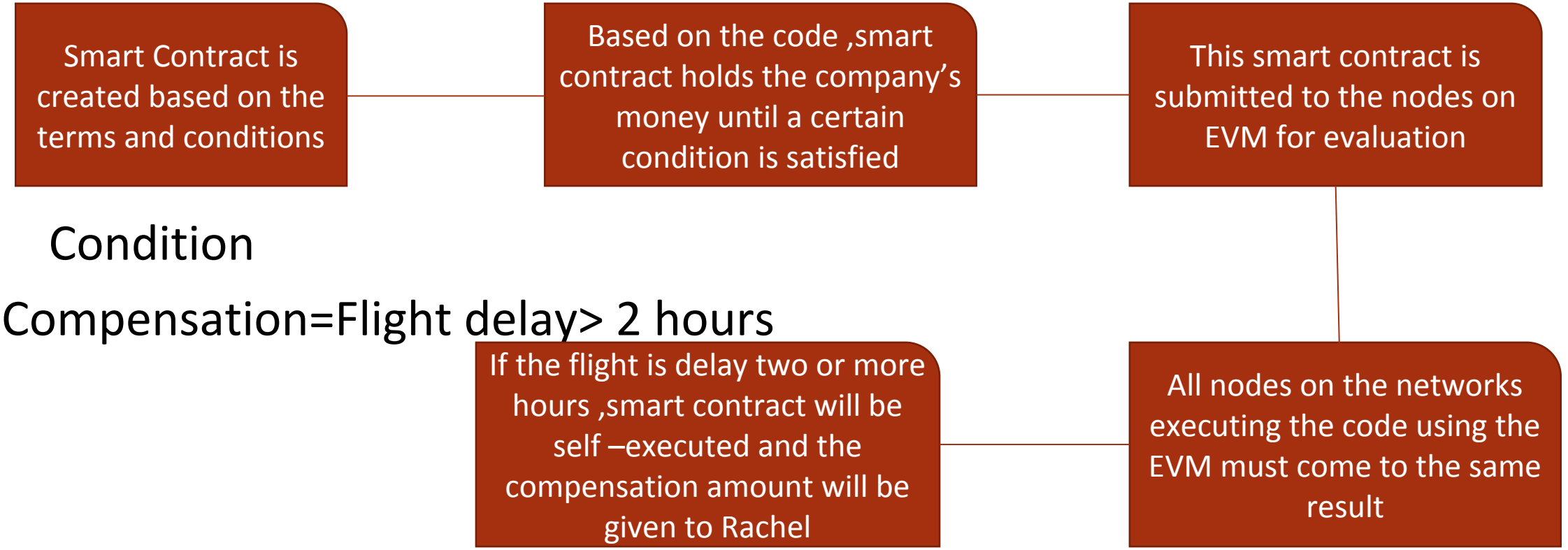
AXA flight delay insurance is one of the examples of Ethereum Smart Contract

The smart contract is linked to the databases that records flight status

It enable automatic compensations when there is delay of two hours or more

- Note- AXA is an insurance company

Cont….



Condition

Compensation=Flight delay> 2 hours

Advantages of Smart Contract

- No intermediaries – The process executes without the need of a third party.
- Automated- They are automated with the code which eliminates manual effort for execution.
- High Speed- Since smart contract run on programming code the speed of its execution is higher than the traditional contract.
- Secure- As data is stored in the decentralized system, the chances of modifying data is difficult.
- Accuracy- Based on the requirements, terms and conditions of a contract is recorded accurately.

Consensus Algorithm

- A consensus algorithm is a procedure through which all the peers of the Blockchain network reach a common agreement about the present state of the distributed ledger.
- Consensus algorithms achieve reliability in the blockchain network and established trust between unknown peers in a distributed computing environment.
- The consensus protocol make sure that every new block that is added to the Blockchain is the one and only version of the truth that is agreed upon by all the nodes in the Blockchain.
- In the context of blockchain, consensus is the valid agreement for adding the new blocks in the blockchain network.