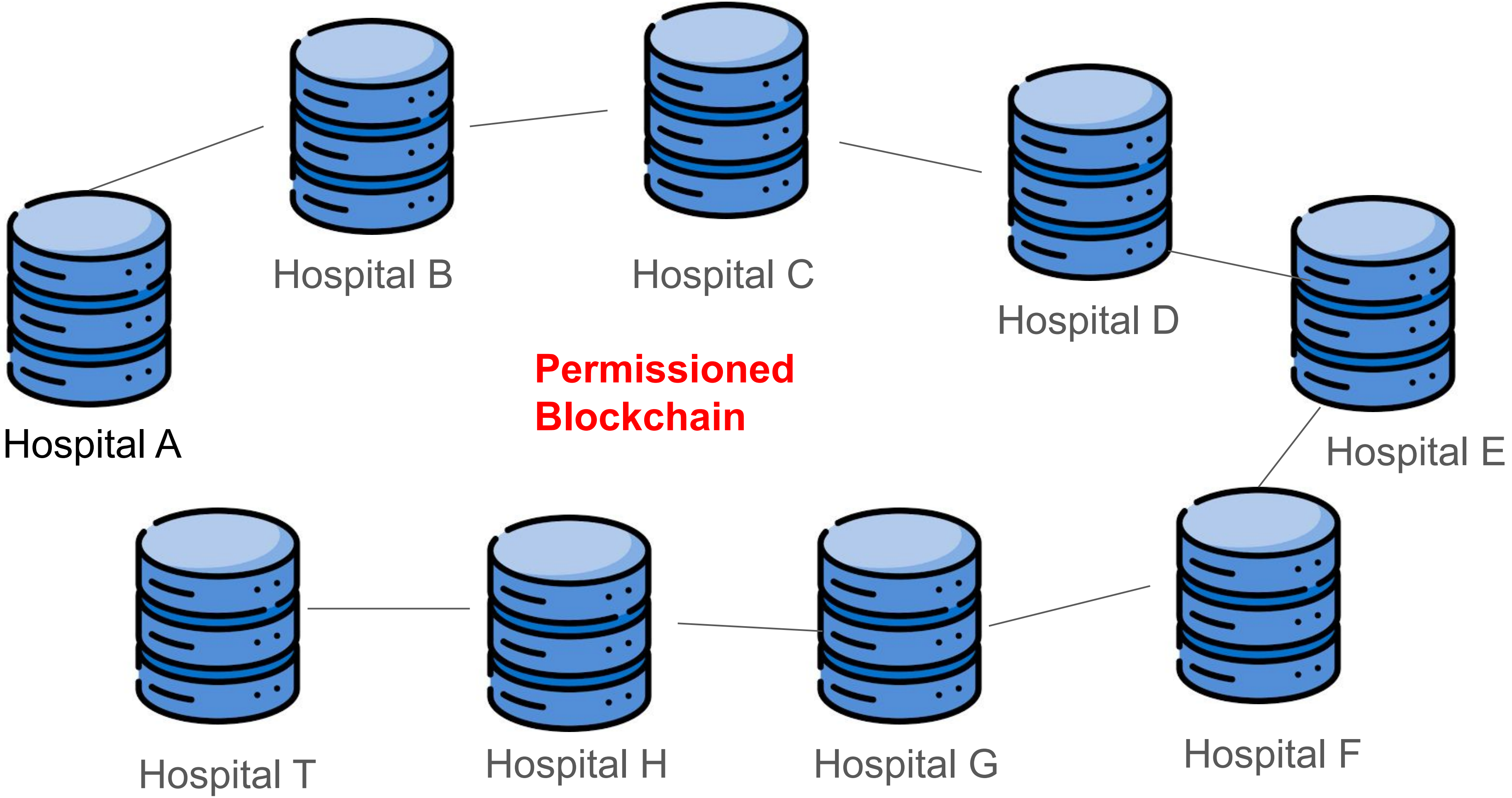
CSC 116 Different Blockchains

BFT > Private / Permissioned Blockchain

PoW > Public Blockchain, e.g., Bitcoins.
（Proof of Work）



Hospital B

Hospital C

Hospital D

Hospital A

Permissioned Blockchain

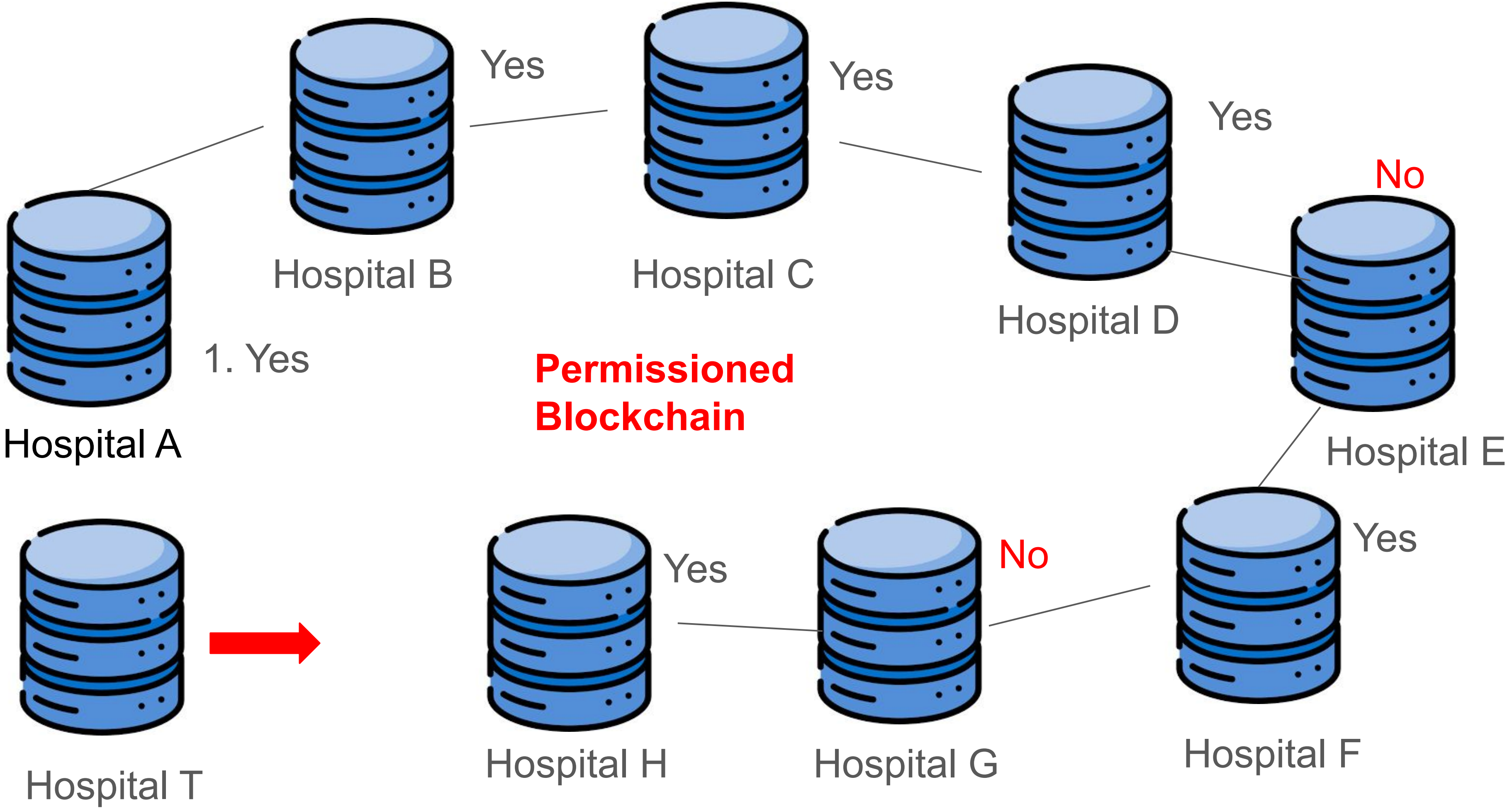
Hospital E

Hospital T

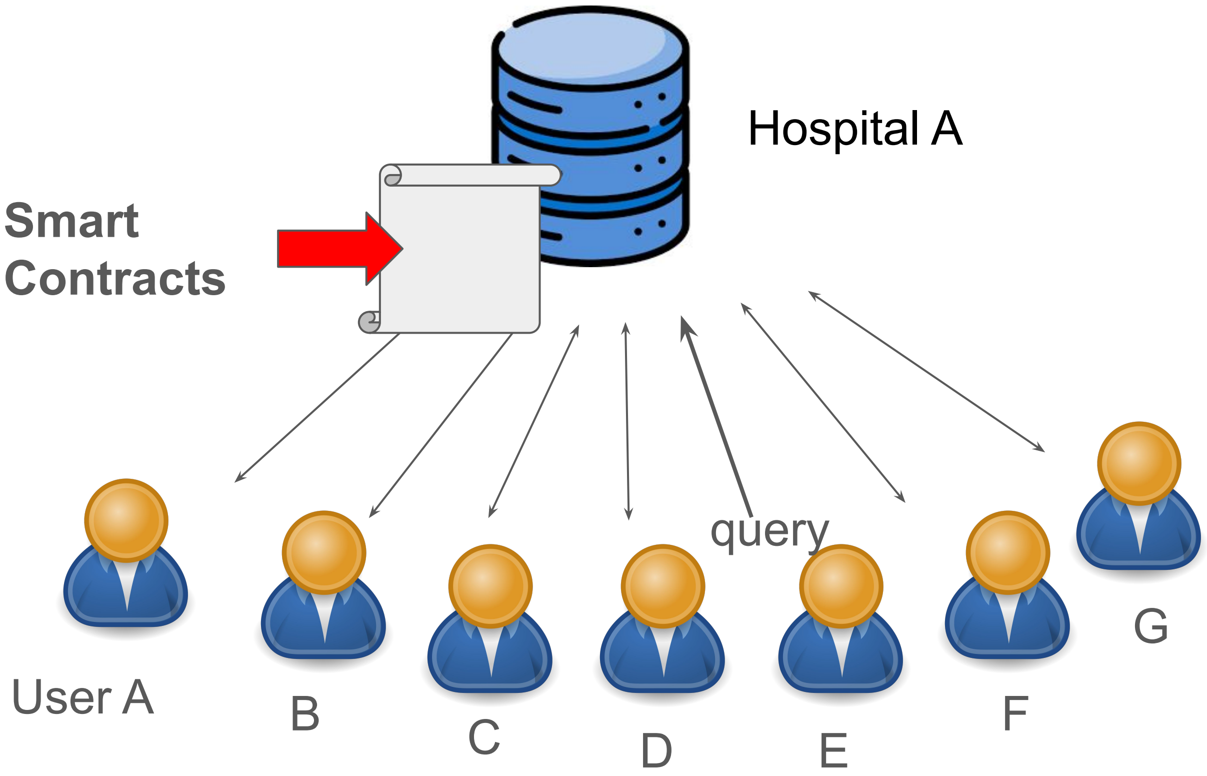
Hospital H

Hospital G

Hospital F



Permissioned Blockchain



