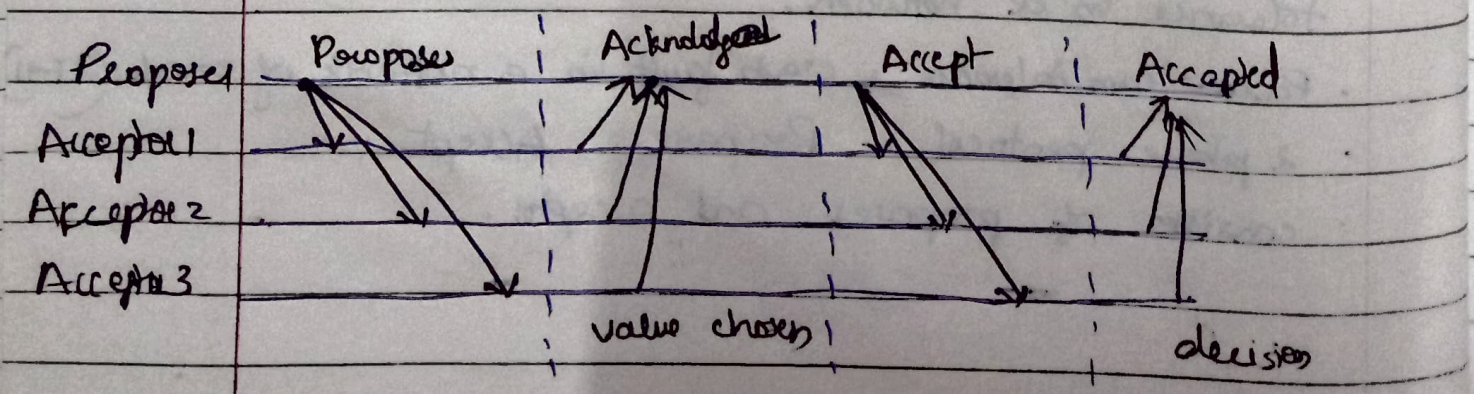
How Paxos Work

- It assumes an asynchronous msg-passing network with less than 50% of crash faults.
- properties – Safety and liveness
- In Safety – : Agreement : no 2 diff values are agreed
  Validity – only proposed values are agreed
- In liveness – Termination – protocol can decide & terminate

- Processes can assume 3 diff role:
  ① Proposer – elected leader that can propose a new value to be decided.
  ② Acceptors – provide a majority decision.
  ③ Learners – nodes that observe the decision process & value

note · A single process can assume all 3 roles.

→ The proposer node proposes a value, which is final only if majority of acceptor nodes accept it. The learner node learn this final decision.



Page No.
Date

1. Proposer propose a value by broadcasting a msg <prepare(n)>
2. Acceptors respond with acknowledgement msg if proposal n is highest that acceptor has responded so far. <ack (n,v,s)>
3. If majority is received proposer sends "accept" msg <accept(n,v)>
4. If majority acceptors accept, agreement is achieved.
5. Finally, in learning phase, acceptors broadcast <accepted(n,v)> to proposers

How PBFT works

1. client sends a request to invoke a service operation in the primary
2. primary multicasts the request to backups.
3. Replias execute the request and send a reply to client
4. client waits for replies from all replicas with same result.

Pre-Prepare Sub-protocol :

