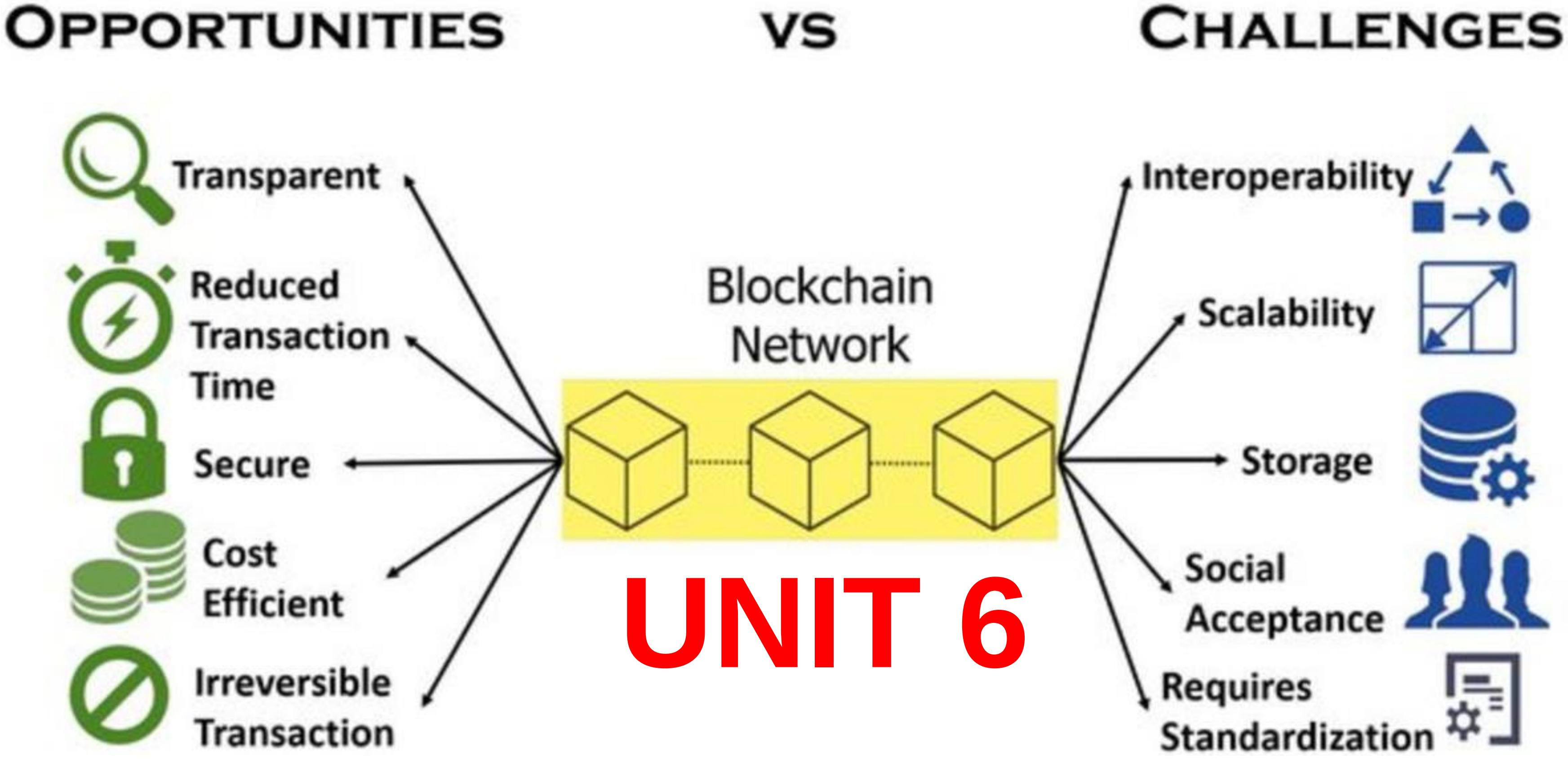


UNIT 6

BLOCKCHAIN OPENNESS

Lecturer: Ph.D Lê Quang Huy

CONTENTS

1. INTRODUCTION

1.1. BLOCKCHAIN INFRASTRUCTURE

1.2. BLOCKCHAIN TRILEMMA

1.3. BLOCKCHAIN INTEROPERABILITY

1.4. BLOCKCHAIN GOVERNANCE

1.5. DISTRIBUTED LEDGER TECHNOLOGY

1.1. BLOCKCHAIN INFRASTRUCTURE

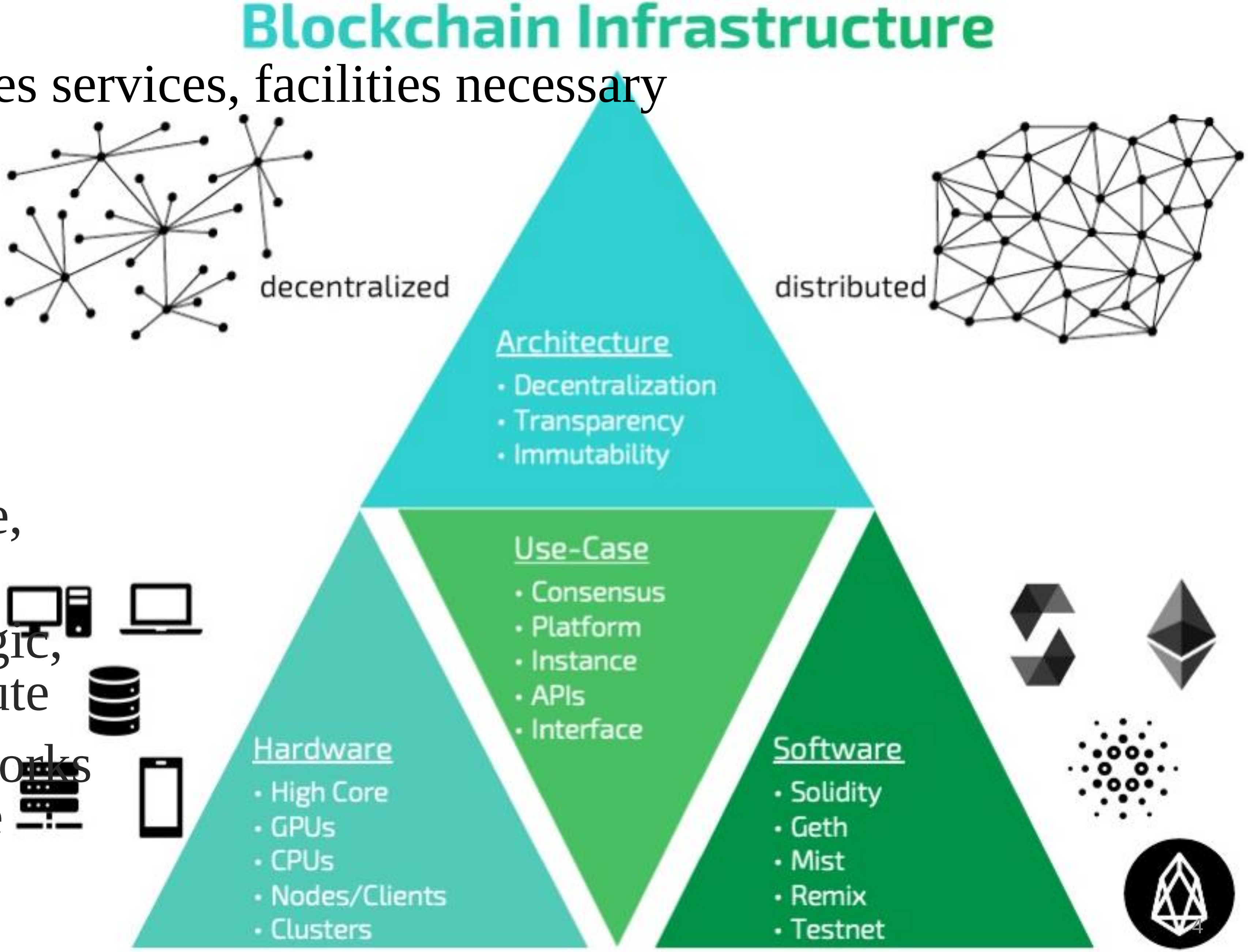
Infrastructure: encompasses services, facilities necessary

Blockchain infrastructure:

- Resources
- Underlying framework
- to function accurately

Components:

- **Storage:** token, database, file system/blobs
- **Processing:** business logic, high performance compute
- **Communications:** networks of data, of value, of state



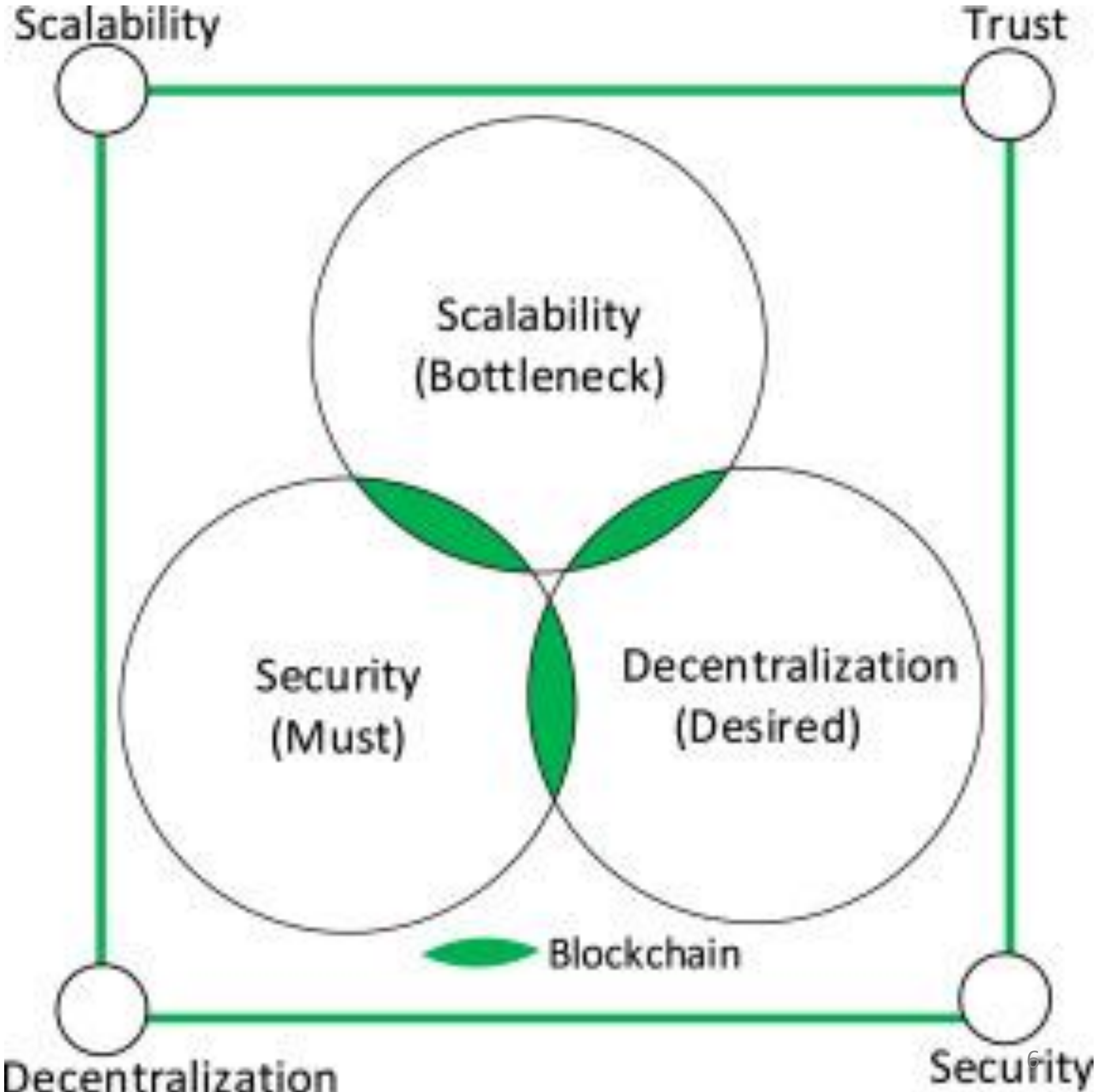
1.1. BLOCKCHAIN INFRASTRUCTURE

THE 3 ELEMENTS OF COMPUTING, *DECENTRALIZED*

STORAGE	PROCESSING	COMMUNICATIONS
TOKEN STORAGE Bitcoin, Zcash, .*	**STATEFUL BIZ LOGIC** Ethereum, Lisk, Rchain, Tezos, .. Client-side compute (JS, Swift)	**DATA** TCP/IP, HTTP, Tokenized Tor
FILE SYSTEM or BLOB IPFS/FileCoin, Eth Swarm, Storj, Sia, Tieron, LAFS	**STATELESS BIZ LOGIC** Crypto Conditions (e.g. BigchainDB). Bitshares, Eos, and all stateful biz logic	**VALUE** Interledger, Cosmos
DATABASE BigchainDB + IPDB, IOTA		
DATA MARKET Ocean Enigma, DataBroker, Datum	**HIGH PERF. COMPUTE** TrueBit, Golem, iEx.ec, Nyriad, VMs, client-side compute	**STATE** PolkaDot, Aeternity

1.2. BLOCKCHAIN TRILEMMA

- Decentralization: blockchain control central entity to group.
  power to govern blockchain.
- Security:
  - Inner: network (51%)
  - Outter: manipulate transactions to steal.
- Scalability: network grow: transaction speed and output.

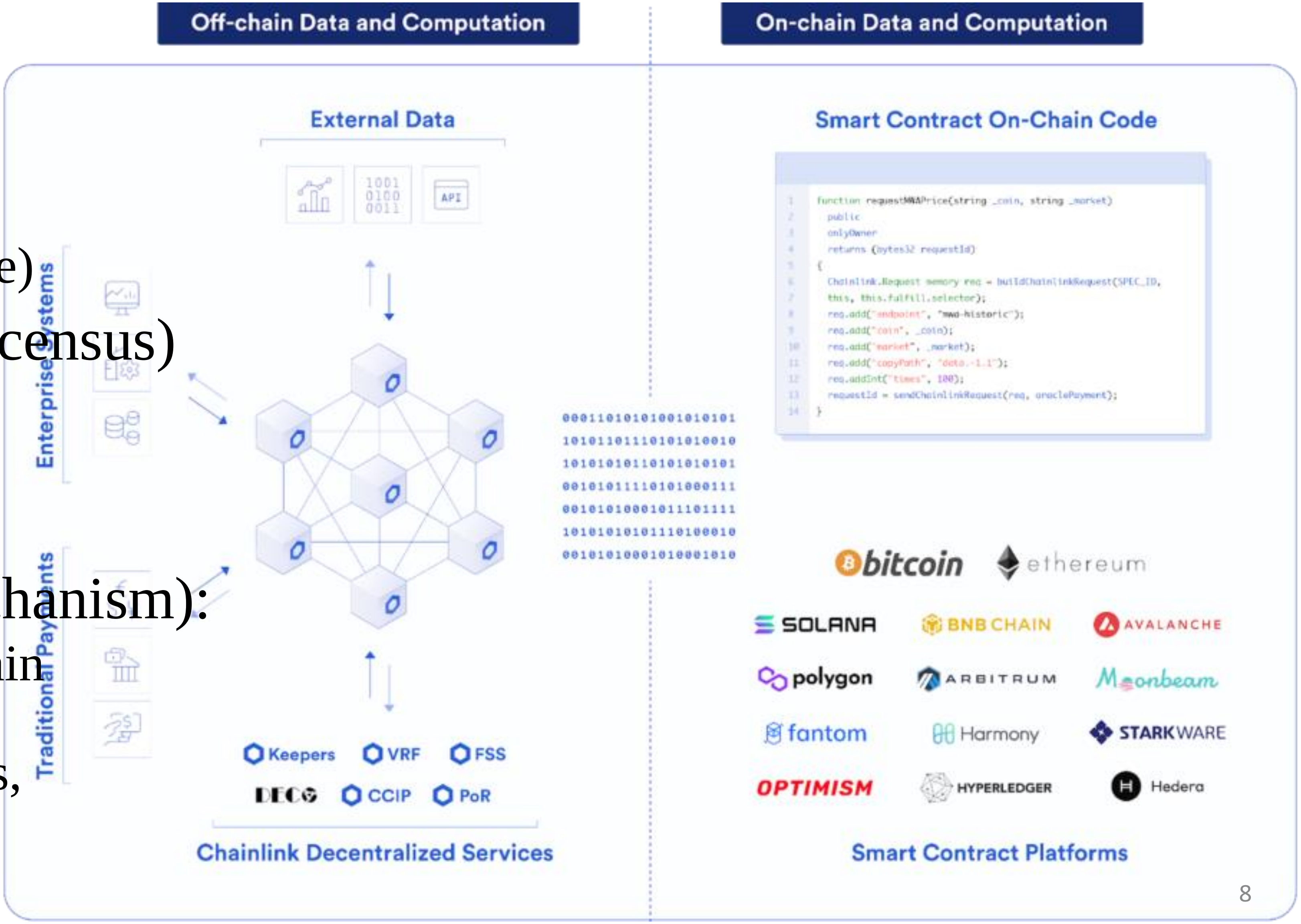


1.3. BLOCKCHAIN INTEROPERABILITY

- Interoperability: sharing and use of data/resources between systems.
- Blockchain Interoperability: ability to exchange data with external systems.
  - Between different blockchains: Cross Chain…
  - Between other systems: Oracle…. .
- Extend blockchain capabilities:
  - combining with off-chain systems