Different users will have different access control

The contracts are codes

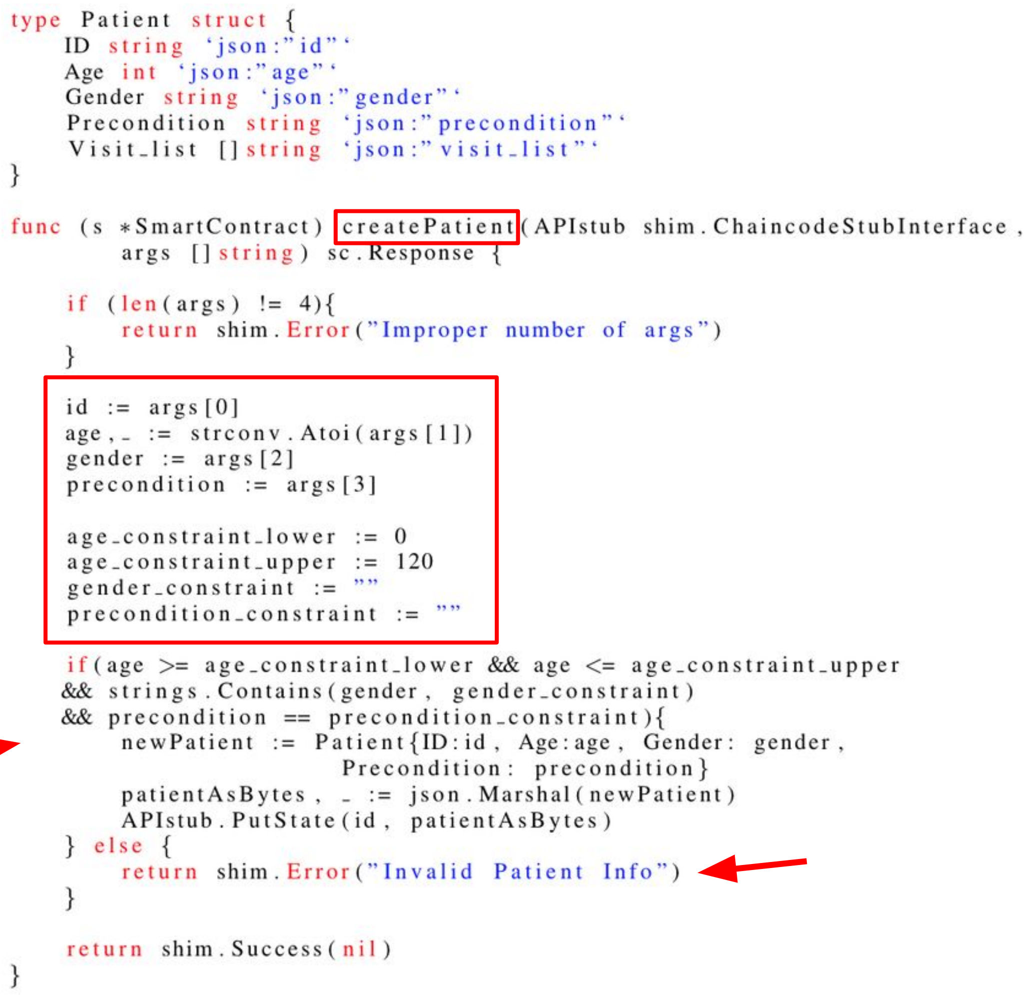
BLOCKCHAIN NETWORK
SELLER
SMART CONTRACT
BLOCKCHAIN NETWORK
BUYER

What is a Smart Contract?

- **Definition:**
  Self-executing code stored on a blockchain that automatically runs when certain conditions are met.

- **Why are they special?**
  - No middlemen.
  - Immutable (can’t be changed).
  - Transparent (everyone can see the code).