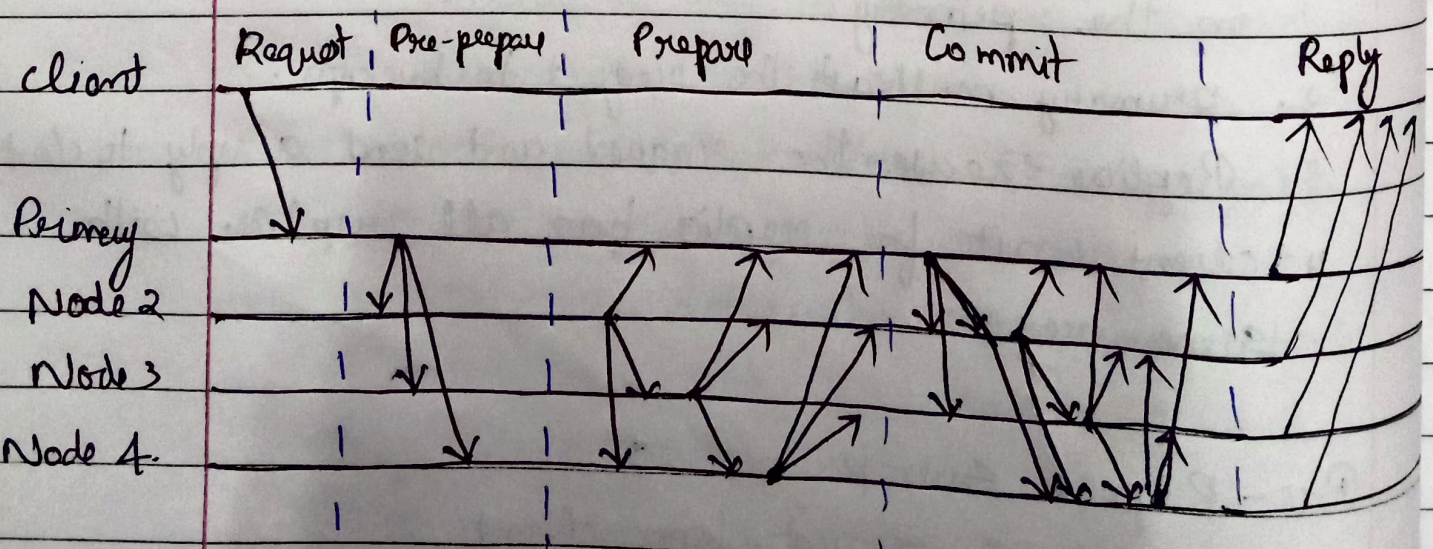
1. Accepts a request from client
2. Assigns the next sequence number
3. Sends the pre-prepare msg to all backup.

Prepare sub-protocol Algo:

1. Accepts pre-prepare msg. - If the backup has not accepted any pre-prepare msgs for the view no. sequence number, accept the msg
2. Sends the prepare msg to all replicas

Commit sub-protocol Algorithm:

1. replica wait for 2F prepare msgs with same view, sequence and request
2. Sends commit msg to all replicas
3. Wait until a 2F+1 valid commit msg arrives and is accepted.
4. Executes the received request
5. Send reply containing result to client.



Page No.
Date

5) ~~N = 5~~ Byzantine fault nodes, F = 5

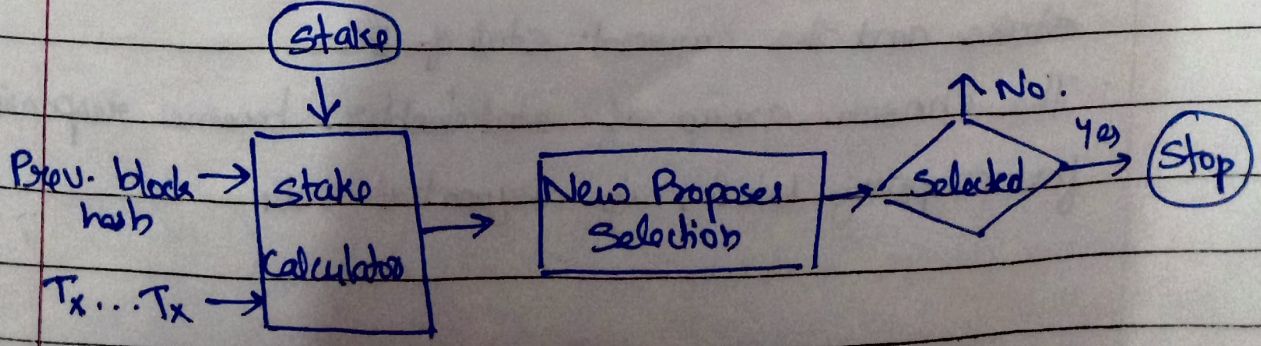
Min number of nodes, ~~N ≥ 3F~~ N = 3F + 1

~~N ≥ 3 × 5 = 15~~ = 3 × 5 + 1

= 15 + 1 = 16

* POS - Proof of Stake.