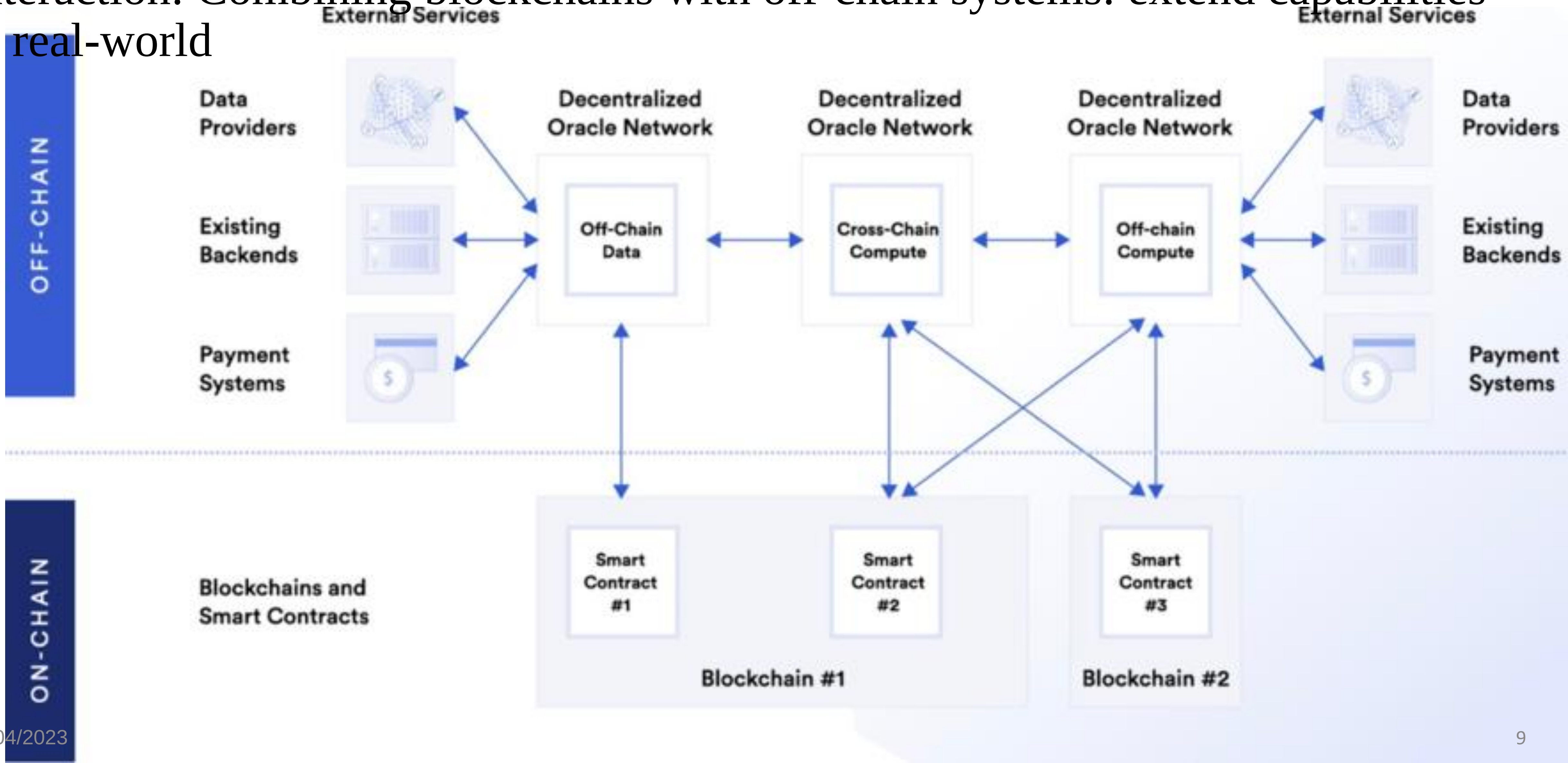
1.3. BLOCKCHAIN INTEROPERABILITY

- Onchain data:
  - Accounts, Balances
    Smart contracts, Token
- Offchain data:
  - external resouses
  - Real-world (weather, finance)
- Onchain computation: (concensus)
  - Verifying ownership
  - Executing smart contracts
  - Adding new blocks
- Offchain computation (mechanism):
  - takes place outside blockchain
  - verifiable randomness,
    transaction ordering services,
    smart contract automation.



1.3. BLOCKCHAIN INTEROPERABILITY

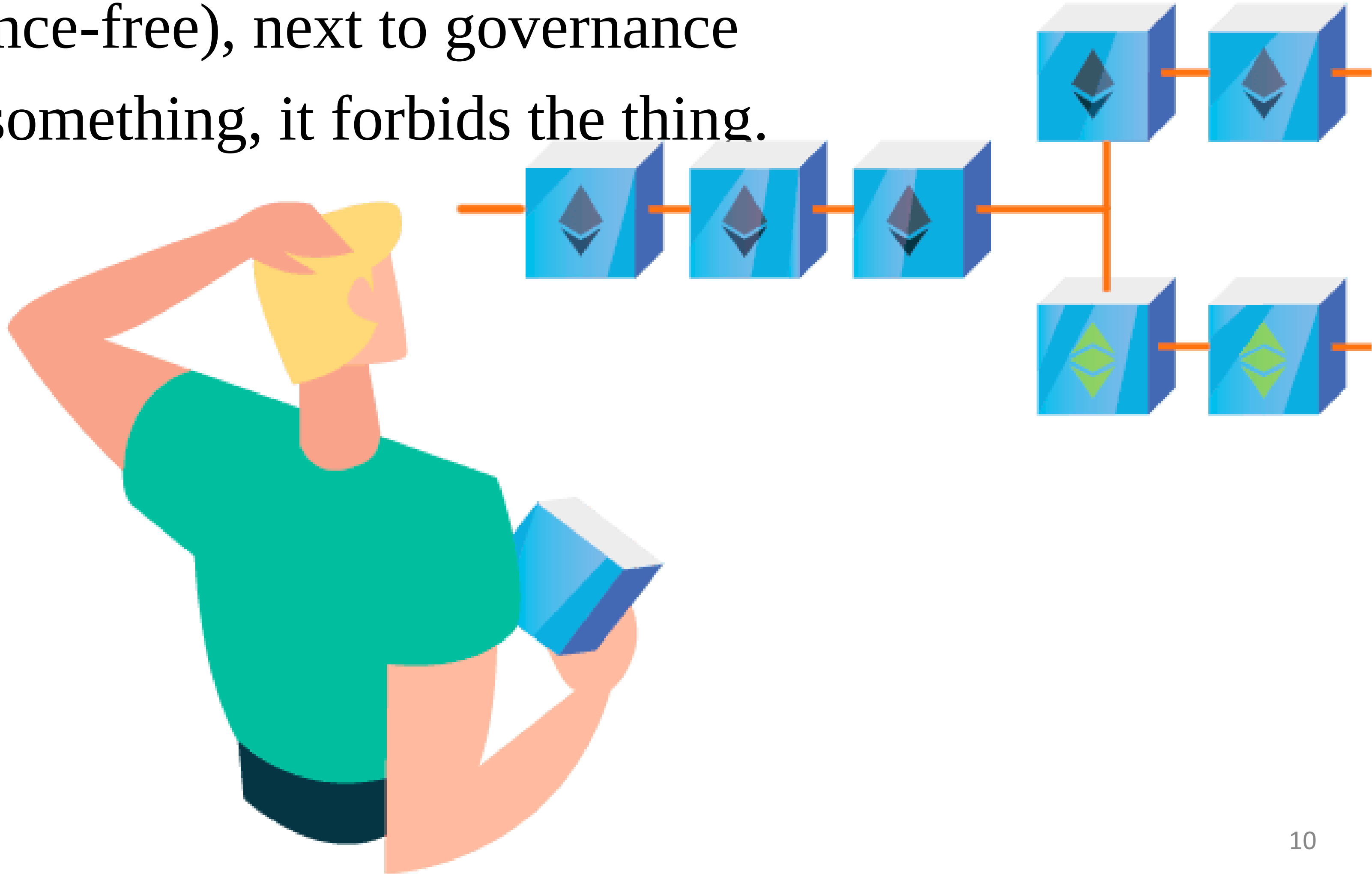
Interaction: Combining blockchains with off-chain systems: extend capabilities to real-world



1.4. BLOCKCHAIN GOVERNANCE

The need for blockchain governance:

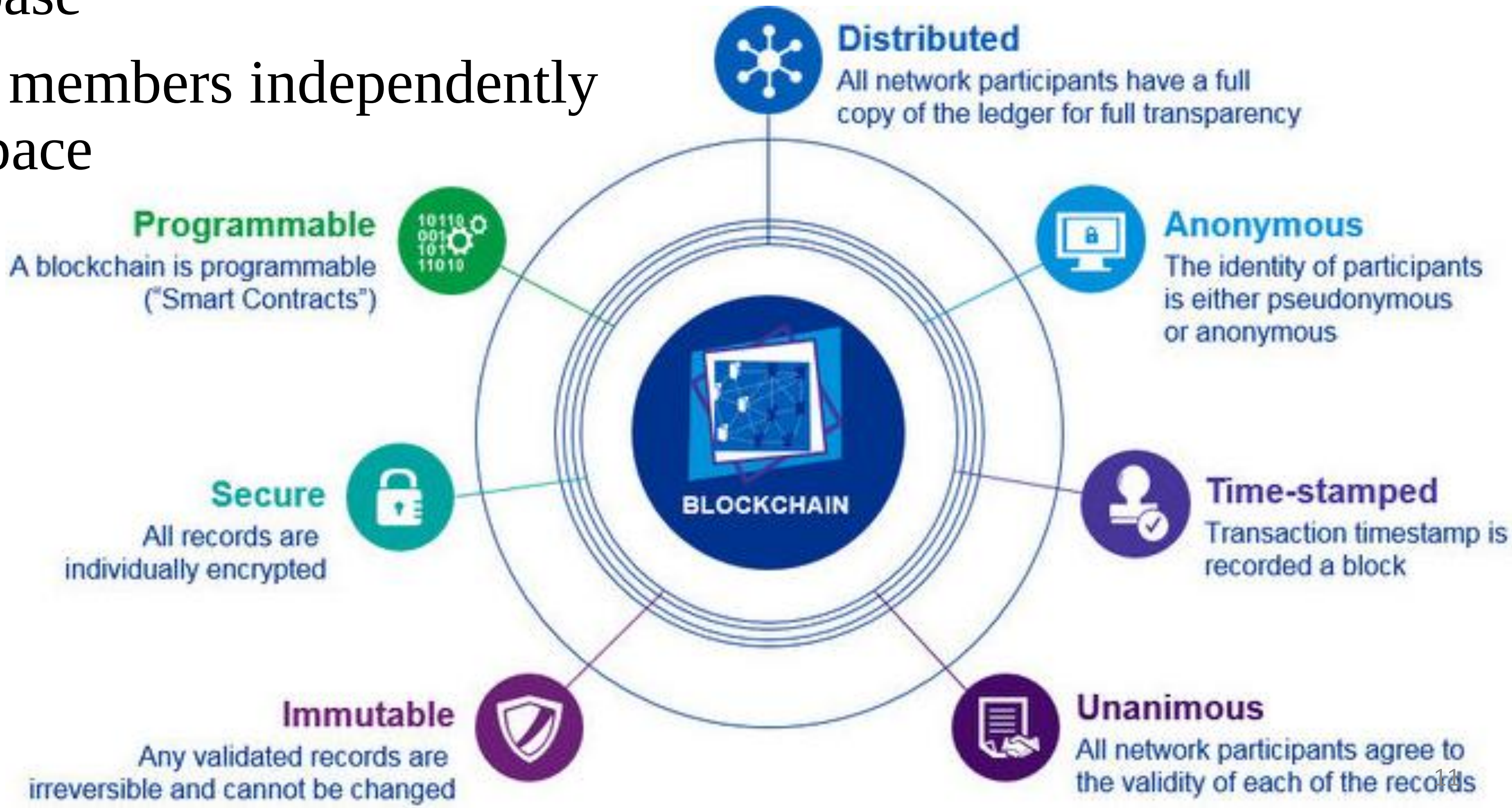
- All global things we use are under governance.
- Earlier blockchain (governance-free), next to governance
- government cannot control something, it forbids the thing.



1.5. DISTRIBUTED LEDGER TECHNOLOGY

Distributed ledger techonology:

- digital system that records transactions
- form of digital database
- updated and held by members independently in a large network space



2. BLOCKCHAIN STORAGE

2.1. BLOCKCHAIN DATA

2.2. ON-CHAIN STORAGE

2.3. OFF-CHAIN STORAGE

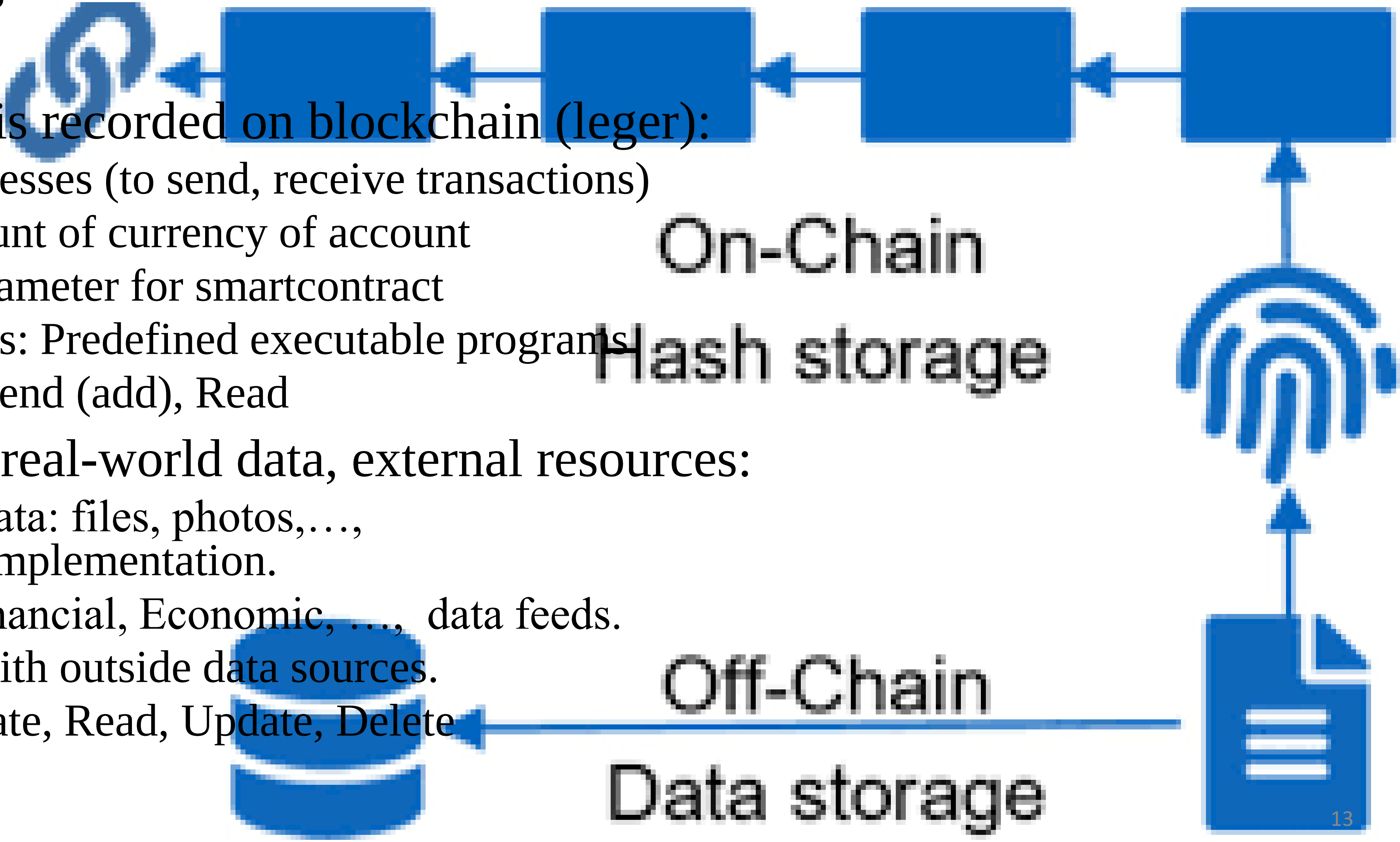
2.4. DATA STORAGE

2.1. BLOCKCHAIN DATA

Blockchain storage: store blockchain data.