Smart Contract

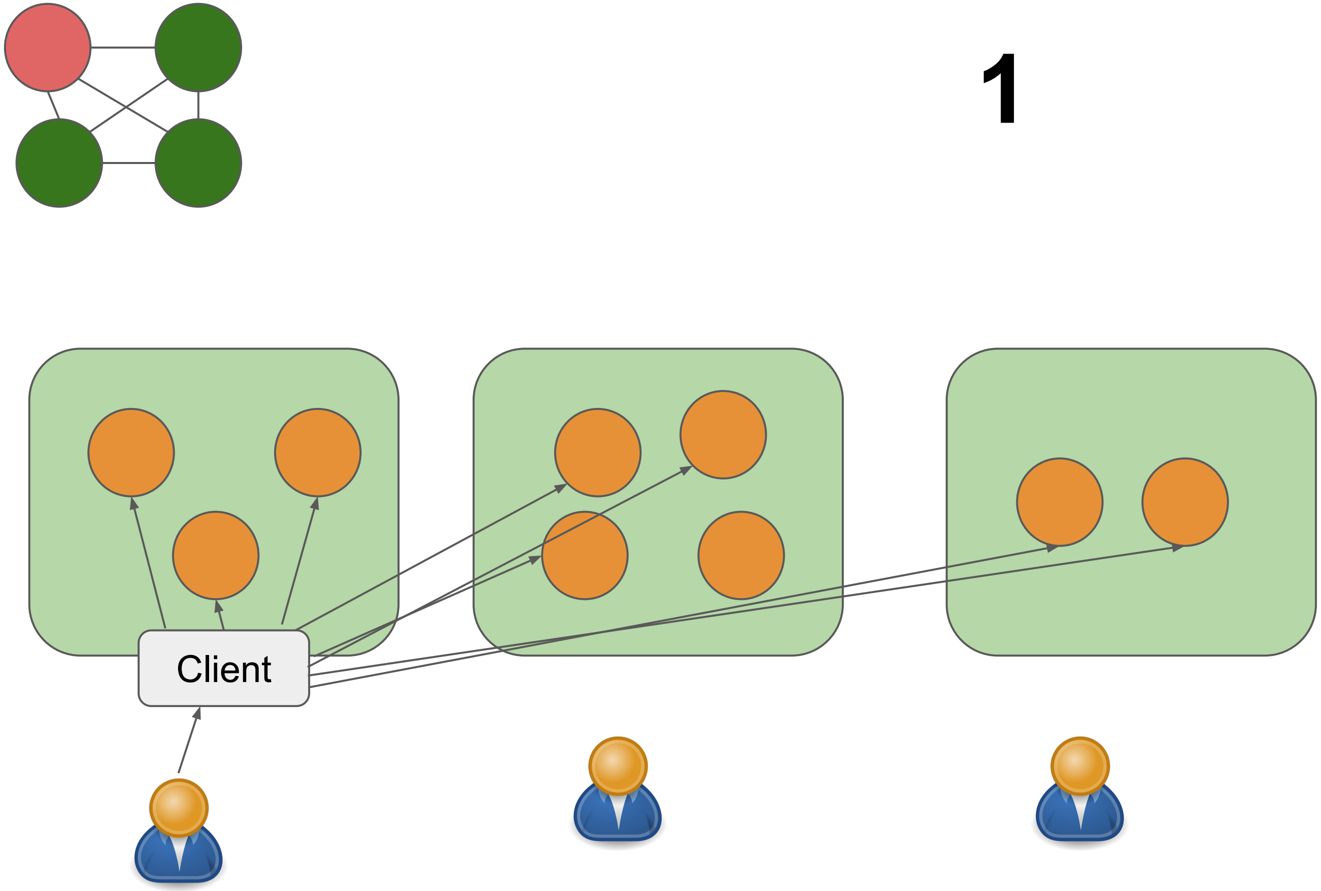


```
type Patient struct {
    ID string ‘json:”id”‘
    Age int ‘json:”age”‘
    Gender string ‘json:”gender”‘
    Precondition string ‘json:”precondition”‘
    Visit_list []string ‘json:”visit_list”‘
}

func (s *SmartContract) createPatient(APIstub shim.ChaincodeStubInterface,
        args []string) sc.Response {

    if (len(args) != 4){
        return shim.Error(”Improper number of args”)
    }

    id := args[0]
    age,_ := strconv.Atoi(args[1])
    gender := args[2]
    precondition := args[3]

    age_constraint_lower := 0
    age_constraint_upper := 120
    gender_constraint := ””
    precondition_constraint := ””

    if(age >= age_constraint_lower && age <= age_constraint_upper
    && strings.Contains(gender, gender_constraint)
    && precondition == precondition_constraint){
        newPatient := Patient{ID:id, Age:age, Gender: gender,
                     Precondition: precondition}
        patientAsBytes, _ := json.Marshal(newPatient)
        APIstub.PutState(id, patientAsBytes)
    } else {
        return shim.Error(”Invalid Patient Info”)
    }

    return shim.Success(nil)
}
```