- alternative to POW
- first used in Peercoin
- Ethereum will soon become POS based.
- Stake represents the number of coins (money)
- If someone has a stake in the system, they will not try to sabotage the system
- Select a stakeholder and grant appropriate rights to it based on staked assets, Once stake is calculated, and stakeholder is selected to propose a block, block proposed by the proposer is readily accepted. The probability of selection increases with a higher stake.
- Higher the stake, better the chances of winning the right to propose the next block.



Types of PoS

a) Chain Based PoS

- only change from PoW mechanism is the block generation method.
- A block is generated in a steps :
  1. Transactions are picked up from memory pool and a candidate block is created
  2. A clock is set with a constant tick interval and at each tick, whether the hash of block header concatenated with clock time is less than product of target value and stake value is checked.

     Hash (B || clock time) < target x stake value.
- Hashing puzzle is solved at regular intervals based on the clock tick.

b) Committee based PoS

- A group of stakeholders is chosen randomly usually by using a verifiable random function (VRF)
- produces a random set of stakeholders based on their stake and the current state of BC.
- The chosen group of stakeholders become responsible for proposing blocks in sequential order

Page No.
Date / /

c) Delegated PoS
- similar to committee-based PoS
- Instead of using a random function, group is chosen by stake delegation
- create blocks in a round robin fashion.
- Delegates are chosen via voting by n/w users.
- Votes are proportional to amount of stake that participants have.
- not decentralized.

d) Hot Stuff.
- latest class of BFT protocol.
- better than traditional PBFT
- developed in 2018
- properties are
  - Linear View Change
  - Optimistic response
  - Chain Quality.

Page No.
Date

6. Transaction Validation:

- performed by bitcoin nodes
- These are 3 main things nodes check -

① That transaction inputs are previously unspent
This validation step prevents double spending by verifying that the transaction inputs have not already been spent by someone else.

② That sum of the transaction outputs is not more than total sum of inputs. Both can be same, or sum of input could be more than total value of output

③ That the digital signatures are valid.

16a Various fields in transaction of bitcoin:

① Input - contains information about the Bitcoin address from which the funds are being transferred. Includes address, amount of funds being transferred and a digital signature to verify that transaction is valid.

② Output :- info about bitcoin address to which funds are transferred. include address and amount of funds.

③ Fee :- amount of bitcoin paid to network as transaction fee. The fee is paid to miners to include transaction in the blockchain

④ Timestamp - contains the time at which the

transaction was created.

⑤ Transaction ID :- contains unique identifier for transaction. generated using a hashing algo and is used to identify the transaction on bitcoin n/w.

⑥ Block height :- contains height of block. Each block has a unique height which is determined by number of blocks that came before it.

⑦ Block hash :- contains hash of the block in which transaction is included. It is used to link the current block to the previous block in the blockchain

16 b) Role of a Bitcoin Miner:

① Syncing up with the network -
Once a new node joins the Bitcoin n/w, it download the Blockchain by requesting historical blocks from other nodes.

② Transaction validation
Transactions are validated by full nodes by verifying and validating signatures and output.

③ Block Validation
Miners and full nodes can start validating blocks received by them by evaluating them against certain rules.

④ Create a new Block

Page No.
Date

⑤ Perform PoW -

core of mining process where miners find a valid block by solving a computational puzzle.