Blockchain Data:

- On-chain Data: is recorded on blockchain (leger):
  - Accounts: addresses (to send, receive transactions)
  - Balances: amount of currency of account
  - Data: input parameter for smartcontract
  - Smart Contracts: Predefined executable programs
  - Activities: Append (add), Read
- Off-chain Data: real-world data, external resources:
  - Unstructured data: files, photos,…, in blockchain implementation.
  - Other Data: Financial, Economic, …, data feeds.
  - API: interact with outside data sources.
  - Activities: Create, Read, Update, Delete



2.2. ON-CHAIN STORAGE

On-chain storage problems:

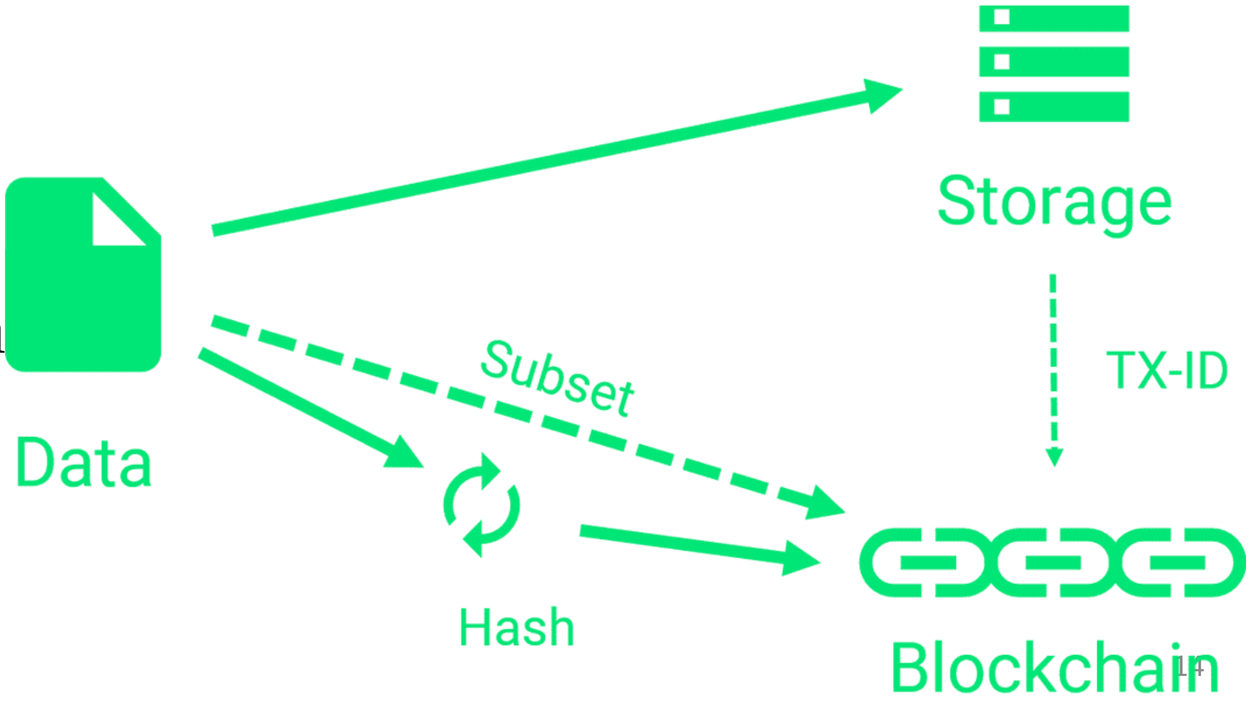
- Cost of Storing Data
- Data size: transaction/block
- Storing Sensible Information
- Slow time actions

Store large Datasets:

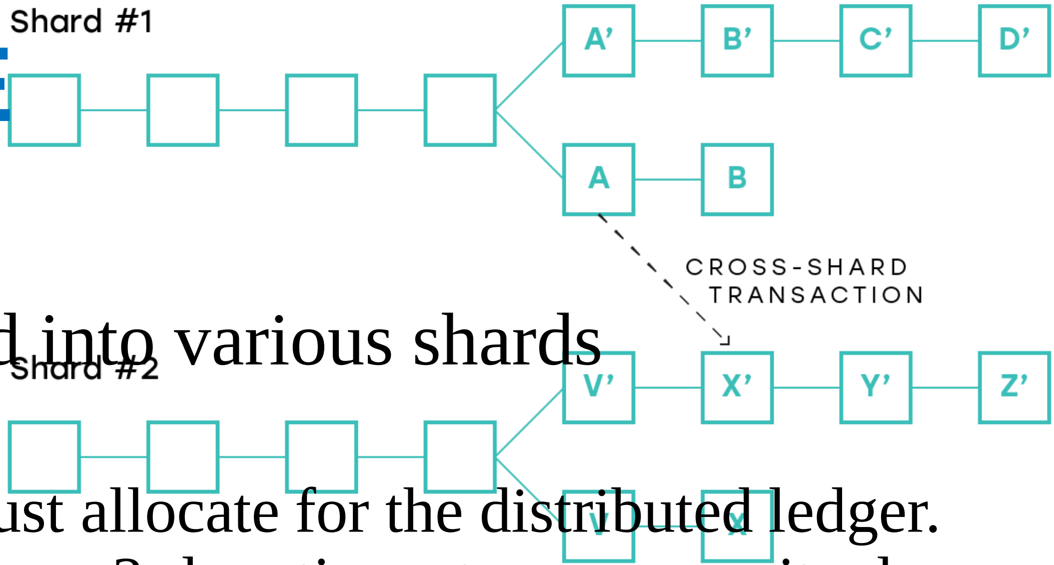
- Storing Hashes of dataset
- Storing a Subset of the Data

On-chain data

- Historical data: all transactions, block data
- Global state: snapshot of all data that smart contracts can read or write

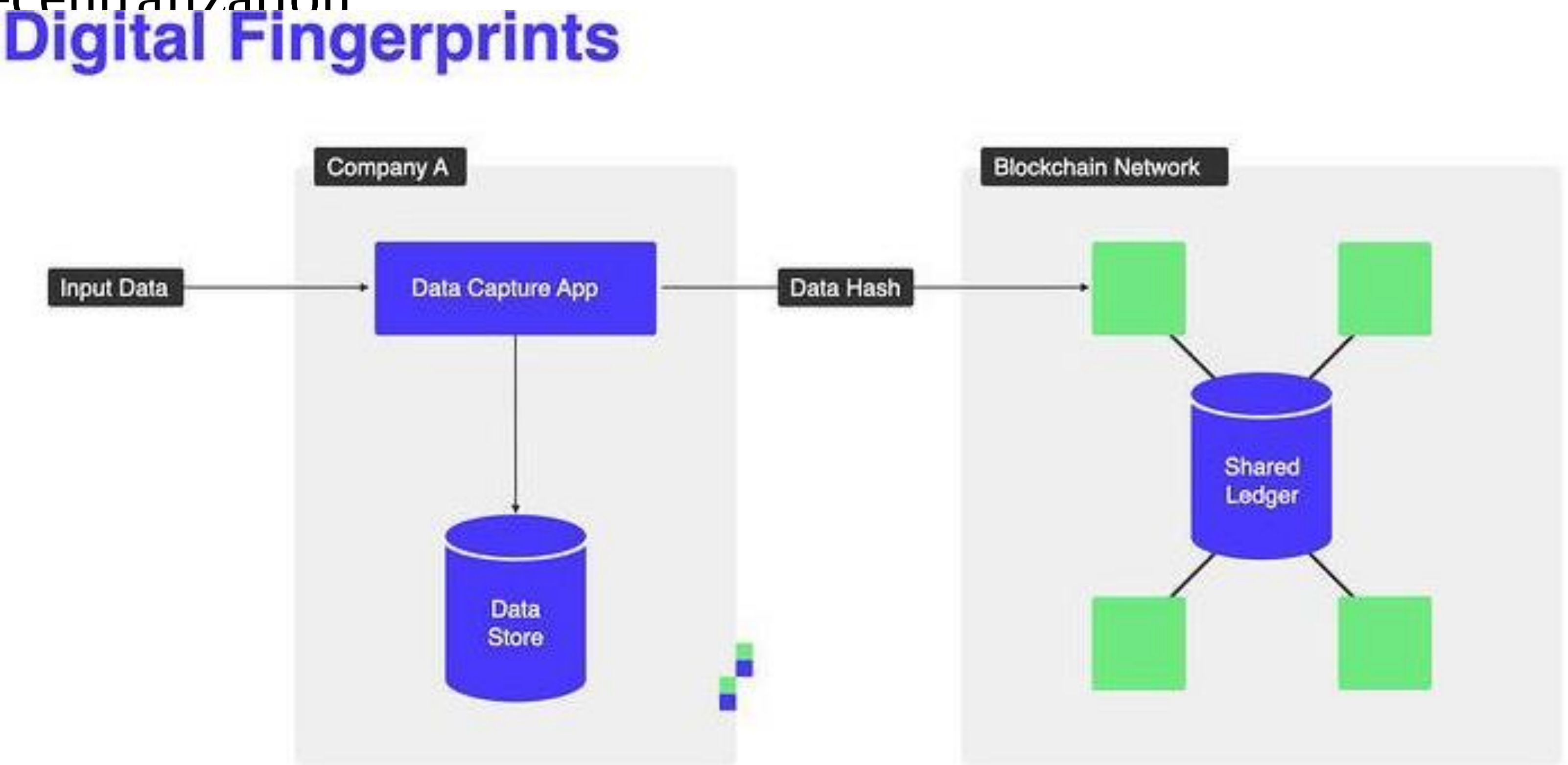


2.2. ON-CHAIN STORAGE



On-chain storage solutions:

- Sharding: partitioning the blockchain workload into various shards
  - Dedicated nodes focusing on unique data types.
  - Reduces the amount of storage space each node must allocate for the distributed ledger. increases on-chain storage capacity without relying on 3rd parties. storage capacity does not come at the expense of decentralization
- Pruning: removing older or
  - Eliminating older transaction
  - Enable more people to run nc



2.3. OFF-CHAIN STORAGE

Off-chain storage solution:

- Peer-to-peer File Systems
  - Like Interplanetary File System (IPFS): breaking up files into shards and storing them in multiple instances on the computers
  - Files are downloaded / distributed quickly;
  - Data is address dependent.
  - File (static data) is downloaded only if the user is online.
  - can access the file only by knowing the name/path.
- Decentralized Cloud Storages:
  - like Dropbox: Swarm, Storj or Sia.
  - do not need to be online to share information
  - Storages are stable, fast and have huge capacities.
  - only suitable for static data and do not support searching by content.
- Distributed Databases:
  - need to store large amounts of structured information and search for content by request.
  - Like: NoSQL. RethinkDB, Apache Cassandra, MongoDB.

2.4. DATA STORAGE

Data storage:

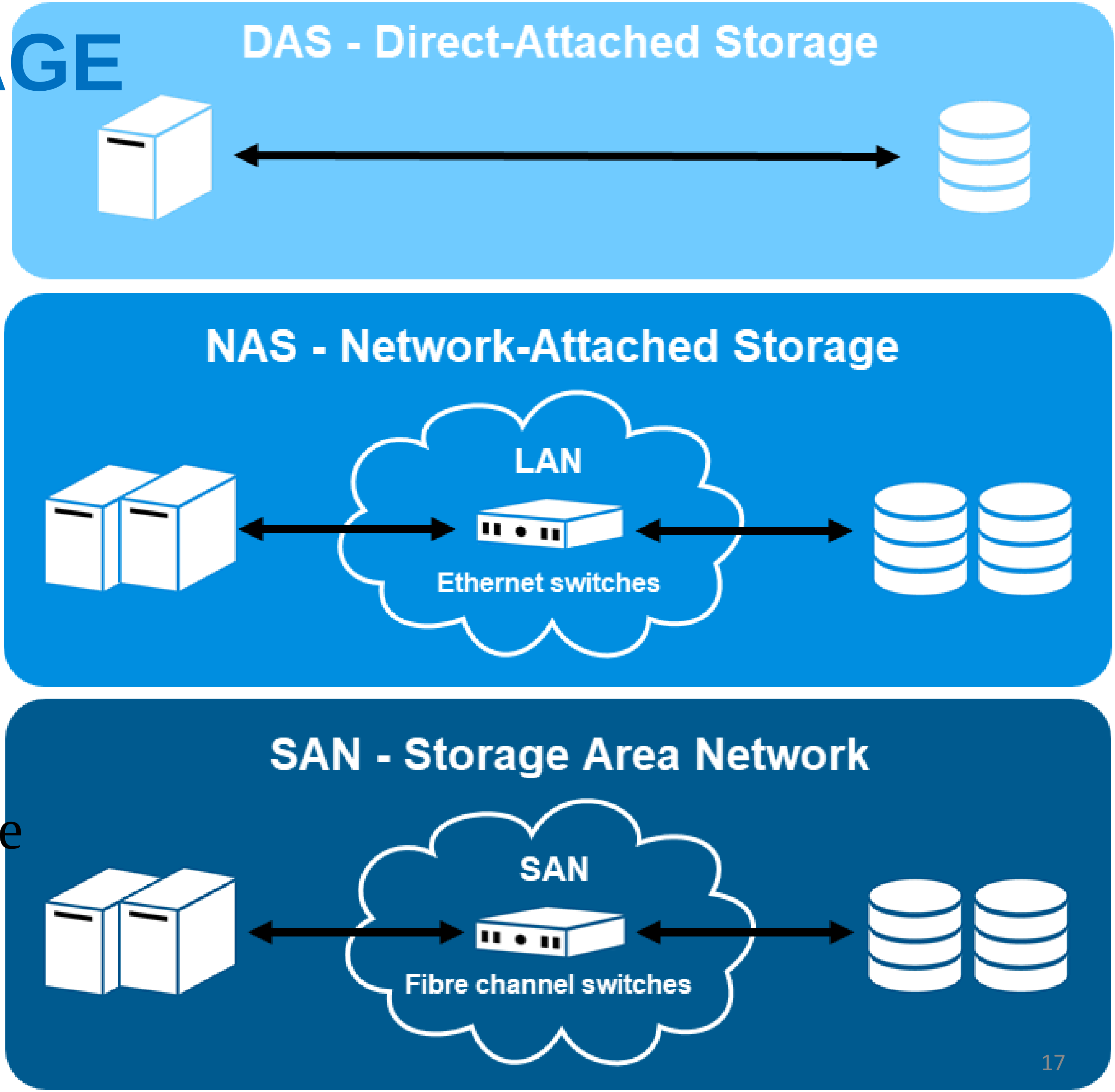
- save digital information

Challenges:

- Large data volumes
- Security
- Data integrity

Types of Data Storage

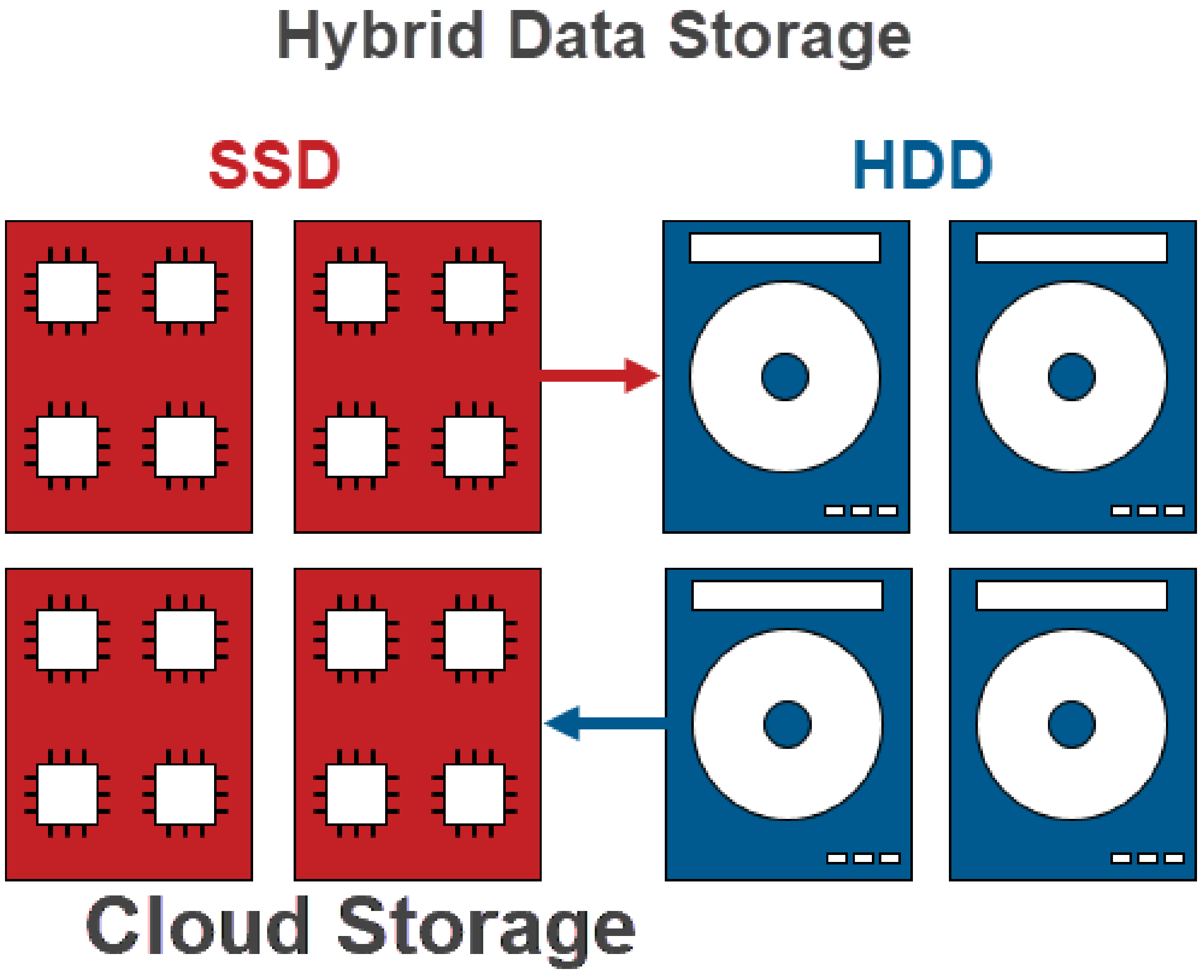
- DAS - Direct-Attached Storage
- NAS - Network-Attached Storage
- SAN - Storage Area Network