Check the new patient infos

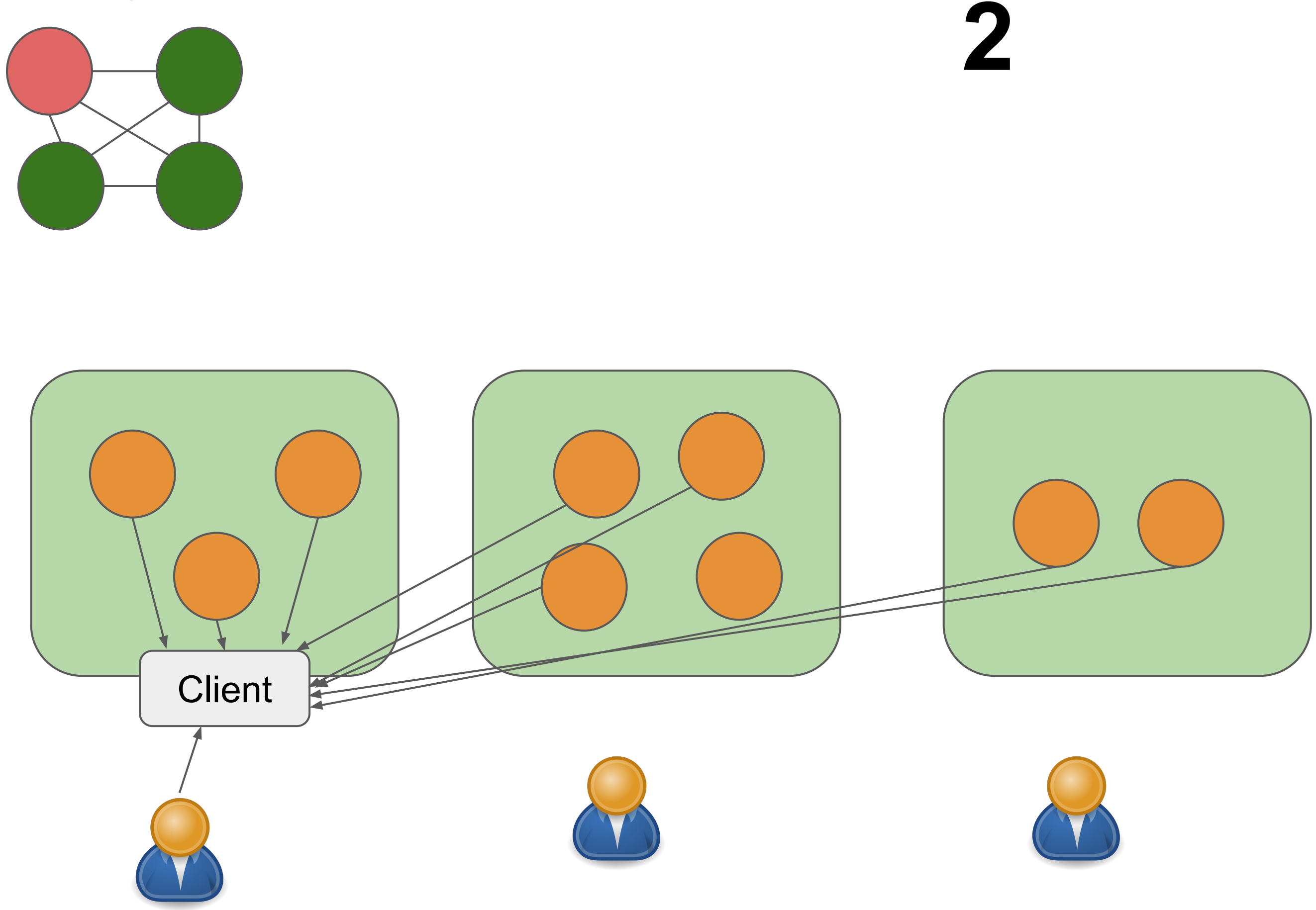
Ordering Service

1



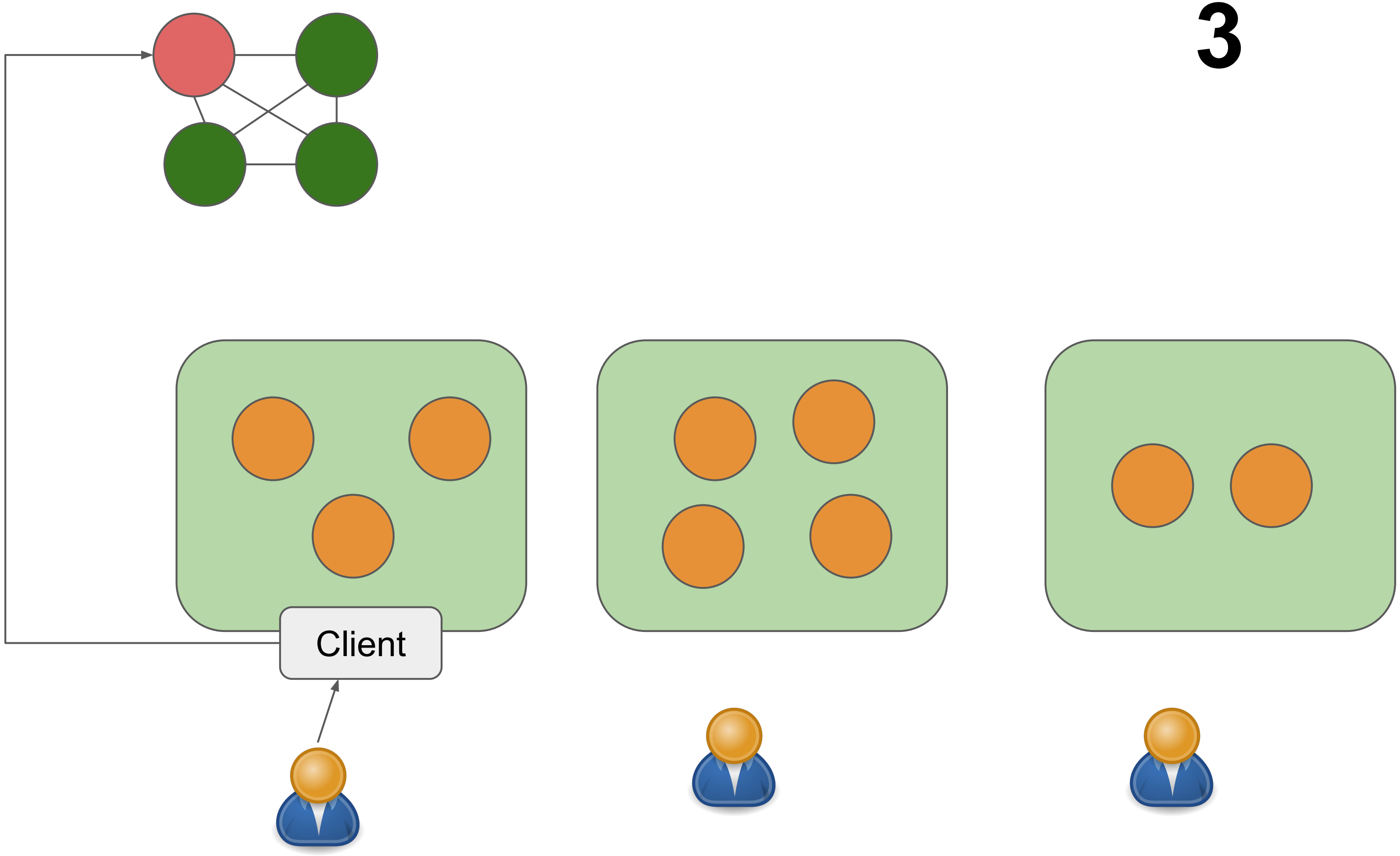
Ordering Service

2