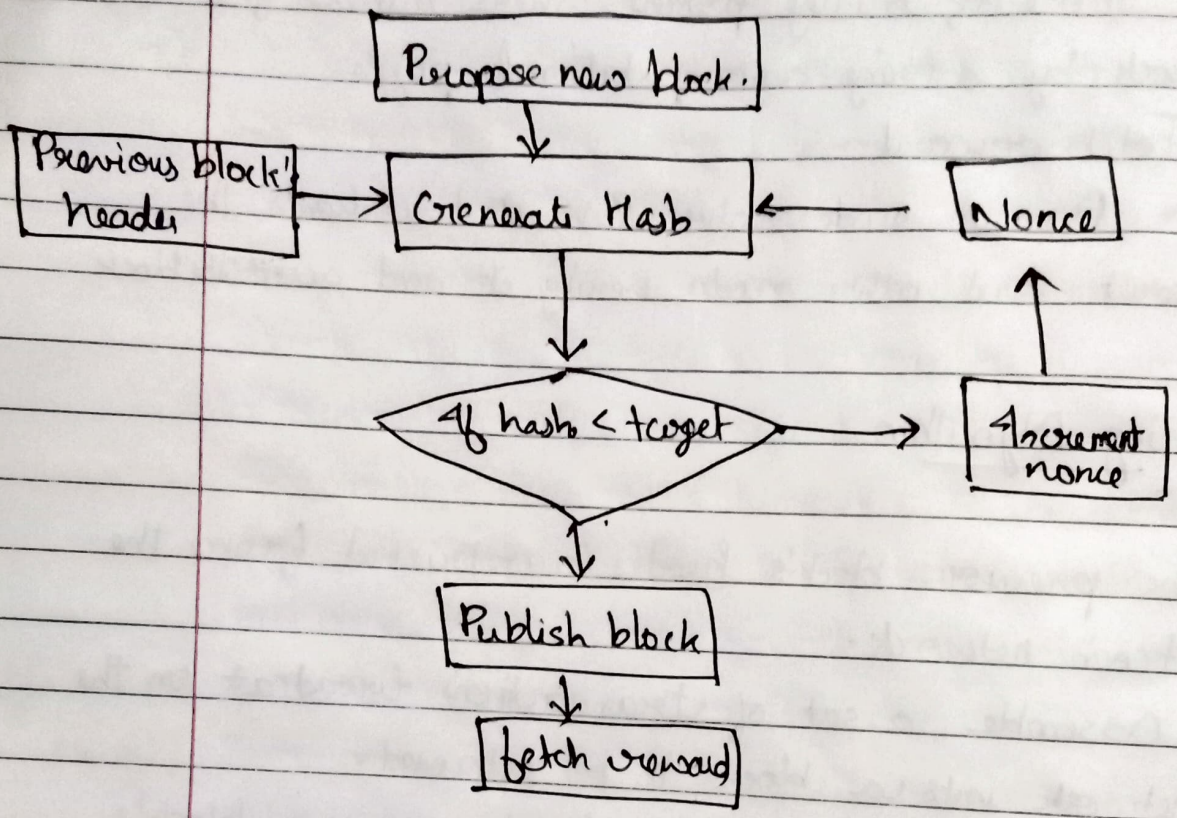
⑥ Fetch reward -

Once a node solves PoW, it broadcasts the result and other nodes verify it and accept the block.

Mining Algorithm

1. The previous block's header is retrieved from the bitcoin network.
2. Assemble a set of transactions broadcast on the network into a block to be proposed.
3. Compute the double hash of previous block's header, combined with a nonce and the newly proposed block, using the SHA256 algo.
4. Check if resulting hash is lower than the current difficulty level then PoW is solved.
   As a result of successful PoW, the discovered block is broadcasted to the n/w and miners fetch the reward.
5. If the resultant hash is not less than the current difficulty level, then repeat process after incrementing the nonce.