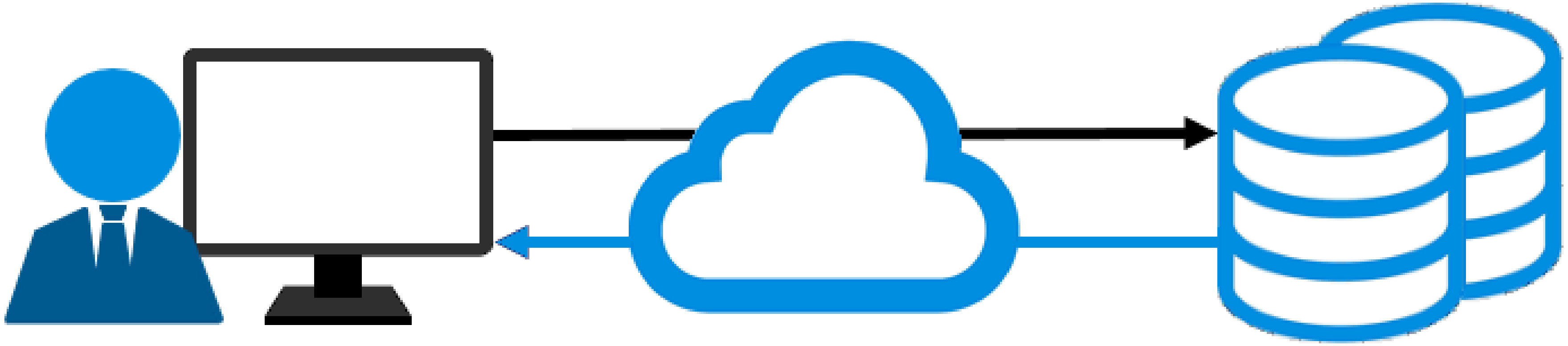


2.4. DATA STORAGE

Data Storage Solutions:

- Flash and SSD
- Hybrid
- Cloud

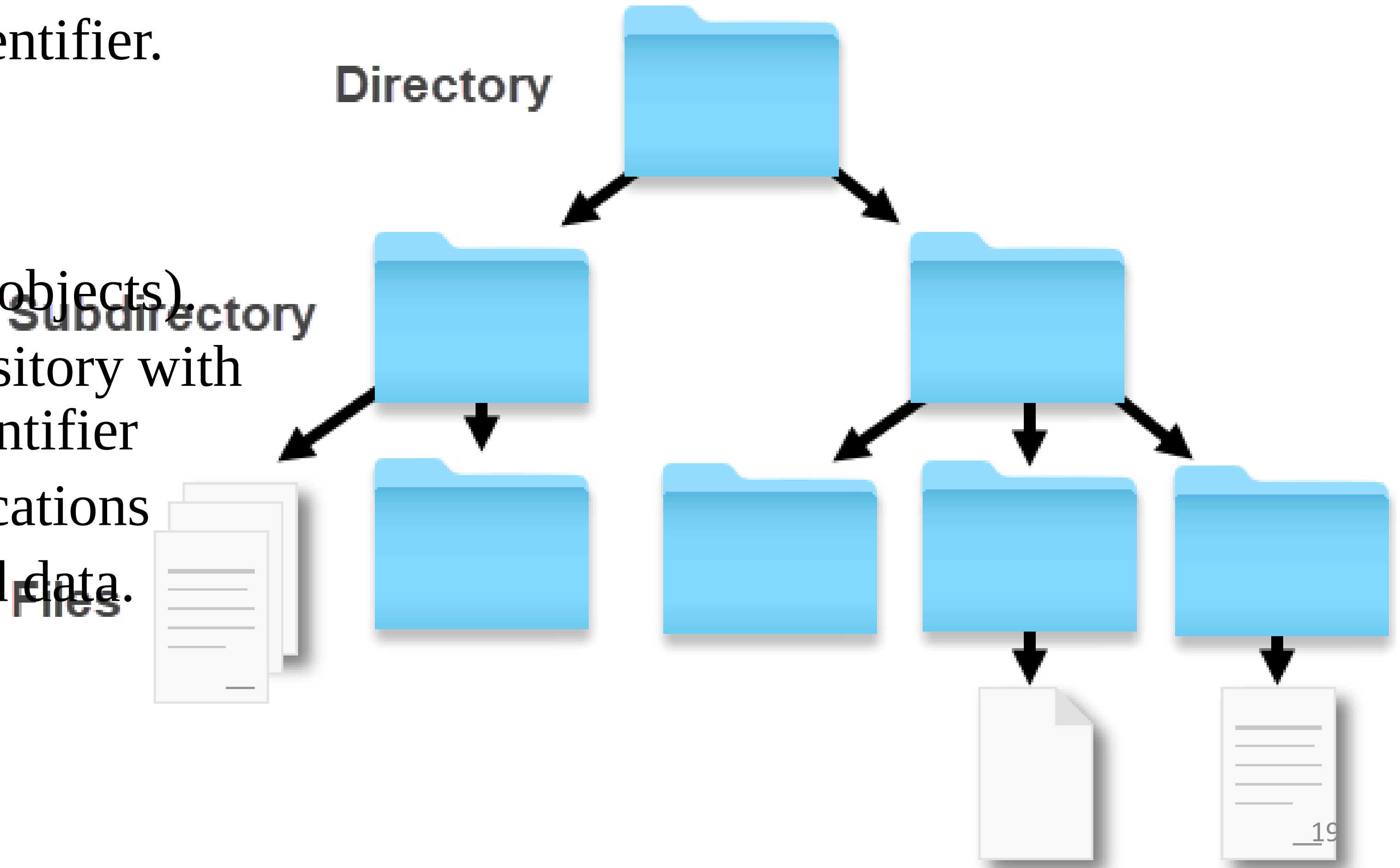




2.4. DATA STORAGE

Forms of Data Storage

- File Storage: hierarchical organizing and storing data
- Block Storage:
  - Group data to blocks: unique identifier.
  - SAN, cloud-based storages.
- Object Storage:
  - organizes data in discrete units (objects).
  - Each object is independent repository with data, metadata, and a unique identifier
  - Cloud-native, AI, big data applications
  - Scalable, enormous unstructured data.



3. BLOCKCHAIN SCALABILITY

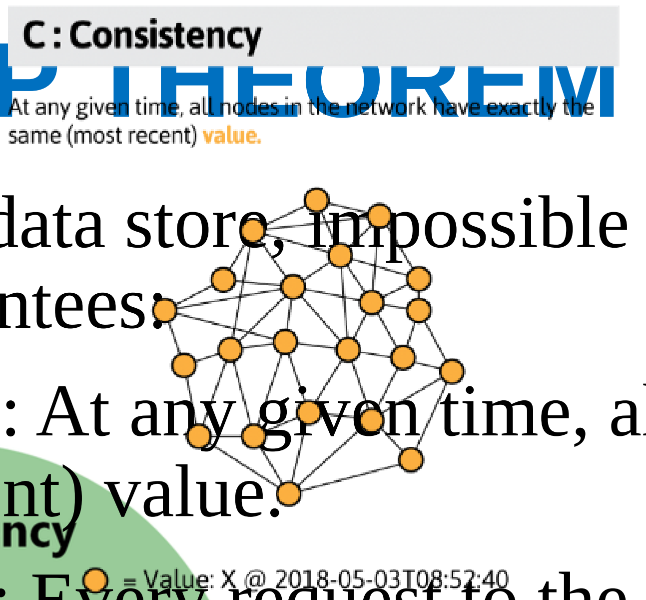
3.1. CAP THEOREM

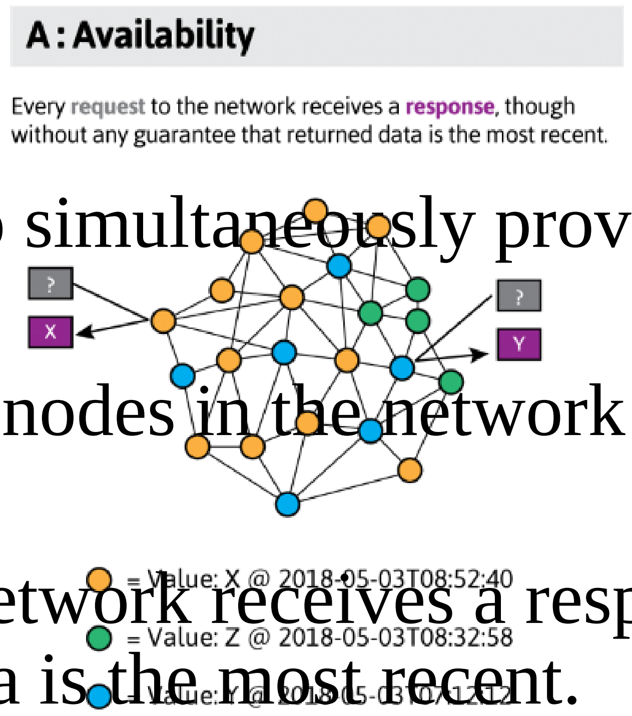
3.2. BLOCKCHAIN TRILEMMA

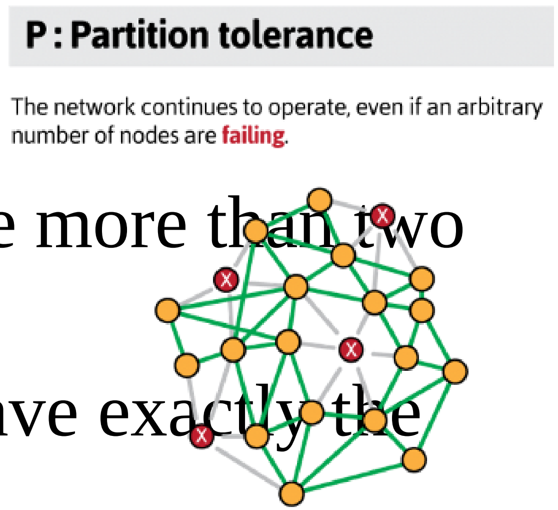
3.3. SCALABILITY PROBLEMS

3.4. SCALABILITY SOLUTIONS

3.1. CAP THEOREM

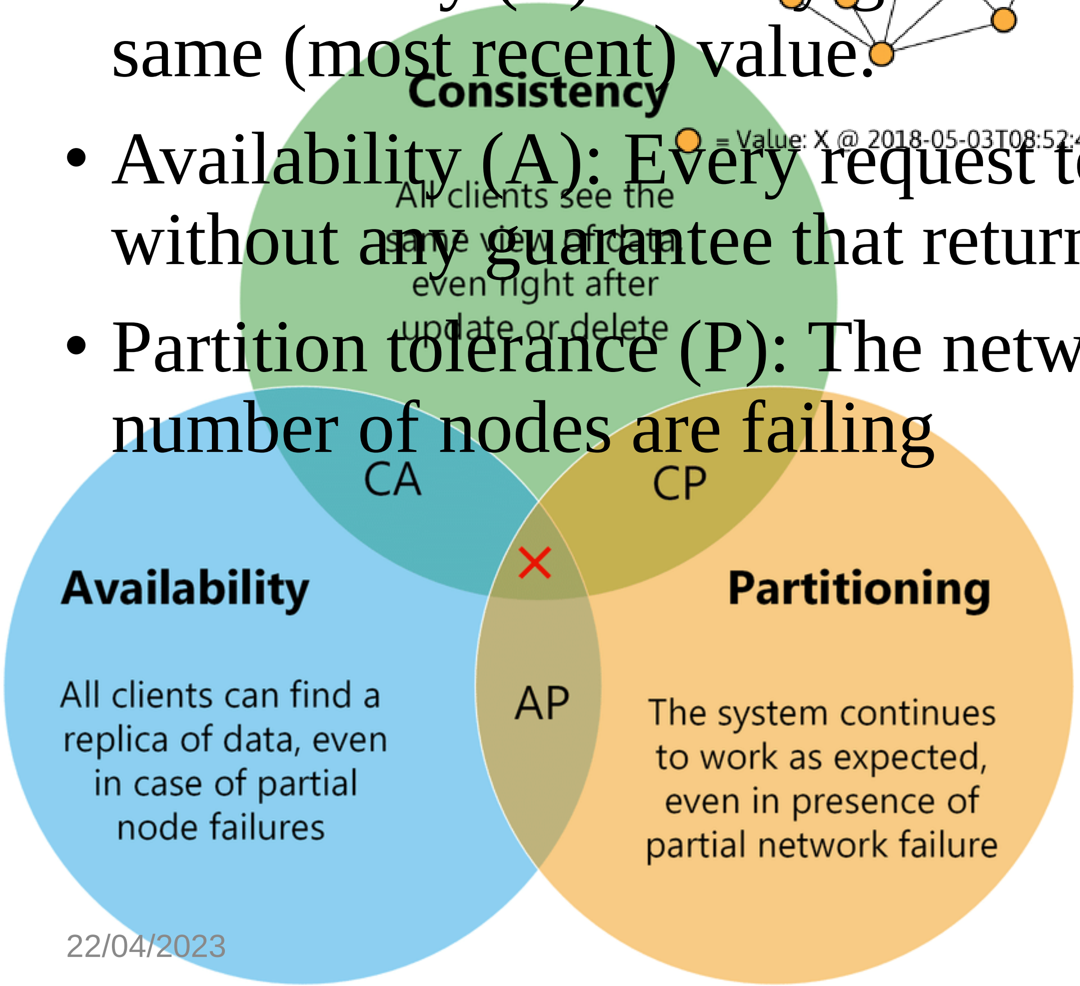






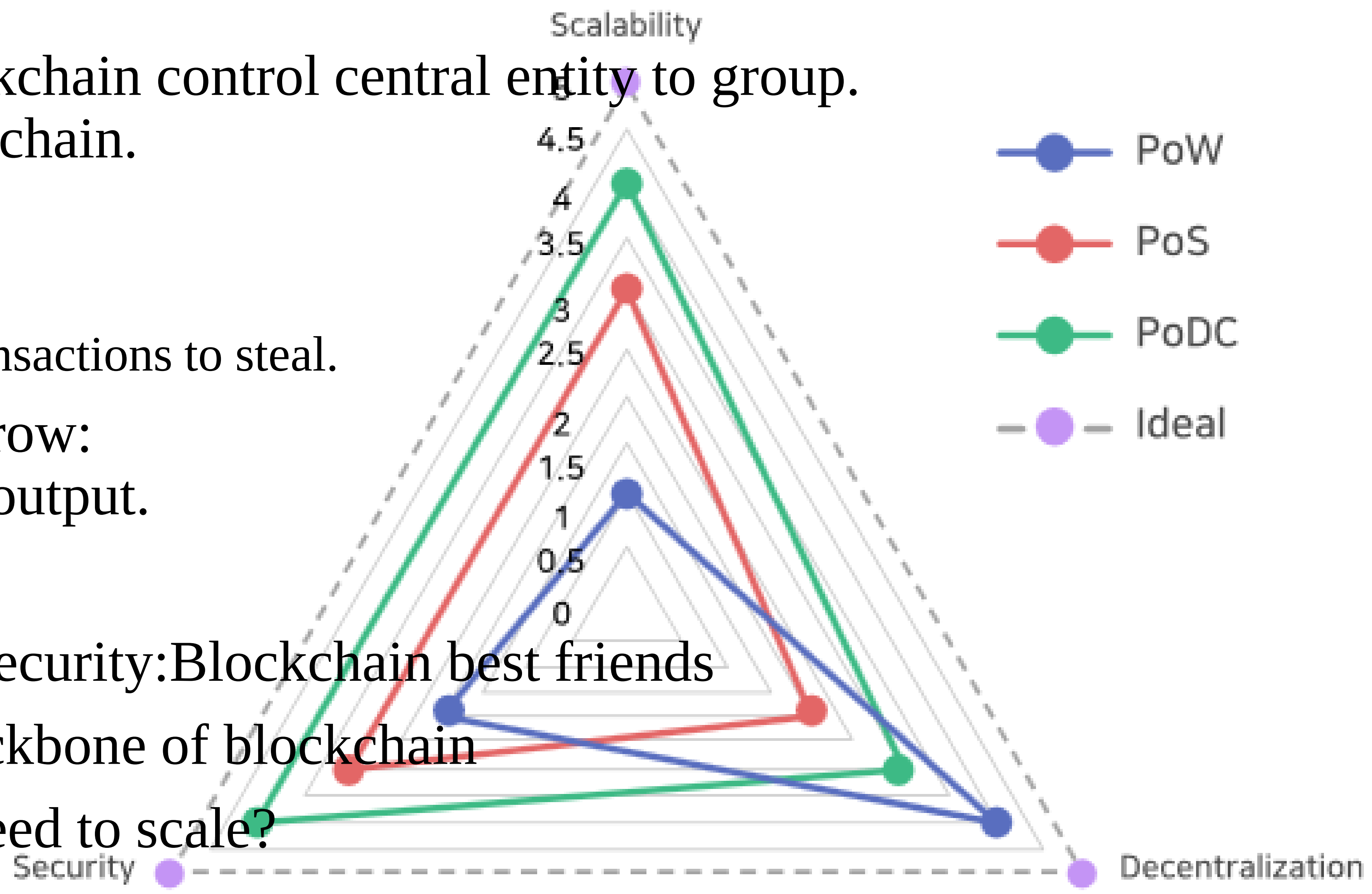
For a distributed data store, impossible to simultaneously provide more than two of the three guarantees:

- Consistency (C): At any given time, all nodes in the network have exactly the same (most recent) value.
- Availability (A): Every request to the network receives a response, though without any guarantee that returned data is the most recent.
- Partition tolerance (P): The network continues to operate, even if an arbitrary number of nodes are failing



3.2. BLOCKCHAIN TRILEMMA

- Decentralization: blockchain control central entity to group. power to govern blockchain.
- Security:
  - Inner: network (51%)
  - Outter: manipulate transactions to steal.
- Scalability: network grow: transaction speed and output.



Comparison of Blockchain Trilemma Solutions

- Decentralization and security:Blockchain best friends
- Decentralization is backbone of blockchain
- why do blockchains need to scale?