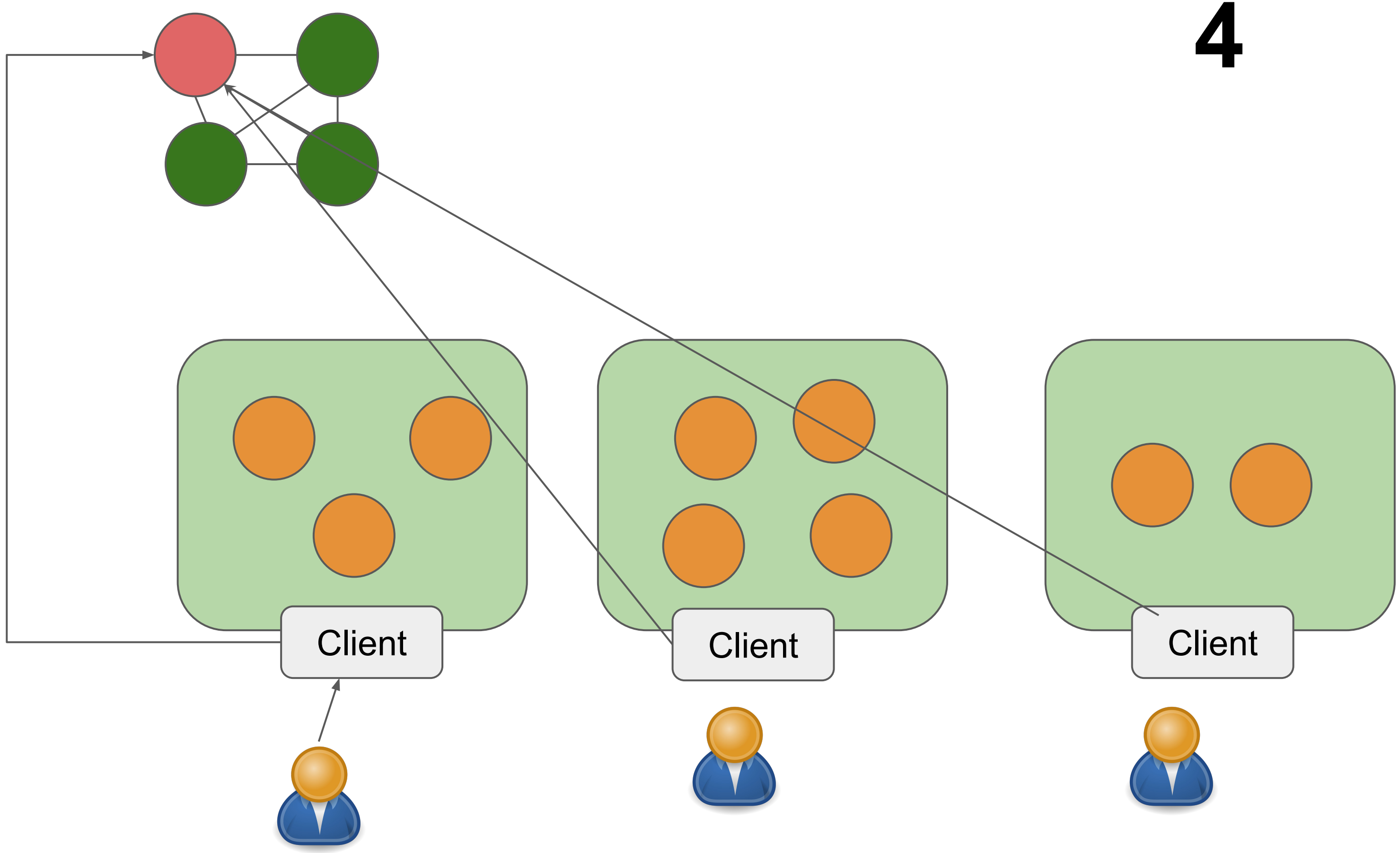
BFT Ordering Service

3

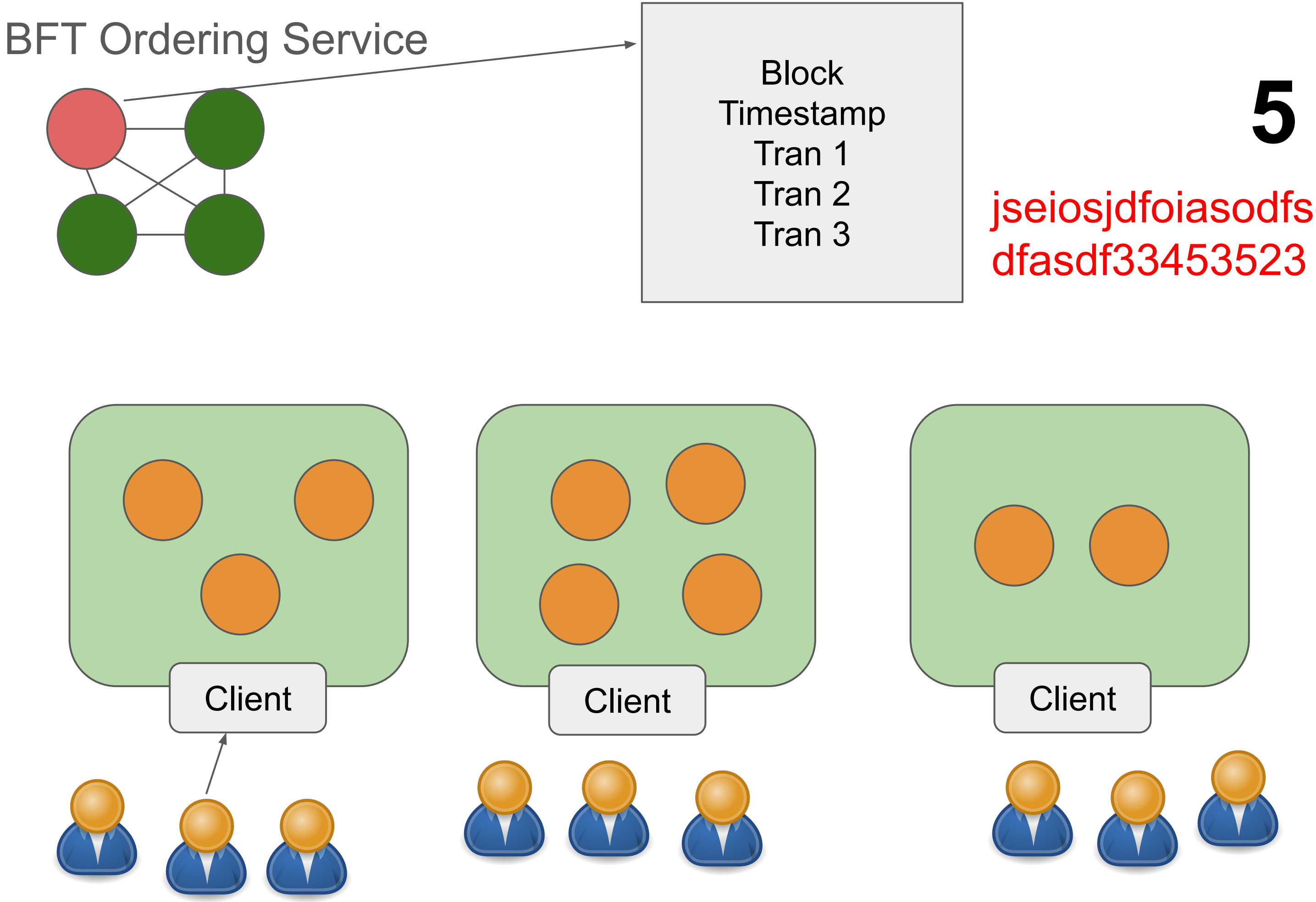


Client

BFT Ordering Service



4



BFT Ordering Service

Block
Timestamp
Tran 1