Flowchart:



Wallets

Wallet s/w is used to generate and store cryptographic keys & performs various useful functions such as receiving and sending Bitcoins, backing up keys, and keeping track of balance available.

- diff types of wallet to store private key:

① Non-deterministic Wallets

contain randomly generated private keys and are also called Jwt a Bunch of Key wallets.

The bitcoin core client generates some keys when

first started and also generates keys as when needed

② Deterministic Wallets

Keys are derived from a seed value via hash function.

③ Hierarchical deterministic Wallets.

store keys in a tree structure derived from seed

④ Brain Wallets

Master private key can also be derived from the hashes of passwords that are memorized

⑤ Paper Wallets.

paper-based with the required key material printed on it

⑥ Hardware wallets

⑦ Online "

⑧ Mobile "

Module - 4

7) Smart contracts can be used to enforce agreements btw parties in the form of business logic by using pre-defined rules and logic to govern the terms of an agreement and automatically execute transaction when certain conditions are met.

By automating, smart contracts can help to reduce the need for intermediaries such as lawyers or escrow services, increasing efficiency and transparency in the process.

Examples of how smart contracts can be used for enforcing agreements include supply chain management, real estate transactions and intellectual property rights

8) Blockchain based digital ID cards:

- Digital identity is not only limited to govt-issued ID cards but also online social networks and forums
- Blockchain based online digital identity allows control over personal info sharing
- A single ID issued by Govt can be used easily, and in a transparent manner for multiple

services via a single govt blockchain

- Blockchain serves as a platform where a govt is providing various services such as pensions, taxations, or benefits and a single ID to access them
- Citizens can notarize birth certificate, marriage, deeds on the blockchain tied with their digital ID as proof.
- provide privacy and controlling the use of id info

17a) Blockchain in Finance Sector:

- Save-cost

ⓐ Insurance

- BC can help to prevent fraud claims, increase speed of claim processing and enable transparency.
- BC can reduce overall cost and effort required to process claims. Claims can be automatically verified and paid via smart contracts and associated ID of the insurance policyholder.
- Smart contracts in combination with IOT and oracles can automate the entire vehicle insurance industry.

ⓑ Post trade Settlement

- Trade lifecycle - Execution, clearing and settlement.
- A single distributed ledger with appropriate smart contracts can simplify whole process and can enable buyers and sellers to talk directly to each other

- all participants on the BC can immediately see a single version of truth regarding the state.
- P2P settlement is possible which reduce complexity, cost, risk and time.
- Intermediaries can be eliminated.

(c) Financial Crime Prevention:

- Know Your Customer (KYC) and Anti Money Laundering (AML) are the key enablers for prevention of financial crime.
- Blockchain can reduce this time consuming processes by sharing a distributed ledger between all institutions that contain verified and true IDs of customers. This ledger can only be updated via consensus with the participants.
- Provide a single shared ~~file~~ view of all financial transactions in the system that are cryptographically secure, authentic and auditable

(d) Payments

All the existing payment systems are centralized and governed by traditional financial service industry rules and practices. Blockchain can bring:

- Decentralization

- faster settlement
- Better Resilience.

17 b) Oracles:

- essential component of smart contract and blockchain ecosystem.
- External data might be required to control the execution of some business logic in smart contract. In such situation, oracles can be used to provide external data to smart contracts.
- An oracle can be defined as an interface that delivers data from an external source to smart contract.
- Oracles are trusted entities that use a secure channel to transfer off-chain data to a smart contract.
- Oracles can deliver diff types of data like weather reports, real-world news, Corporate actions to data coming from an IOT device etc.

How Oracle Works:

① Smart Contract sends a request for data to an oracle.

② Request is executed and required data is requested from source.

③ The data is sent to a notary to generate

cryptographic proof (usually a digital signature)

④ data with Proof of Validity is sent to Oracle.

⑤ The requested data with its proof can be optionally saved on a decentralized storage system

⑥ data with proof of validity is sent to smart contract.