3.3. SCALABILITY PROBLEMS

Scalability Problem:

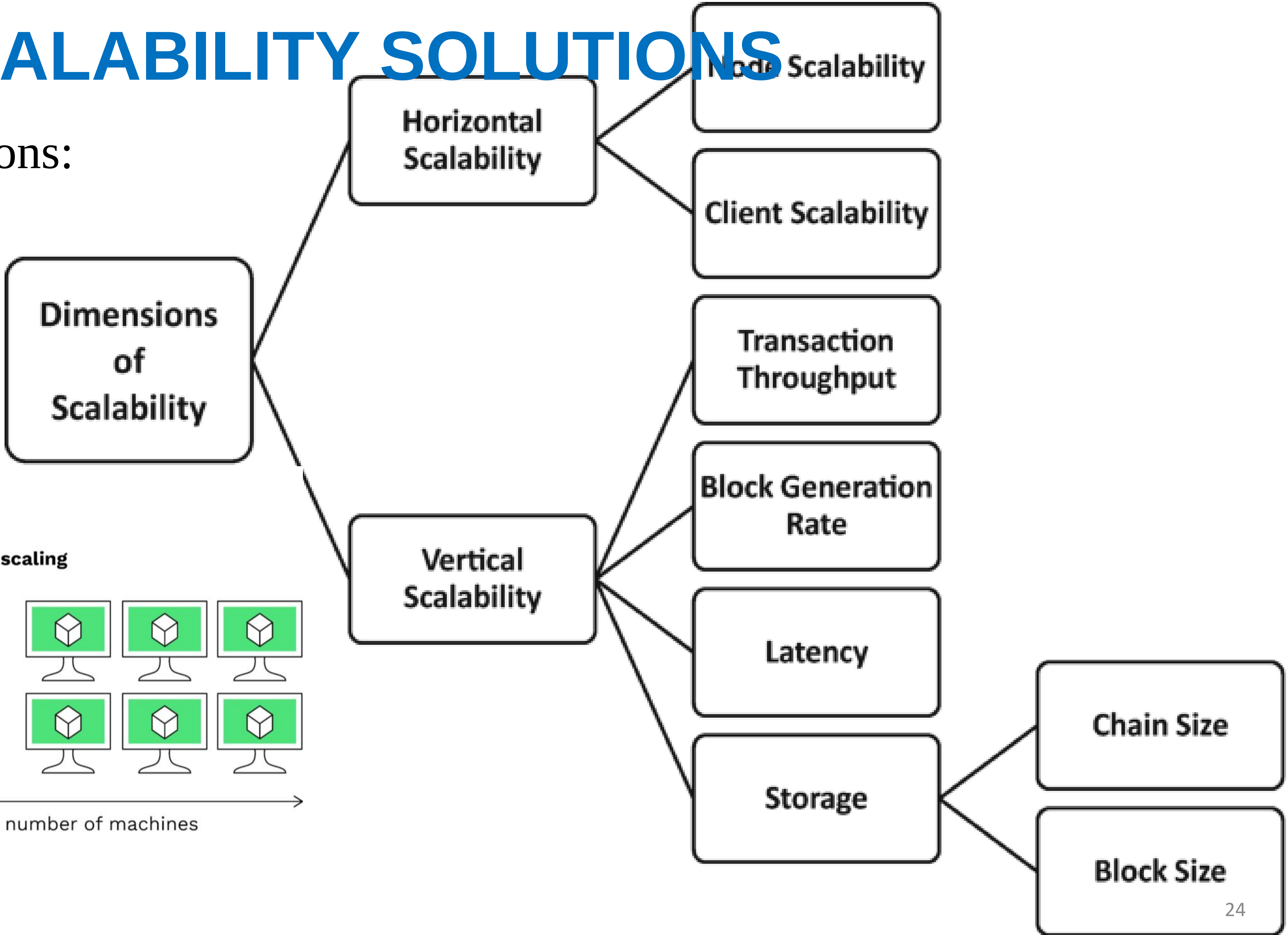
- Transactions, user increase
- Data grow
- Instant transactions

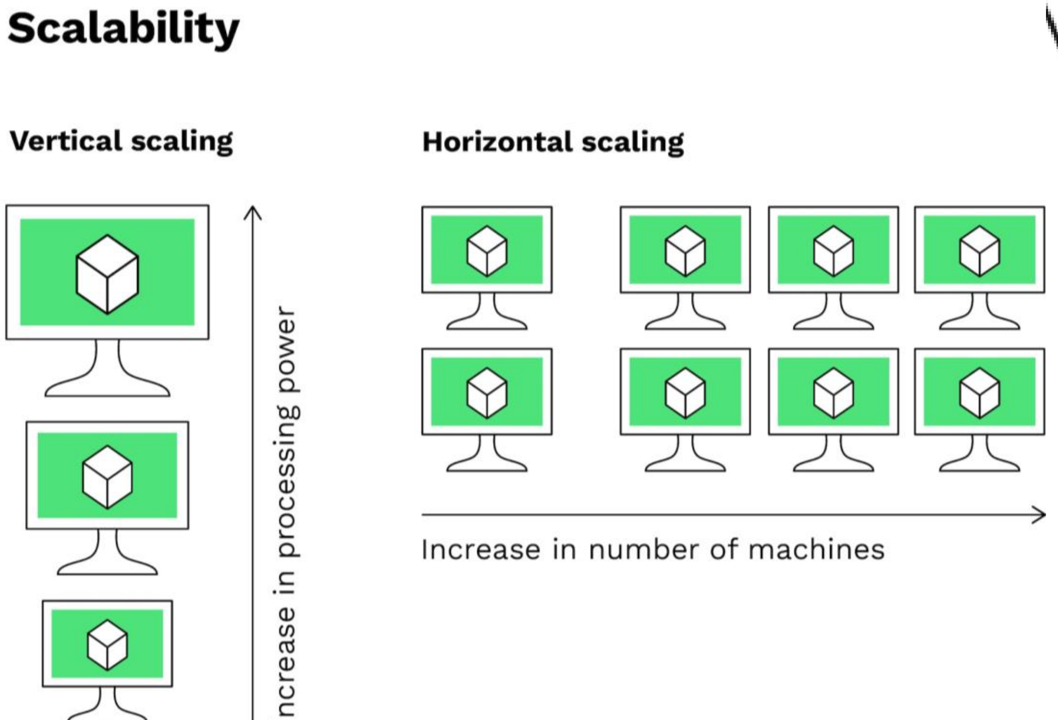
Scalability :

- Network
- Throughput
- Storing large data

3.4. SCALABILITY SOLUTIONS

Scalability solutions:





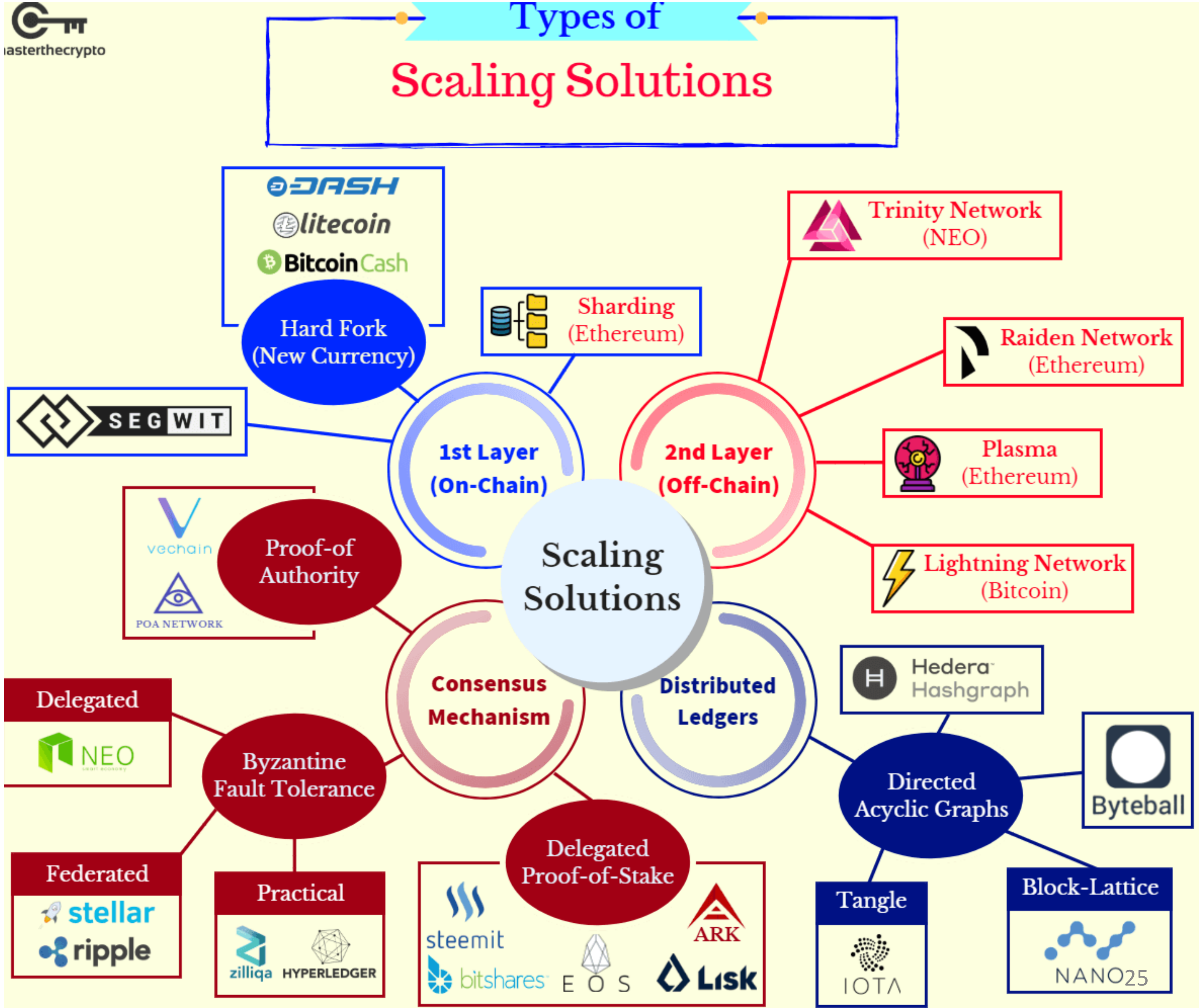
3.4. SCALABILITY SOLUTIONS

Key Properties of Scaling:

- Execution
- Storage
- Consensus

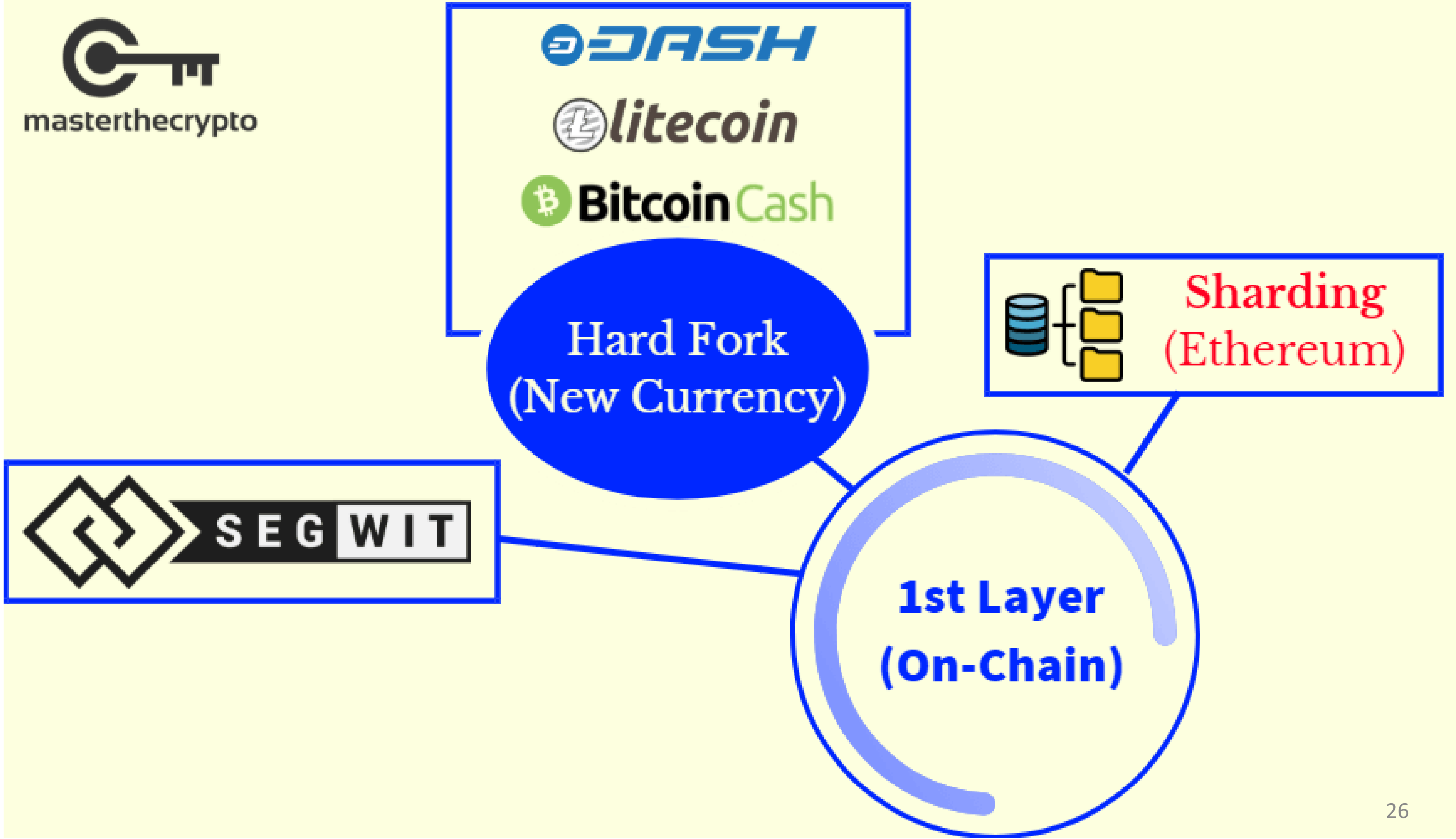
Scalability directions:

- 1st Layer (on-chain)
- 2nd Layer (off-chain)
- Consensus Layer



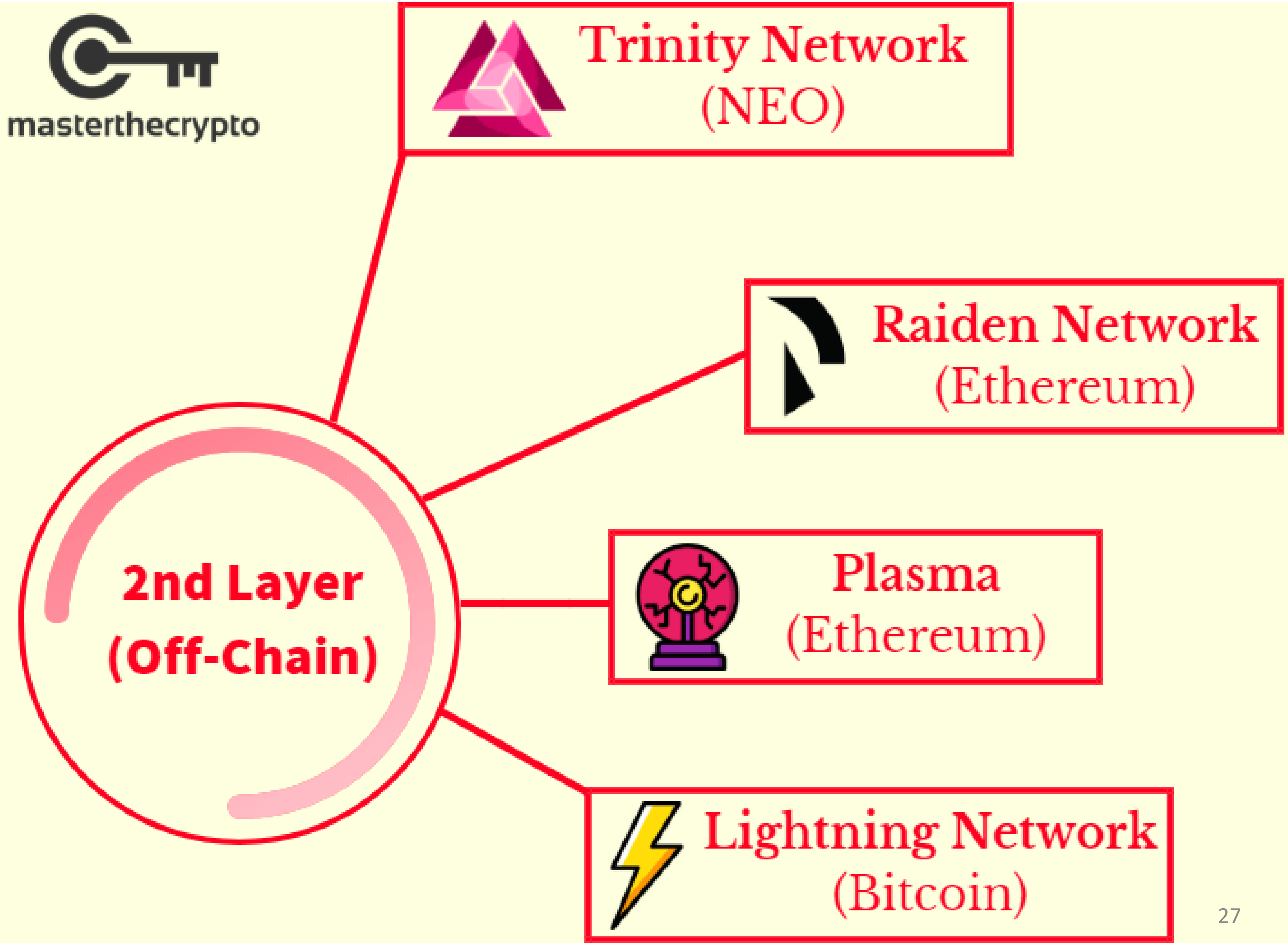
3.4. SCALABILITY SOLUTIONS

1st Layer Scalability:



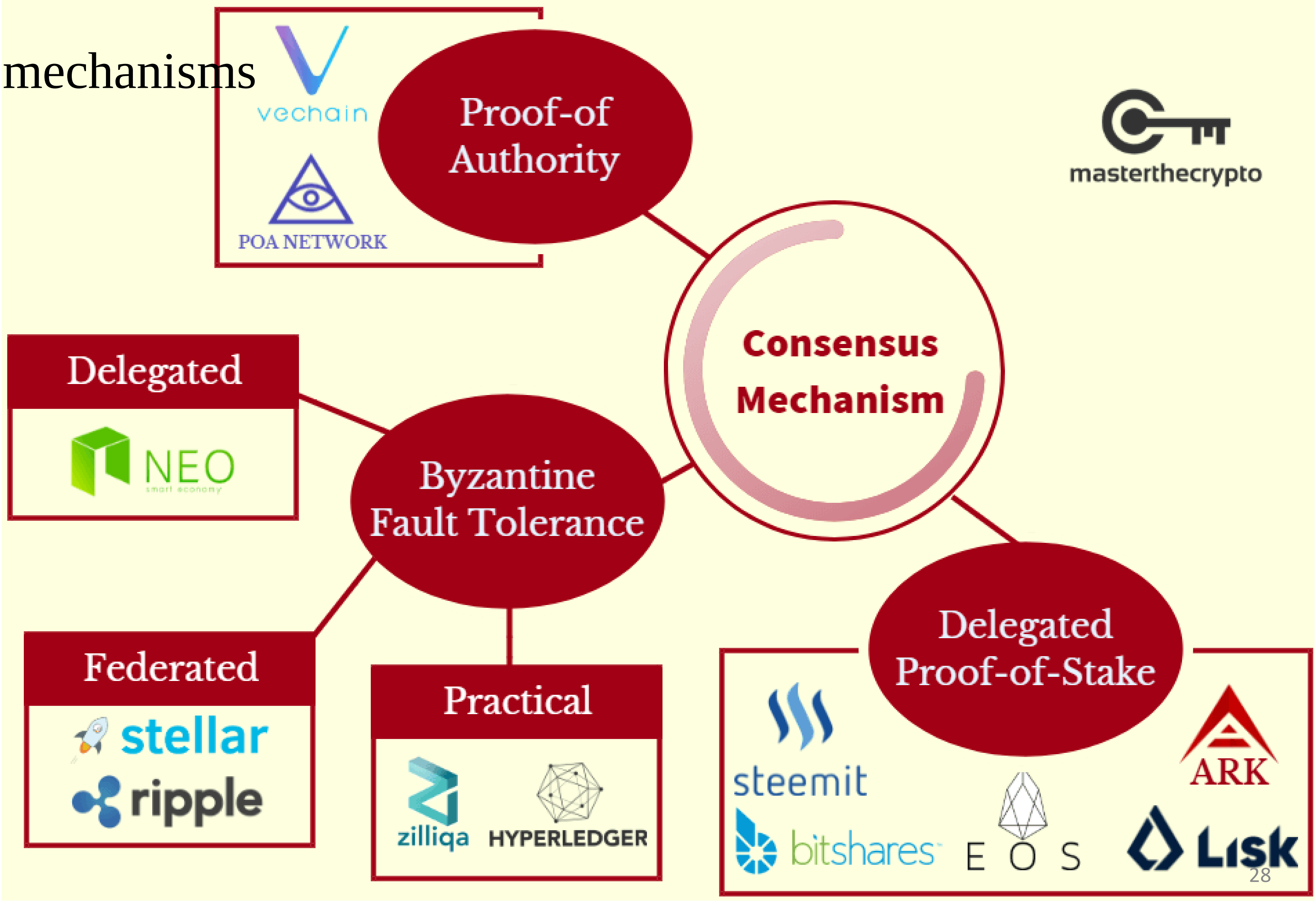
3.4. SCALABILITY SOLUTIONS

2nd Layer Scalability:



3.4. SCALABILITY SOLUTIONS

Scalable consensus mechanisms



4. BLOCKCHAIN CROSS-CHAIN

4.1. INTRODUCTION

4.2. BLOCKCHAIN CROSS-CHAIN

4.3. CROSS-CHAIN MECHANISMS

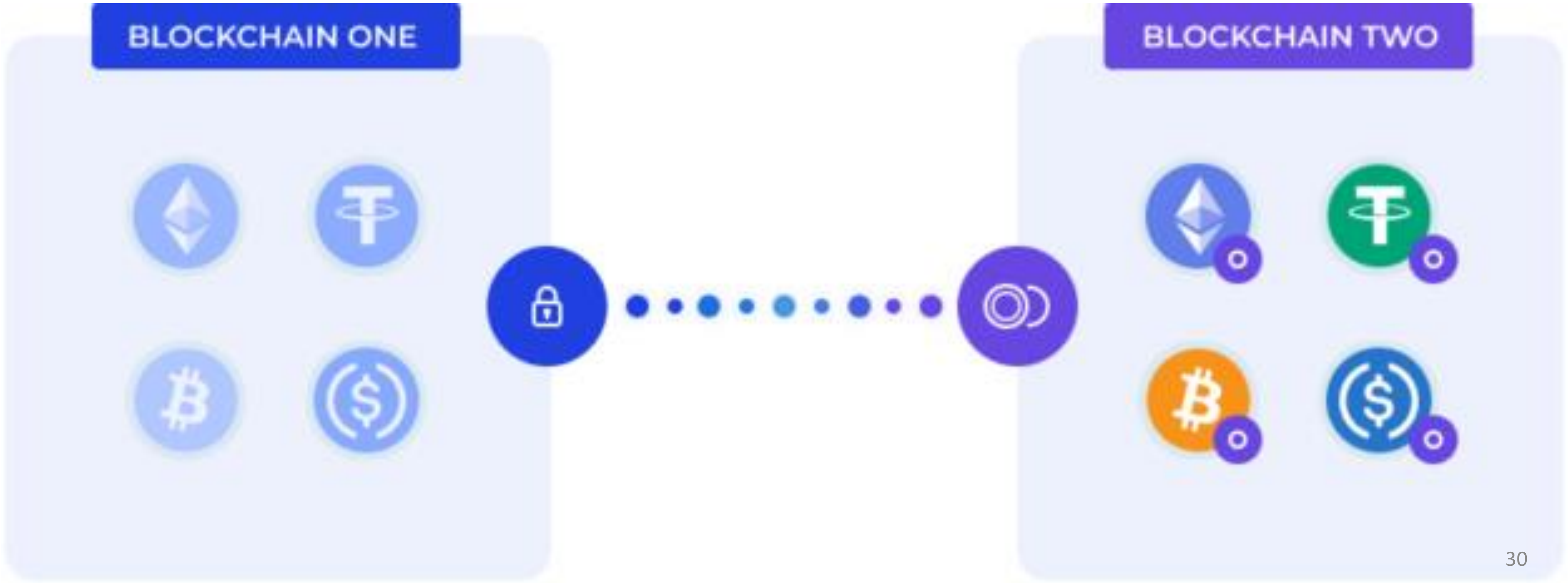
4.4. CROSS-CHAIN CLASSIFICATION

4.5. CROSS-CHAIN COMMUNICATION

4.1. INTRODUCTION

Blockchain problems:

- Blockchain operate in isolation
- Not allow to communicate with each other



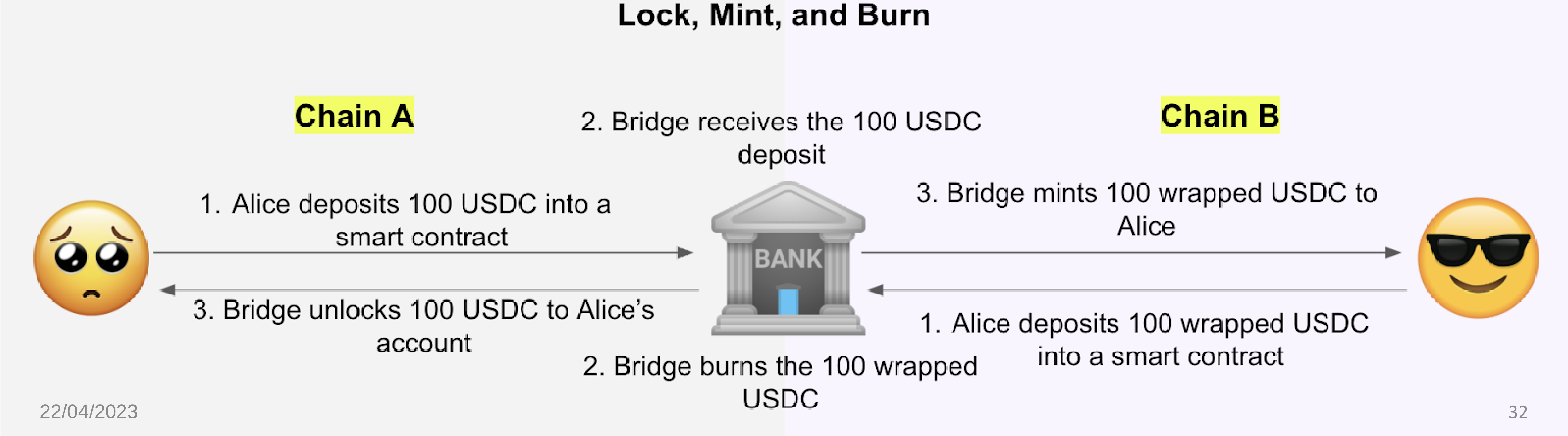
4.2. BLOCKCHAIN CROSS-CHAIN

Blockchain cross-chain: cross-chain bridge

- Interoperability (connects independent) between blockchains
- Enables transfer of assets and information between two or more blockchains
- Protocol allowing users to access other protocols easily

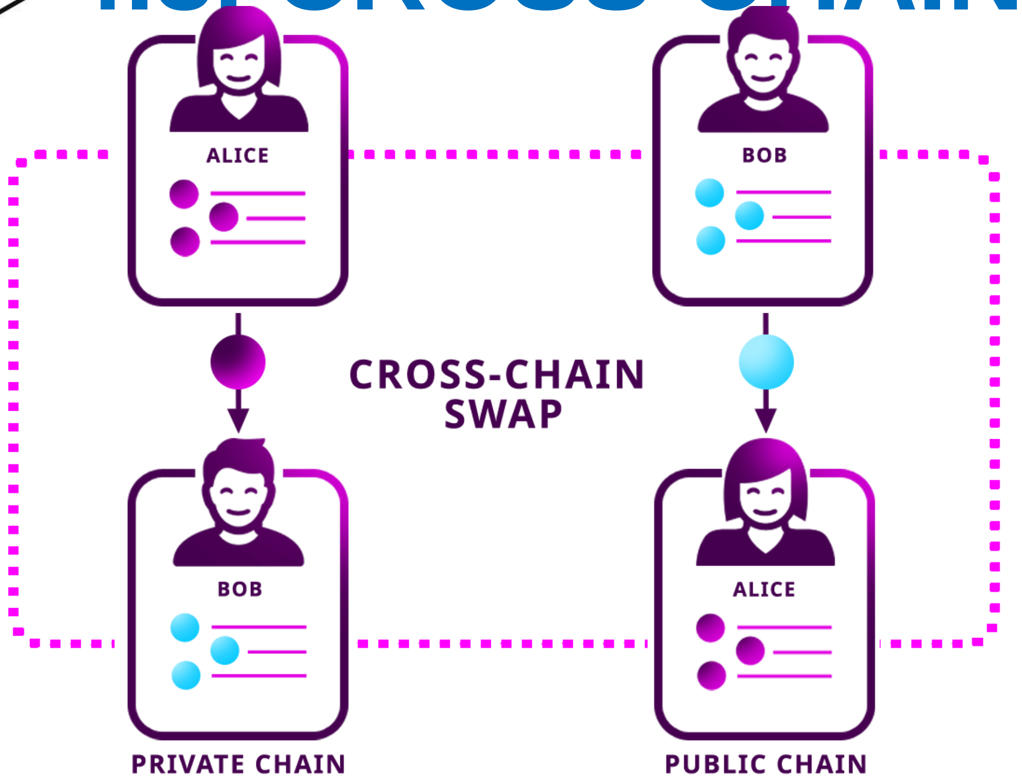
4.3. CROSS-CHAIN MECHANISMS

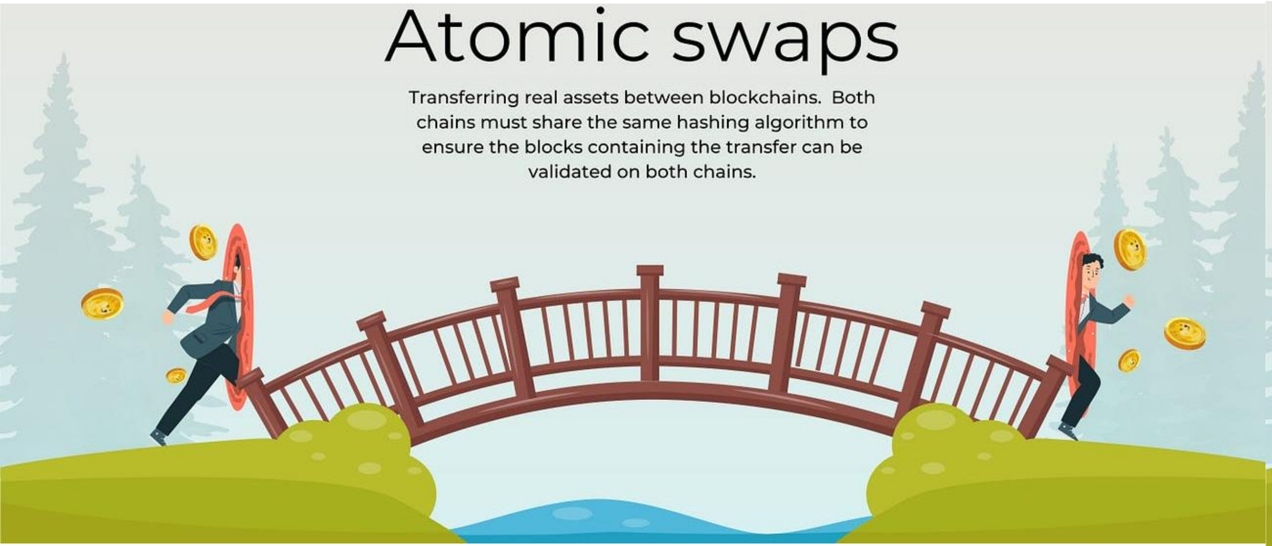
- Lock and mint: lock tokens on chain A, then minted tokens on chain B as IOUs.
- Burn and mint: tokens burn on chain A, then tokens are minted on chain B.
- Lock and unlock: When tokens are sent from Chain A to chain B, users lock tokens on chain A, then unlock the same native token from the liquidity pool on chain B.