Tran 2
Tran 3

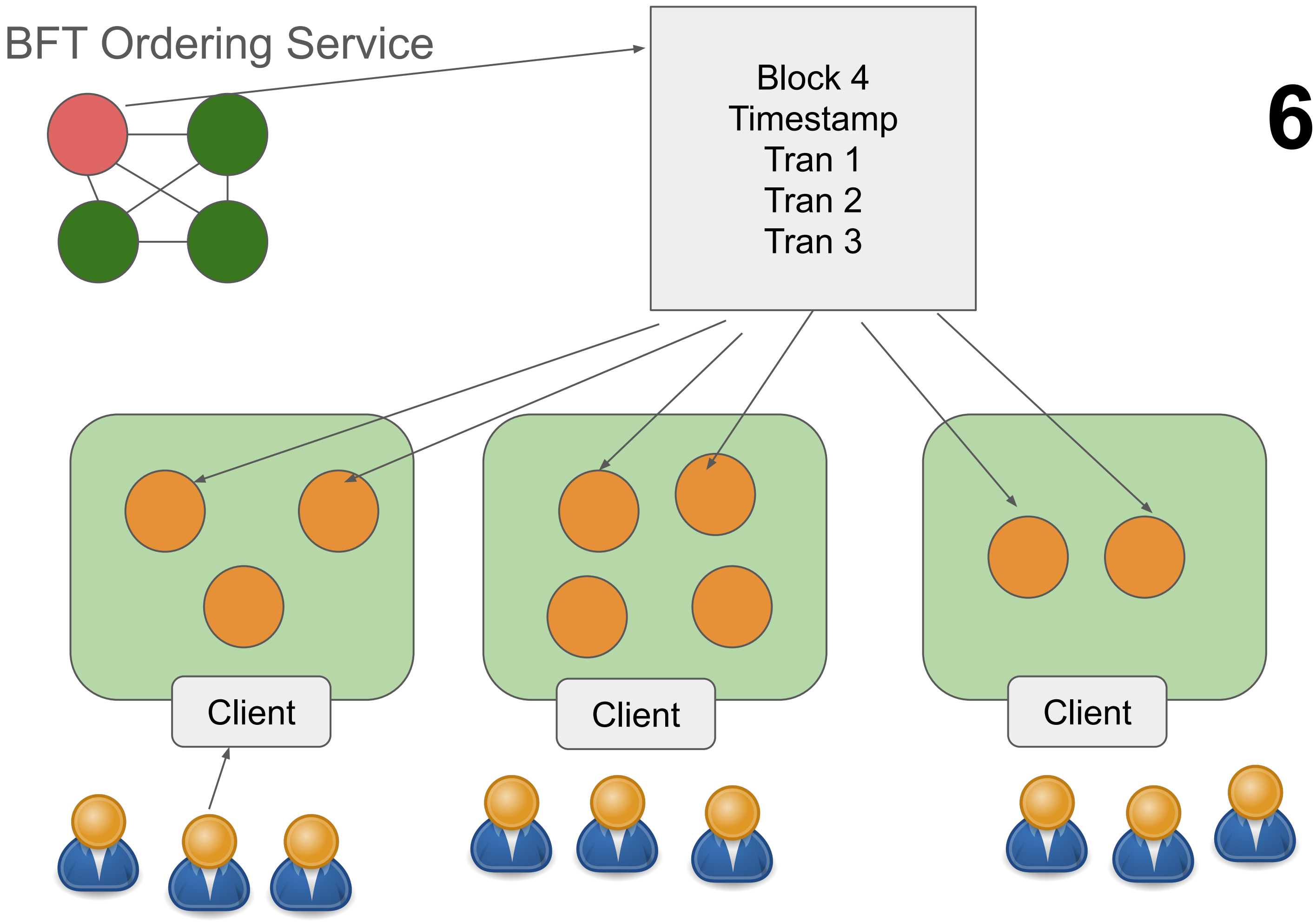
Client

Client

Client

5

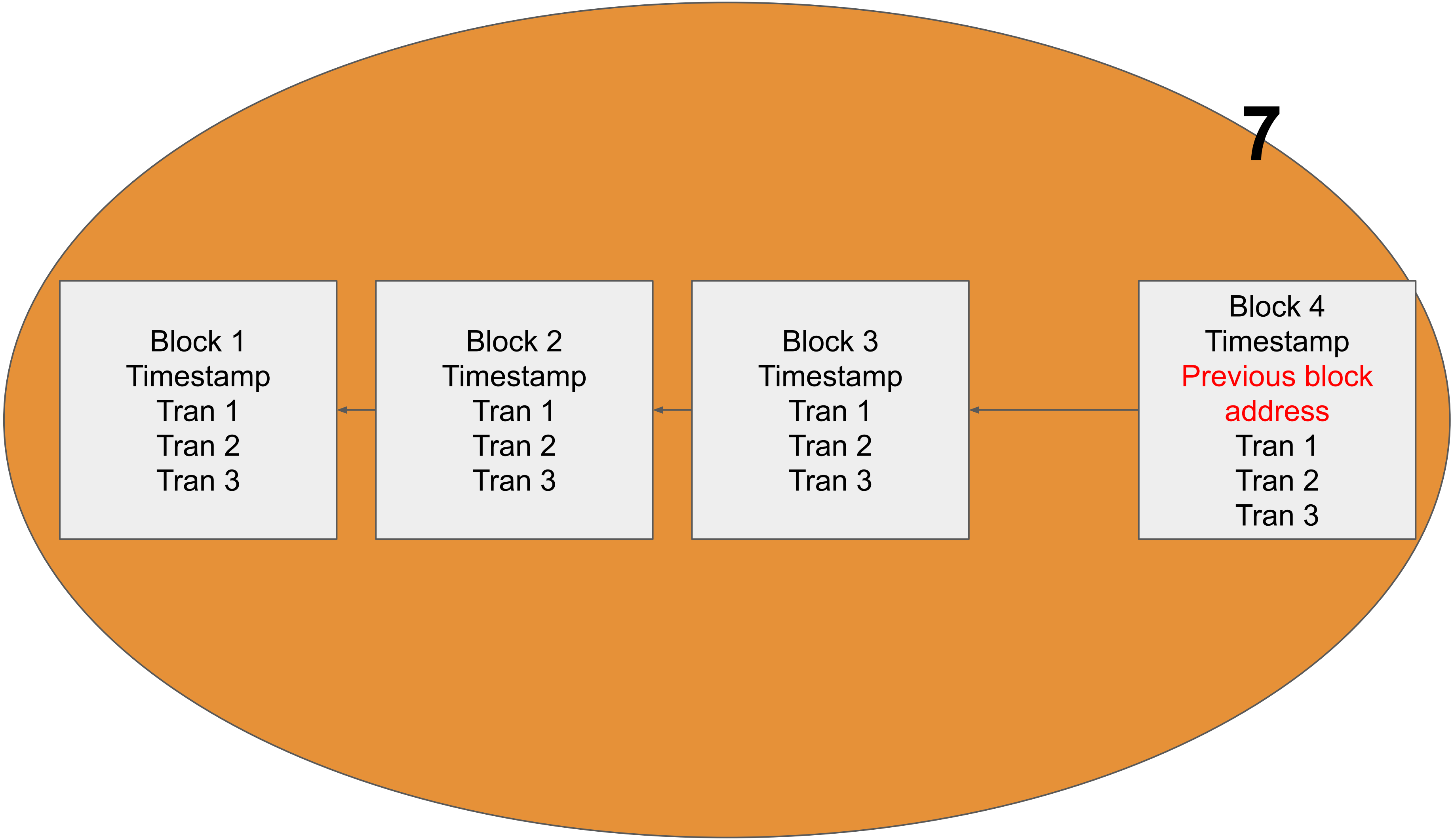
jseiosjdfoiasodfsdfjaskdfs
dfasdf33453523



BFT Ordering Service
Block 4
Timestamp
Tran 1
Tran 2
Tran 3
Client
Client
Client

6

7



Block 1
Timestamp
Tran 1
Tran 2
Tran 3

Block 2
Timestamp
Tran 1
Tran 2
Tran 3

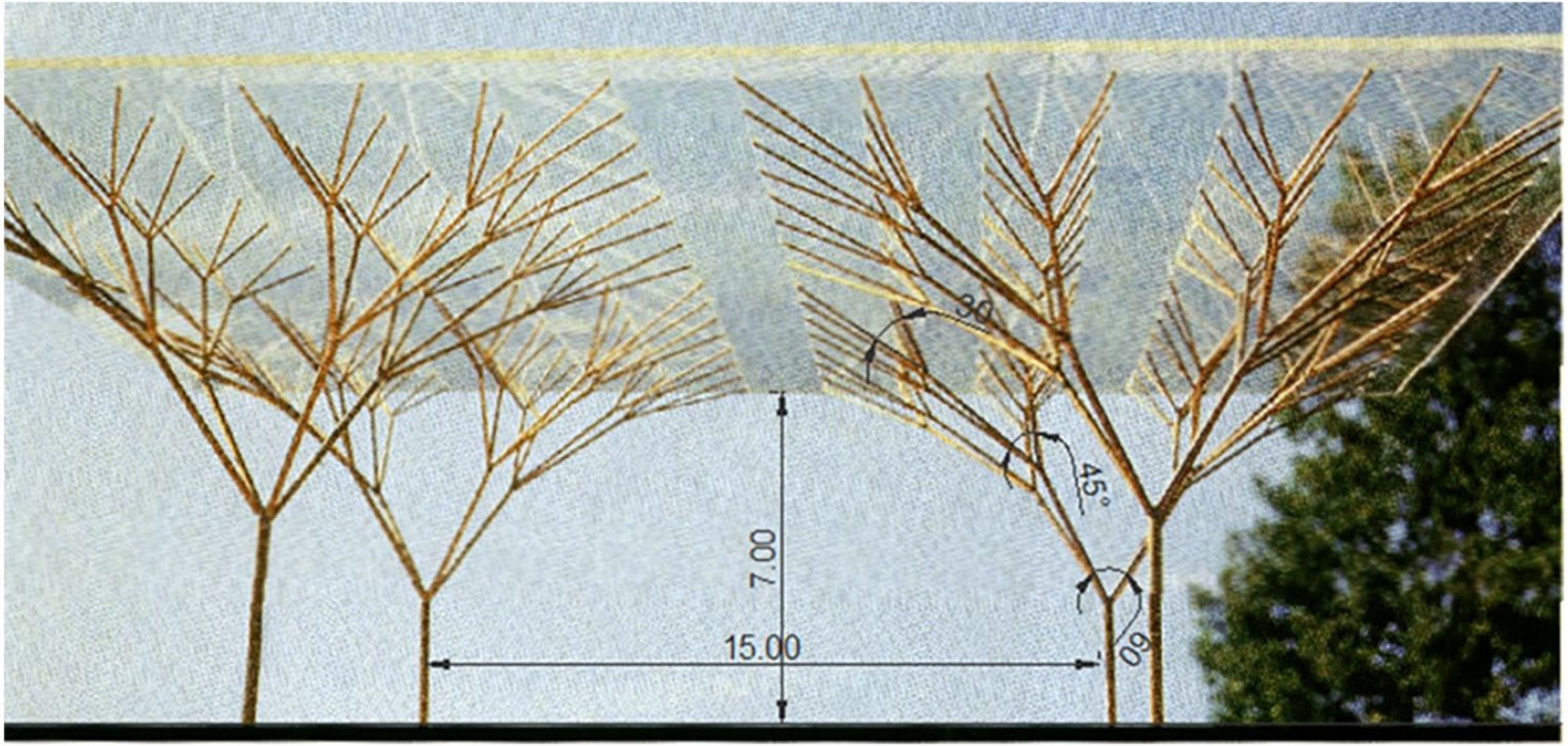
Block 3
Timestamp
Tran 1
Tran 2
Tran 3

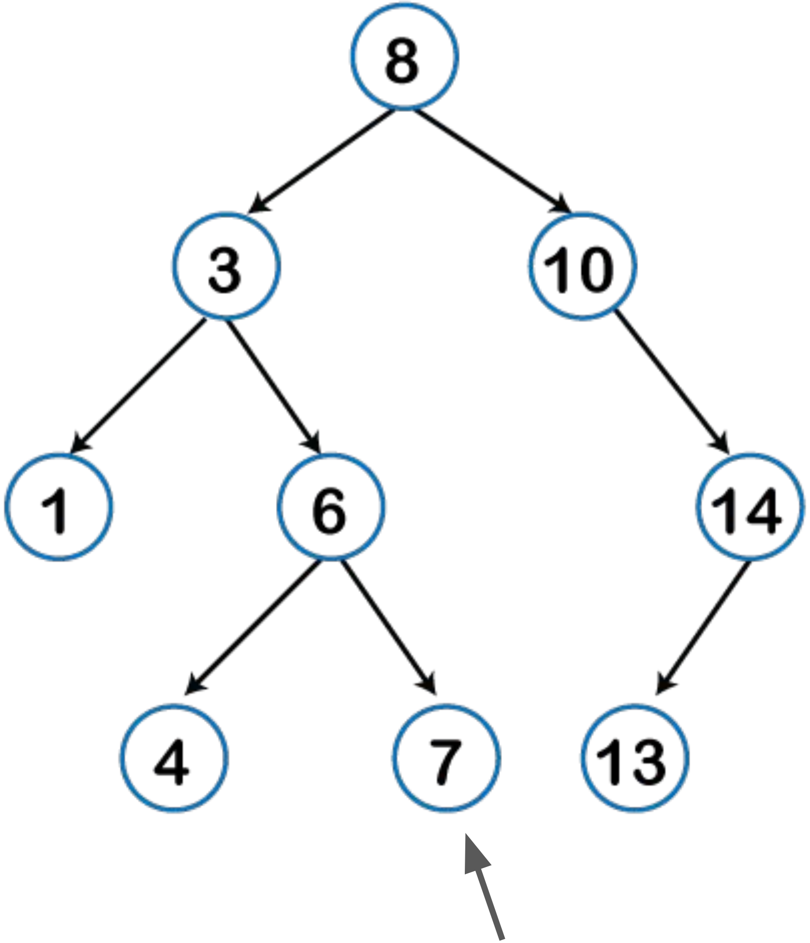
Block 4
Timestamp
Previous block address
Tran 1
Tran 2
Tran 3

Why it must be a chain?

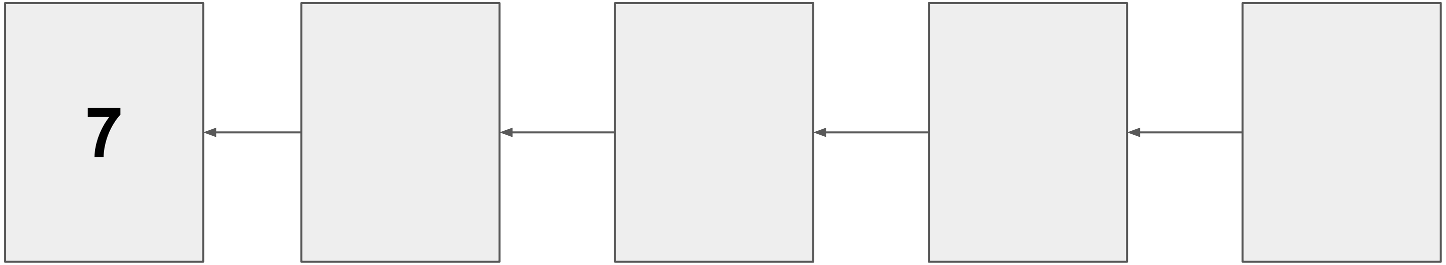
https://andersbrownworth.com/blockchain/blockchain