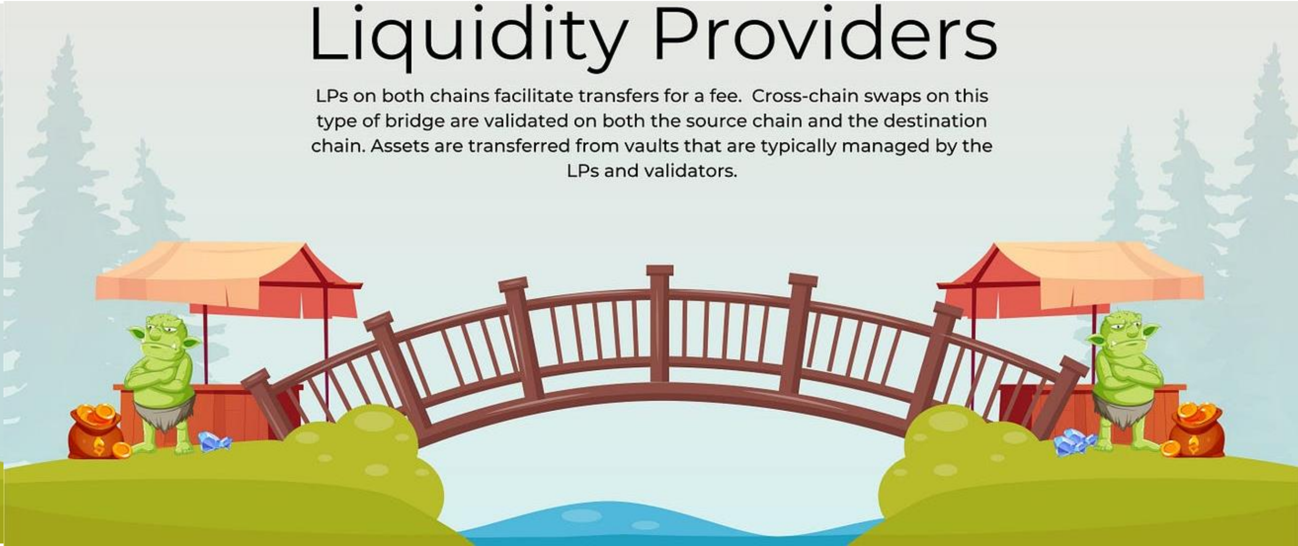


4.3. CROSS-CHAIN MECHANISMS

-





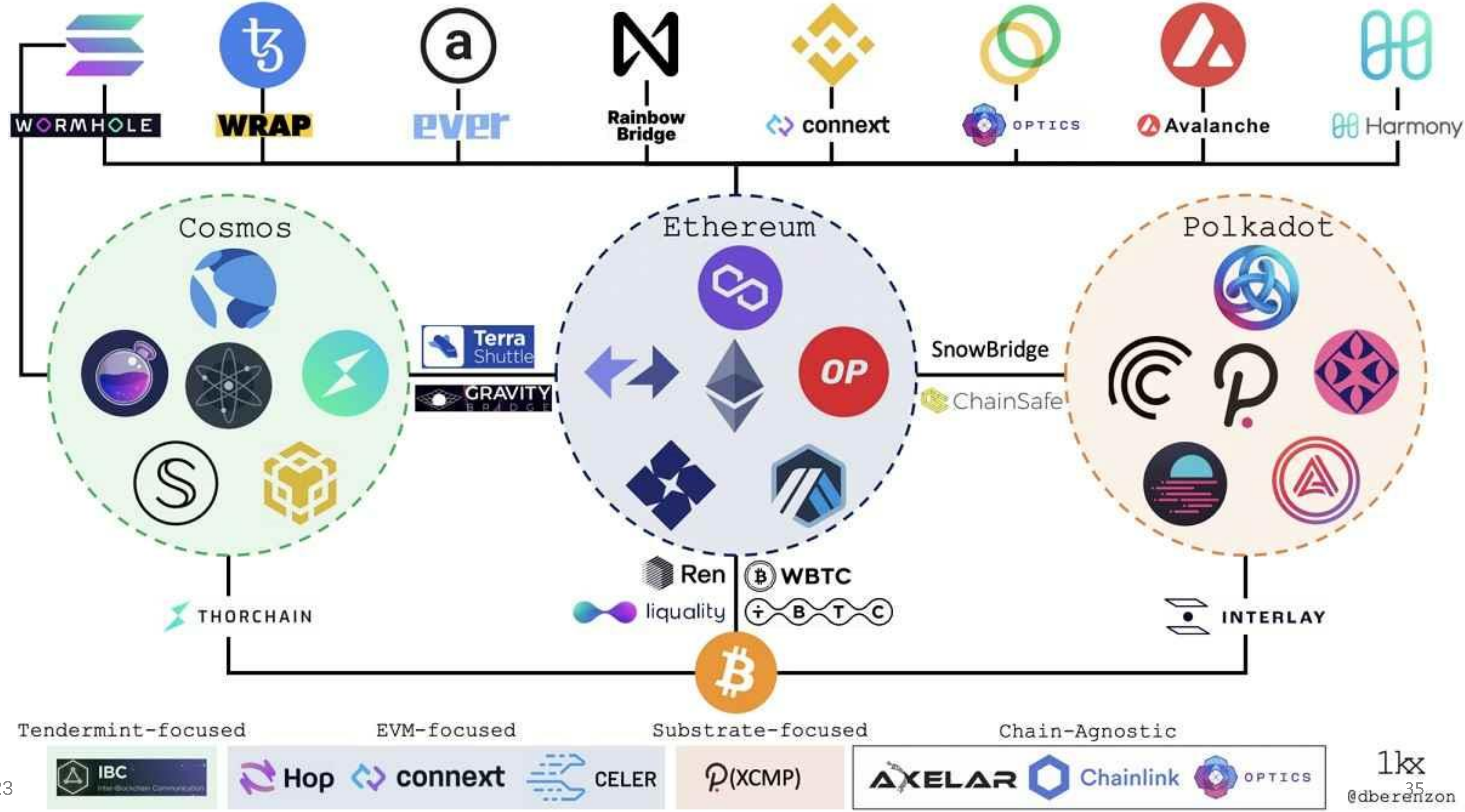


4.4. CROSS-CHAIN CLASSIFICATION

- Chain-To-Chain Bridges
- Multi-Chain Bridges
- Specialized Bridges
- Wrapped Asset Bridges
- Data Specific Bridges
- dApp Specific Bridges

4.5. CROSS-CHAIN COMMUNICATION

•



5. BLOCKCHAIN ORACLE

5.1. INTRODUCTION

5.2. BLOCKCHAIN ORACLE

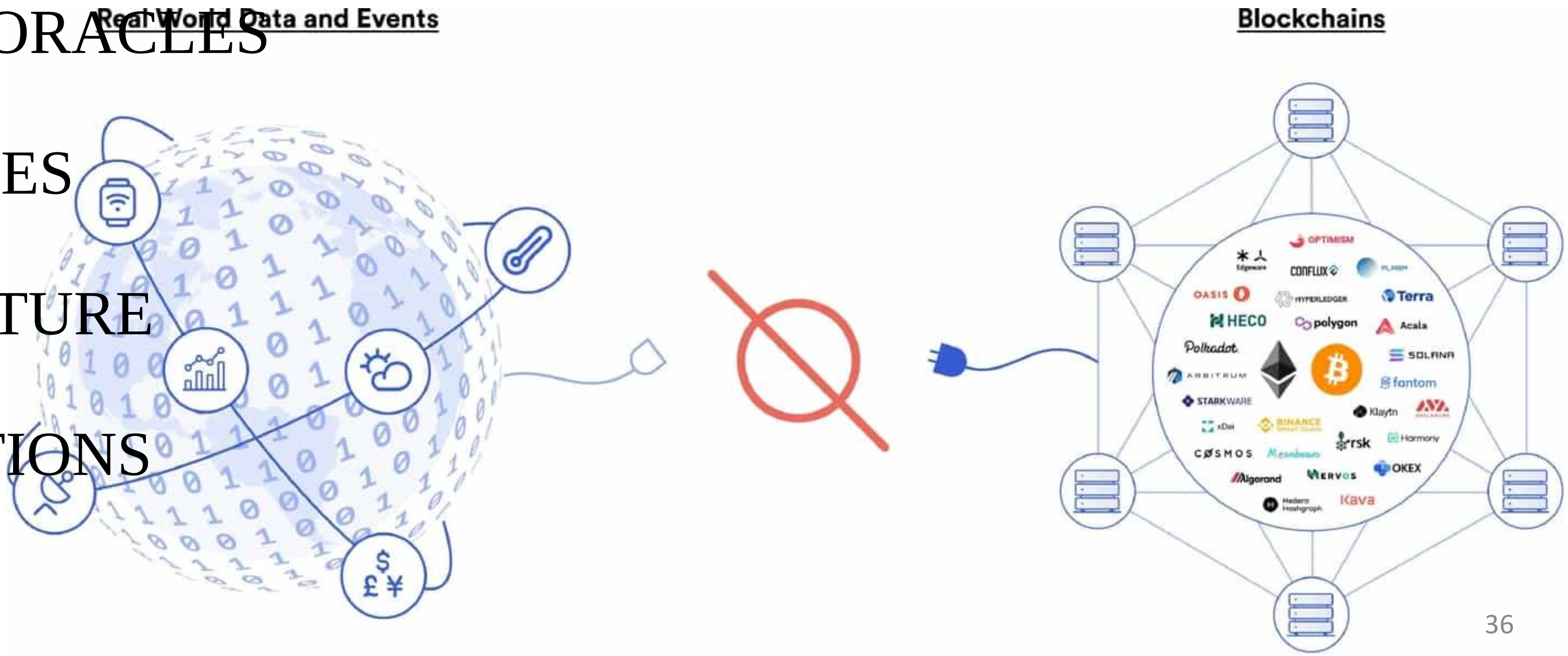
5.3. ARCHITECTURE OF ORACLE

5.4. FUNCTIONS OF ORACLES

5.5. TYPE OF ORACLES

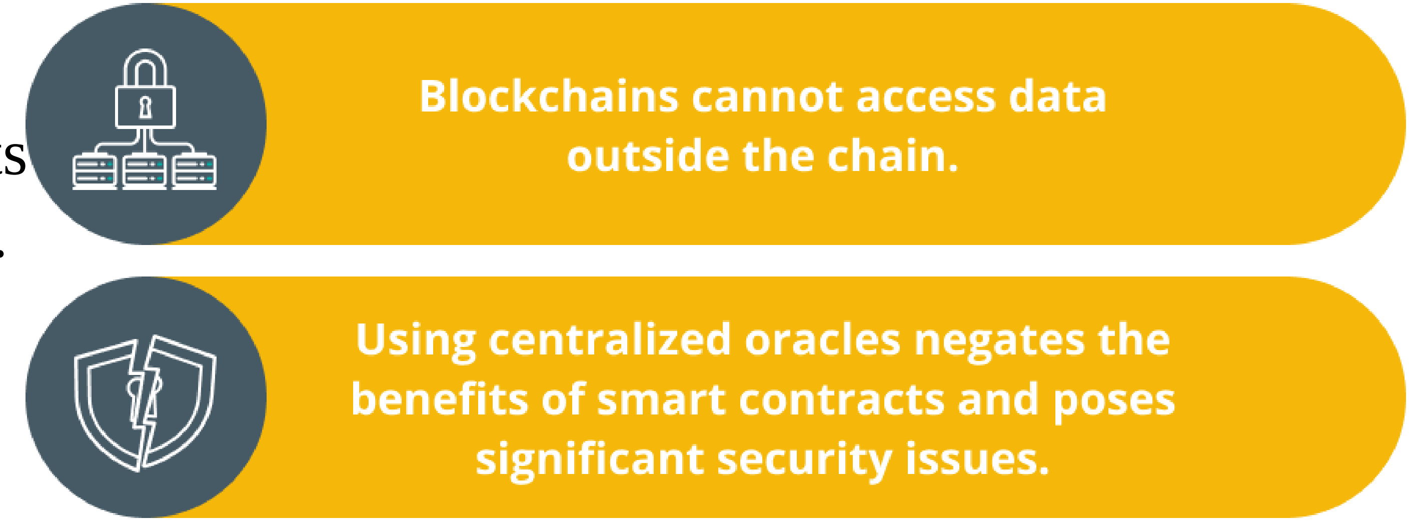
5.6. ORACLE STRUCTURE

5.7. ORACLE SOLUTIONS



5.1. INTRODUCTION

- Blockchains cannot access external data on their own
- Centralized oracles:
  - Negates the benefit of smart contracts
  - Introduces significant privacy issues.



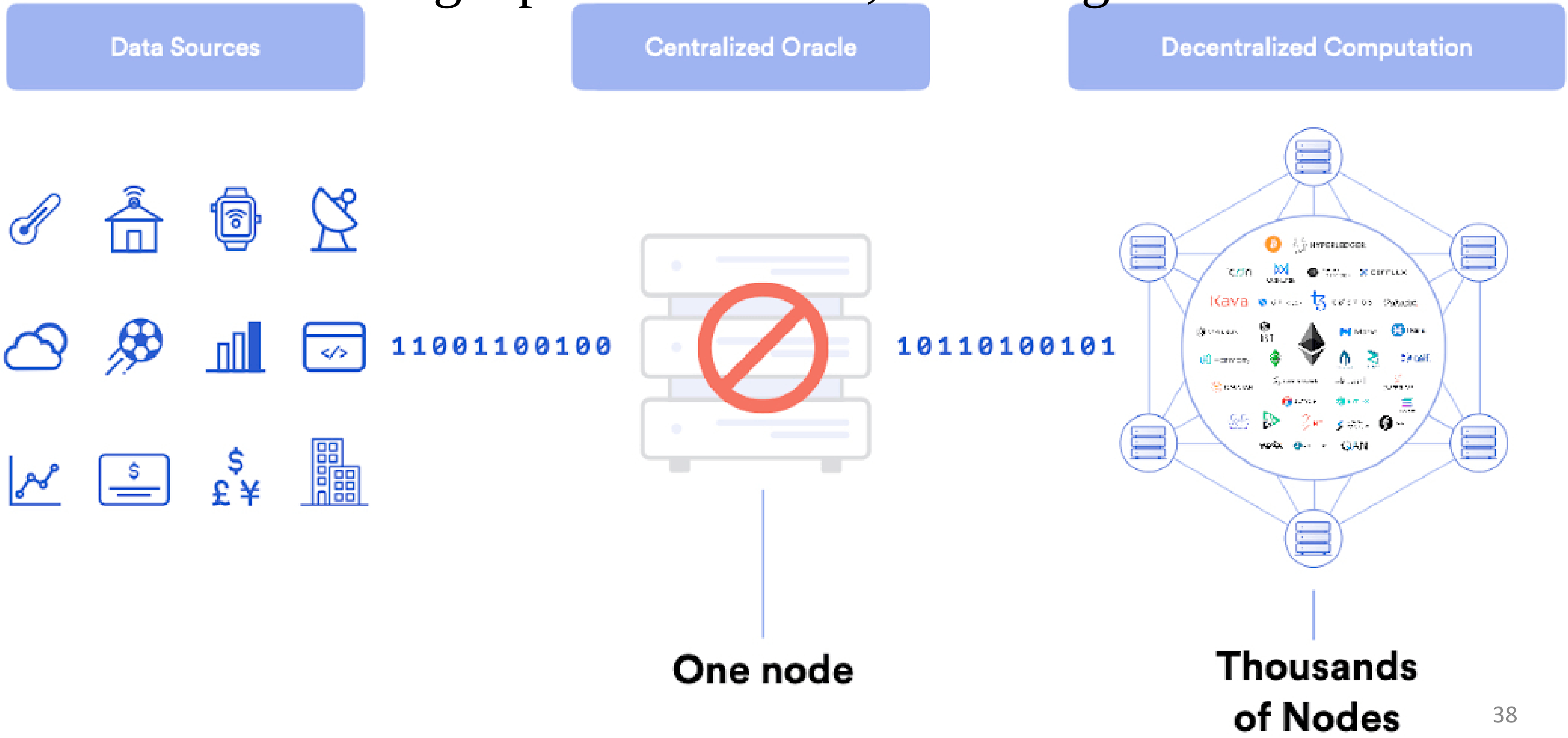
Solve the oracle problem:

- Use of public sources of data,
- Getting consensus among different sources/oracles

5.1. INTRODUCTION

fundamental limitation of smart contracts:

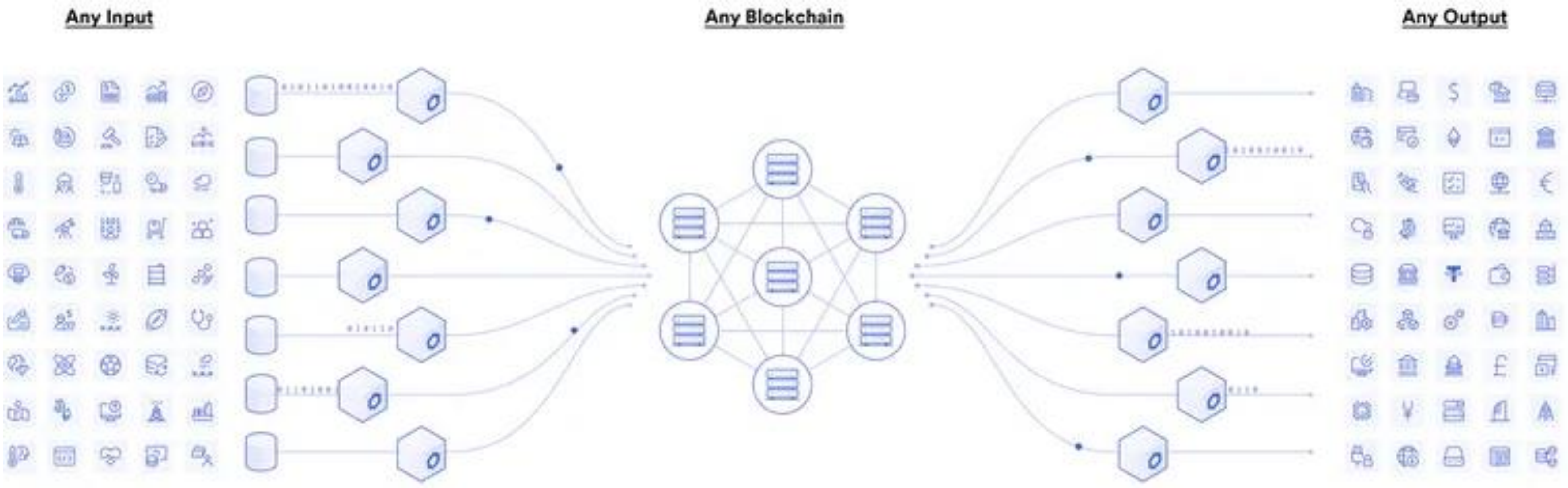
- cannot interact with data and systems outside native blockchain
- centralized oracle introduce a single point of failure, defeating decentralized



5.2. BLOCKCHAIN ORACLE

Blockchain oracles:

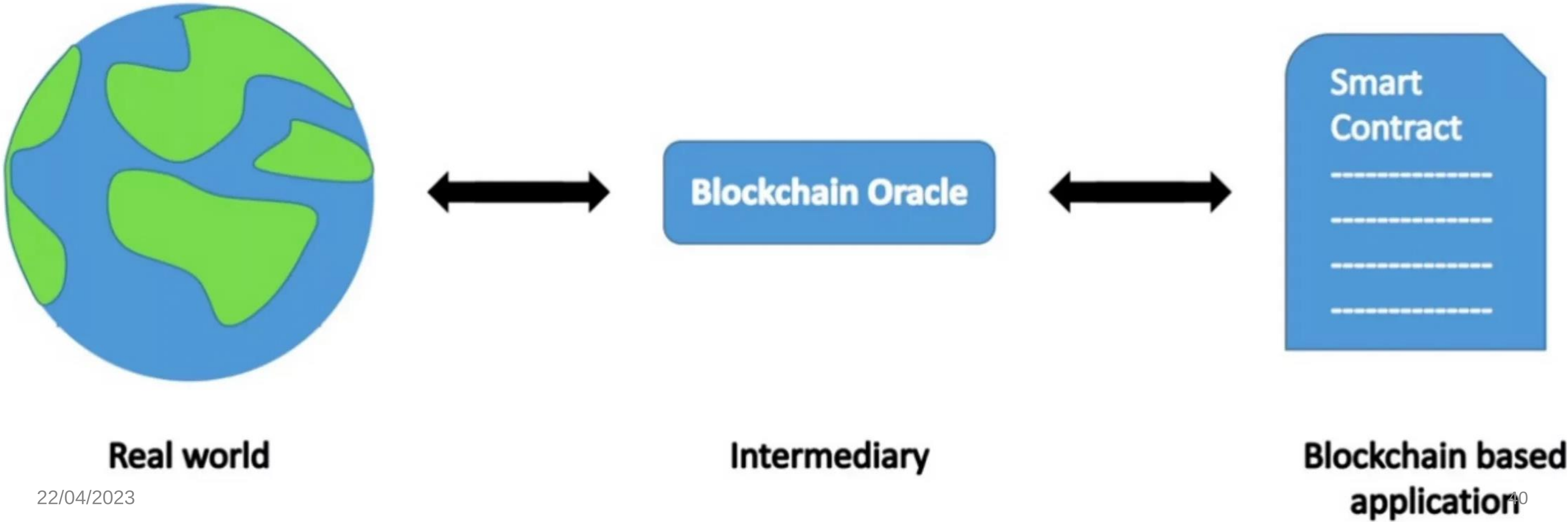
- External systems connect blockchains
- Input/output to/from smartcontracts to/from real world.