Bad query performance




8
3
10
1
6
14
4
7
13

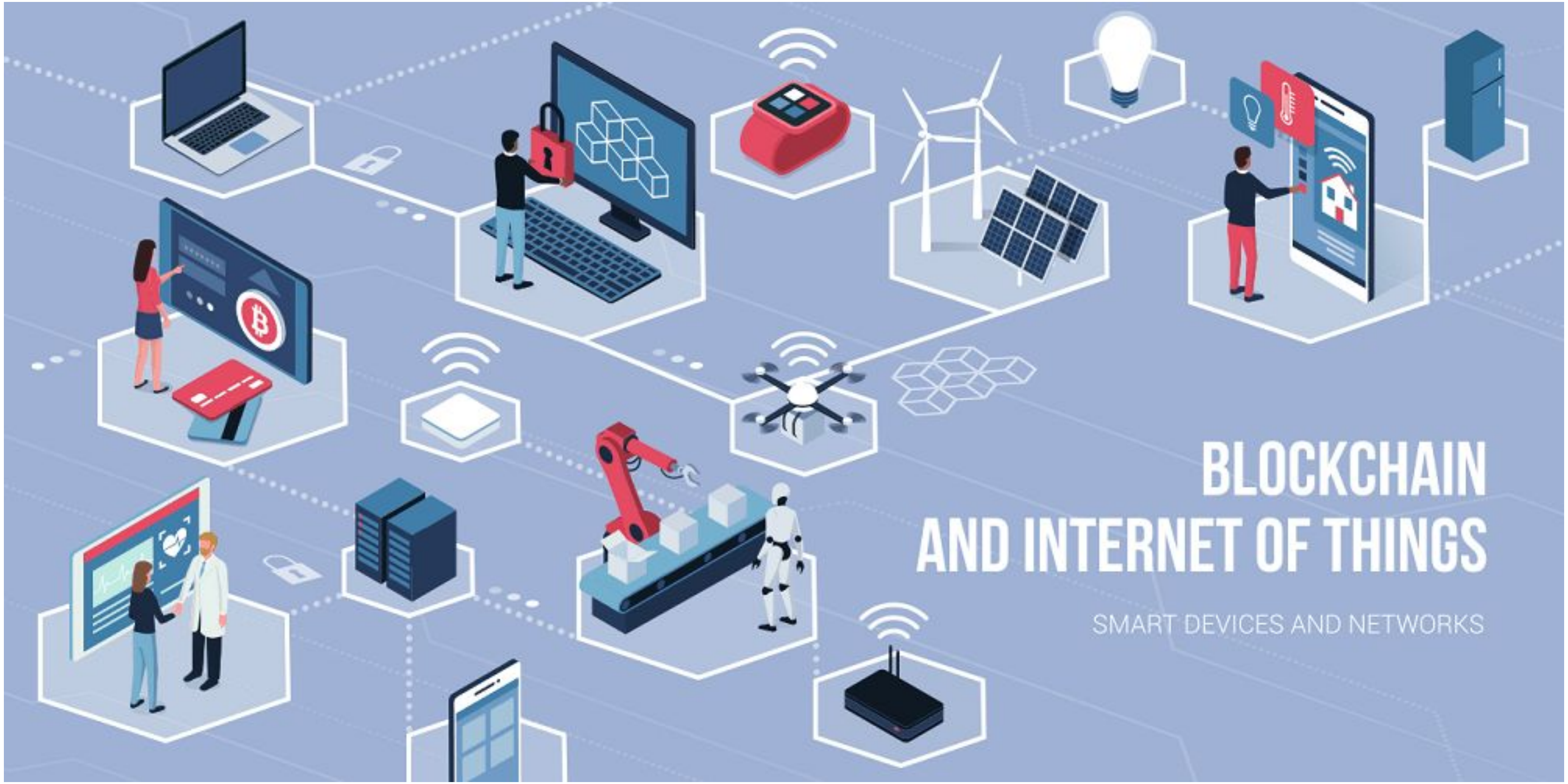
Bad query performance in Blockchain

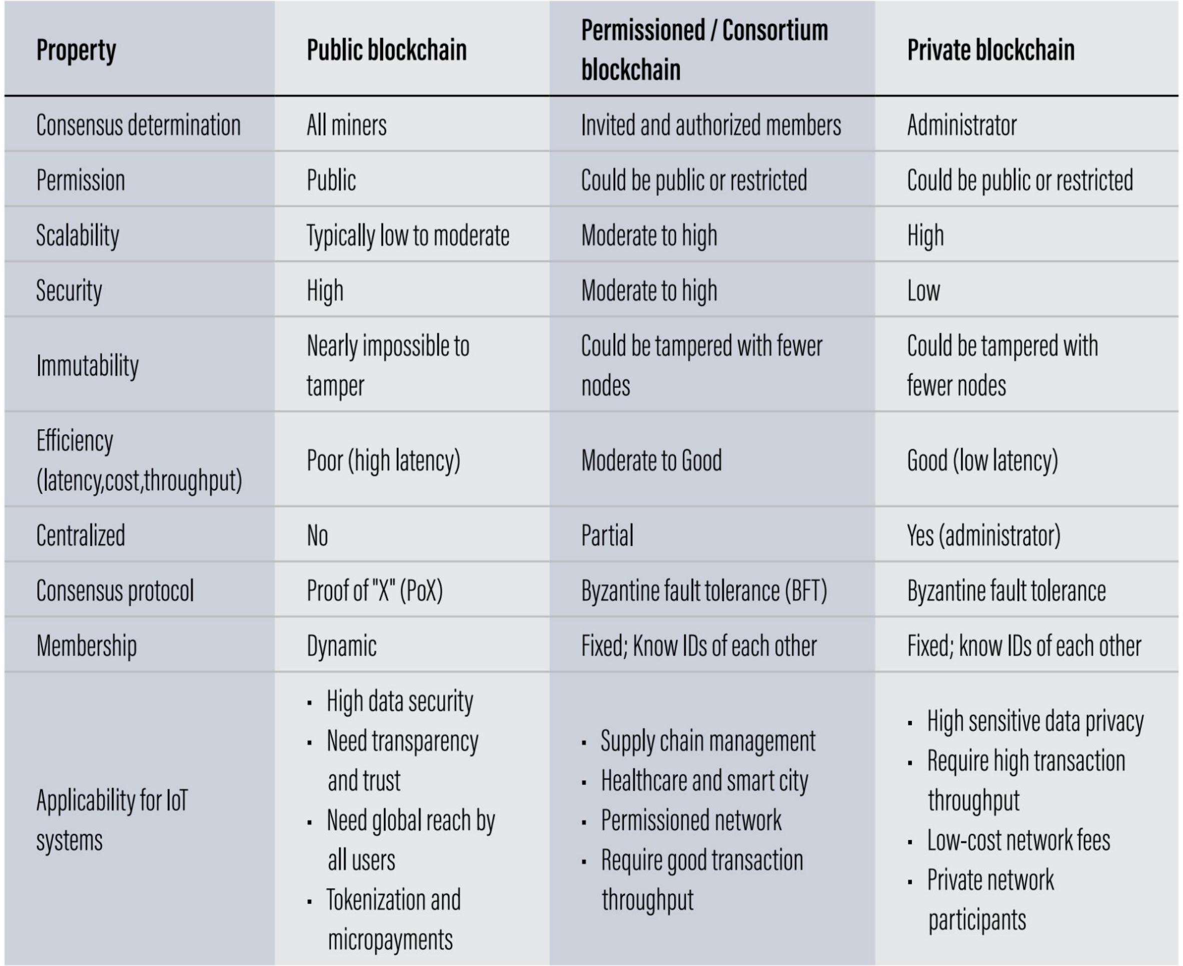


Data Quality in Permissioned Blockchain

While blockchain guarantees immutability and consistency, it does not guarantee that the data being recorded is correct or reliable.

Why?


BLOCKCHAIN
AND INTERNET OF THINGS
SMART DEVICES AND NETWORKS



Property	Public blockchain	Permissioned / Consortium blockchain	Private blockchain
Consensus determination	All miners	Invited and authorized members	Administrator
Permission	Public	Could be public or restricted	Could be public or restricted
Scalability	Typically low to moderate	Moderate to high	High
Security	High	Moderate to high	Low
Immutability	Nearly impossible to tamper	Could be tampered with fewer nodes	Could be tampered with fewer nodes
Efficiency (latency,cost,throughput)	Poor (high latency)	Moderate to Good	Good (low latency)
Centralized	No	Partial	Yes (administrator)
Consensus protocol	Proof of "X" (PoX)	Byzantine fault tolerance (BFT)	Byzantine fault tolerance
Membership	Dynamic	Fixed; Know IDs of each other	Fixed; know IDs of each other
Applicability for IoT systems	▪ High data security ▪ Need transparency and trust ▪ Need global reach by all users ▪ Tokenization and micropayments	▪ Supply chain management ▪ Healthcare and smart city ▪ Permissioned network ▪ Require good transaction throughput	▪ High sensitive data privacy ▪ Require high transaction throughput ▪ Low-cost network fees ▪ Private network participants

TABLE 1. Different types of blockchain systems.

https://ieeexplore.ieee.org/abstract/document/10552257

AWS Blockchain

https://docs.aws.amazon.com/managed-blockchain/latest/hyperledger-fabric-dev/network-components.html