5.2. BLOCKCHAIN ORACLE

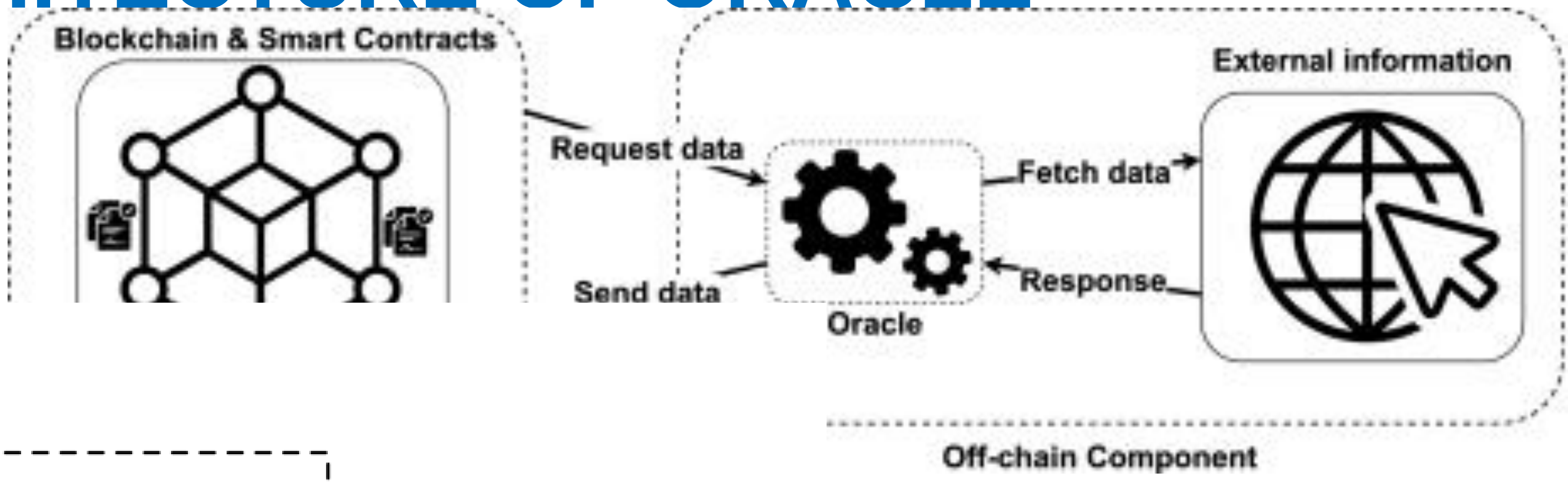
Blockchain oracles:

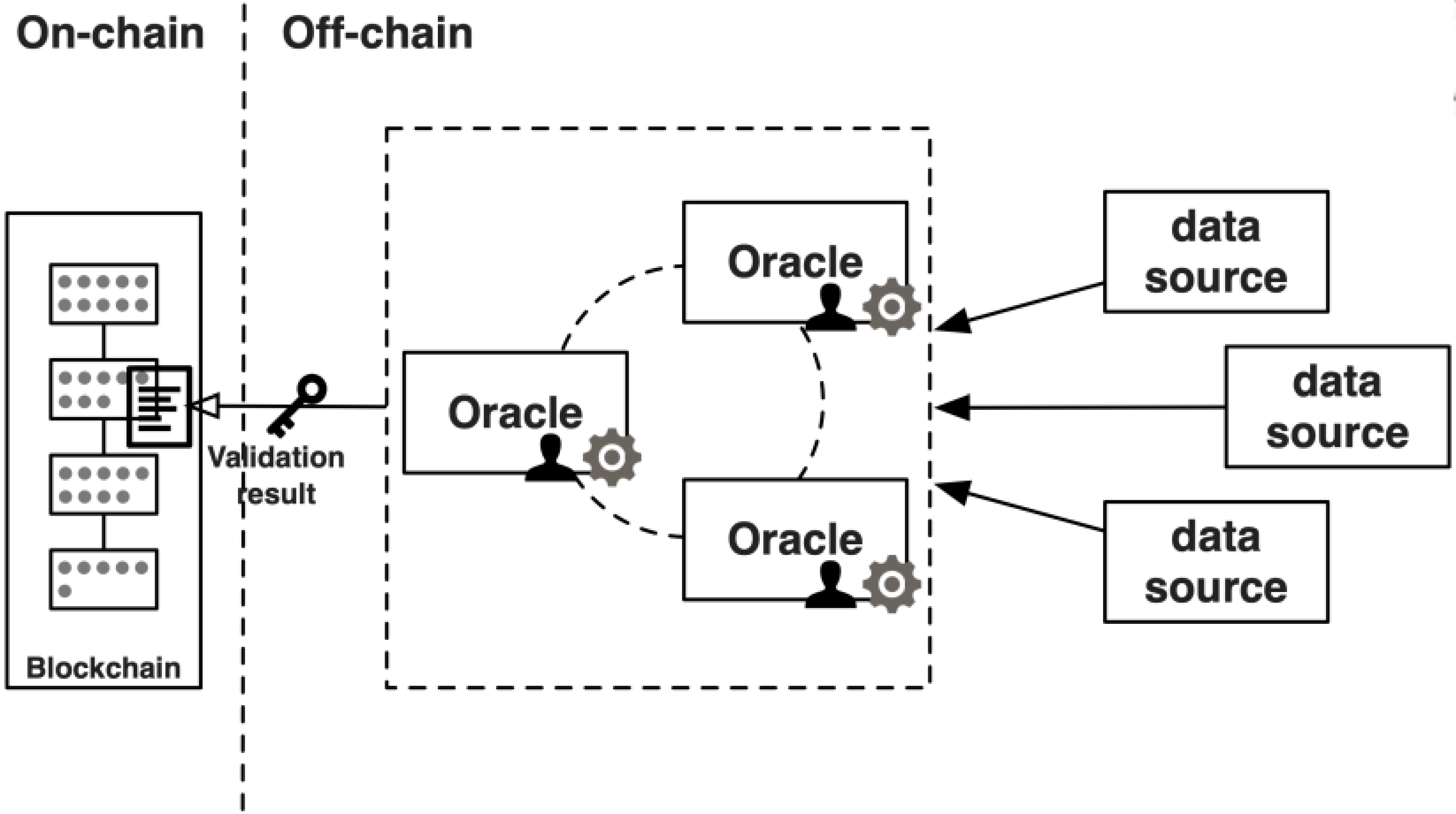
- Independent, third-party service
- bridge blockchain with outside



5.3. ARCHITECTURE OF ORACLE

•

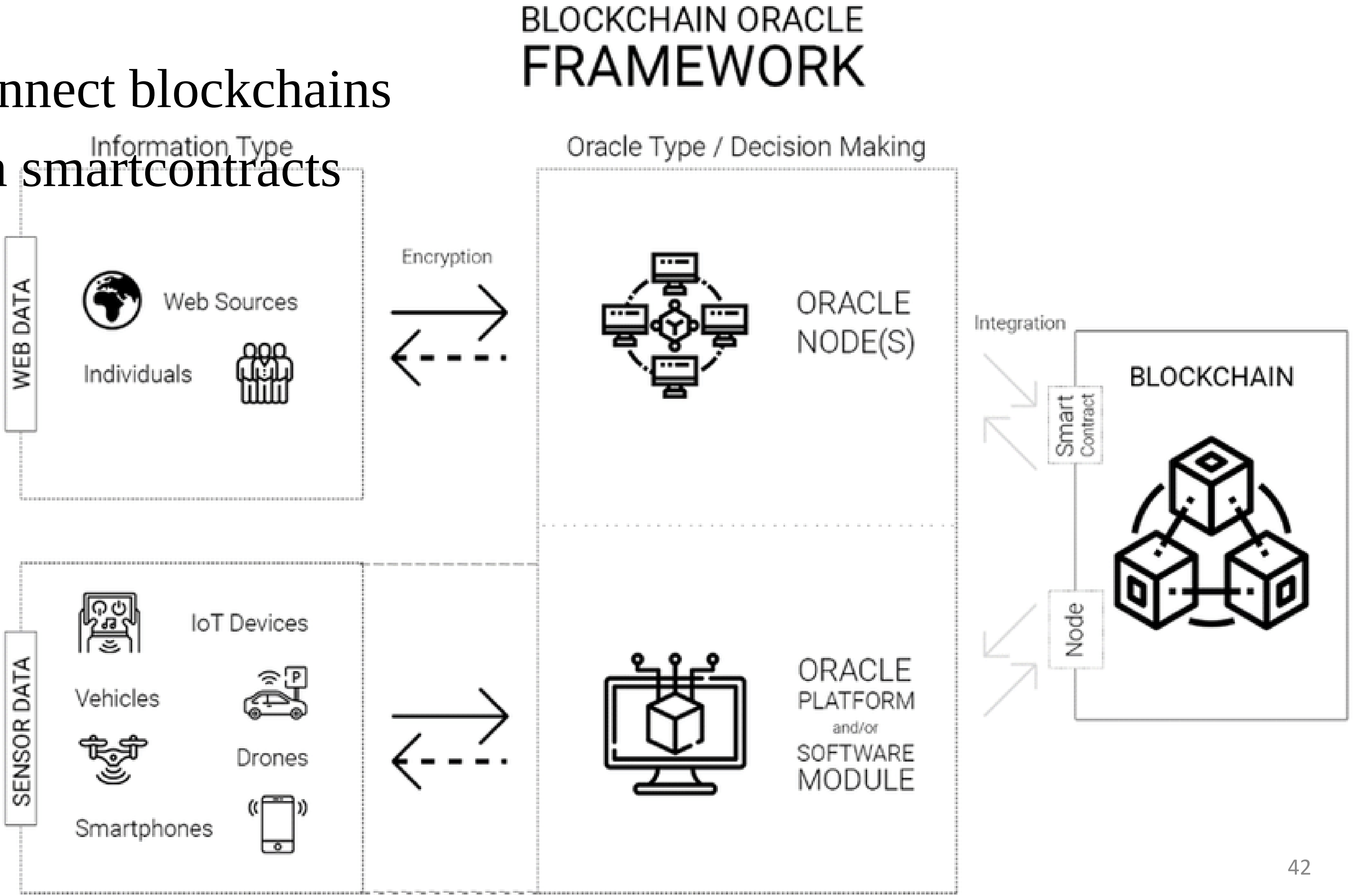




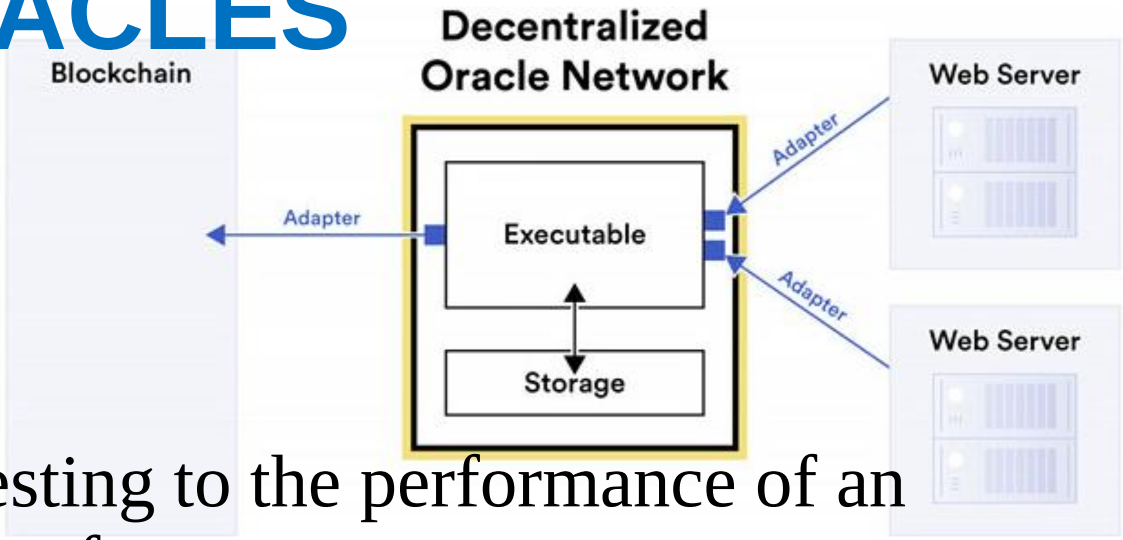
5.3. ARCHITECTURE OF ORACLE

Blockchain oracles:

- External systems connect blockchains
- Input/output to/from smartcontracts to/from real world.



5.4. FUNCTIONS OF ORACLES



- Listen: check for off-chain data requests
- Extract: fetch data from external systems.
- Format: format data from/to external APIs.
- Validate: generate a cryptographic proof attesting to the performance of an oracle service (signing), zero-knowledge proofs.
- Compute: perform secure off-chain computation.
- Broadcast: broadcast transactions to send data and any corresponding proof on-chain for consumption by the smart contract.
- Output: send data to an external system upon the execution of a smart contract.

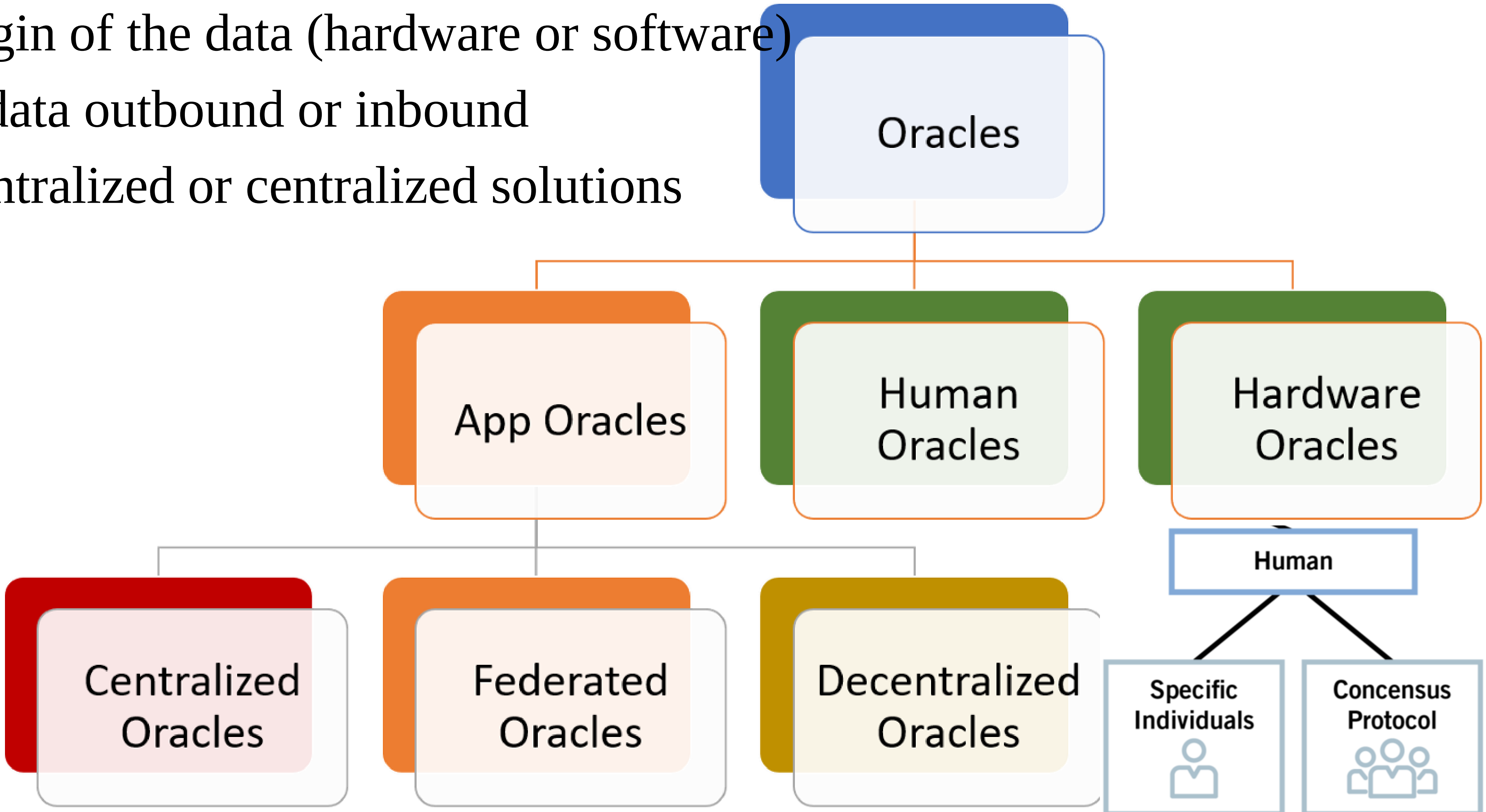
5.5. TYPE OF ORACLES

Blockchain oracles classification:

- Source: origin of the data (hardware or software)
- Direction: data outbound or inbound
- Trust: decentralized or centralized solutions

Source data:

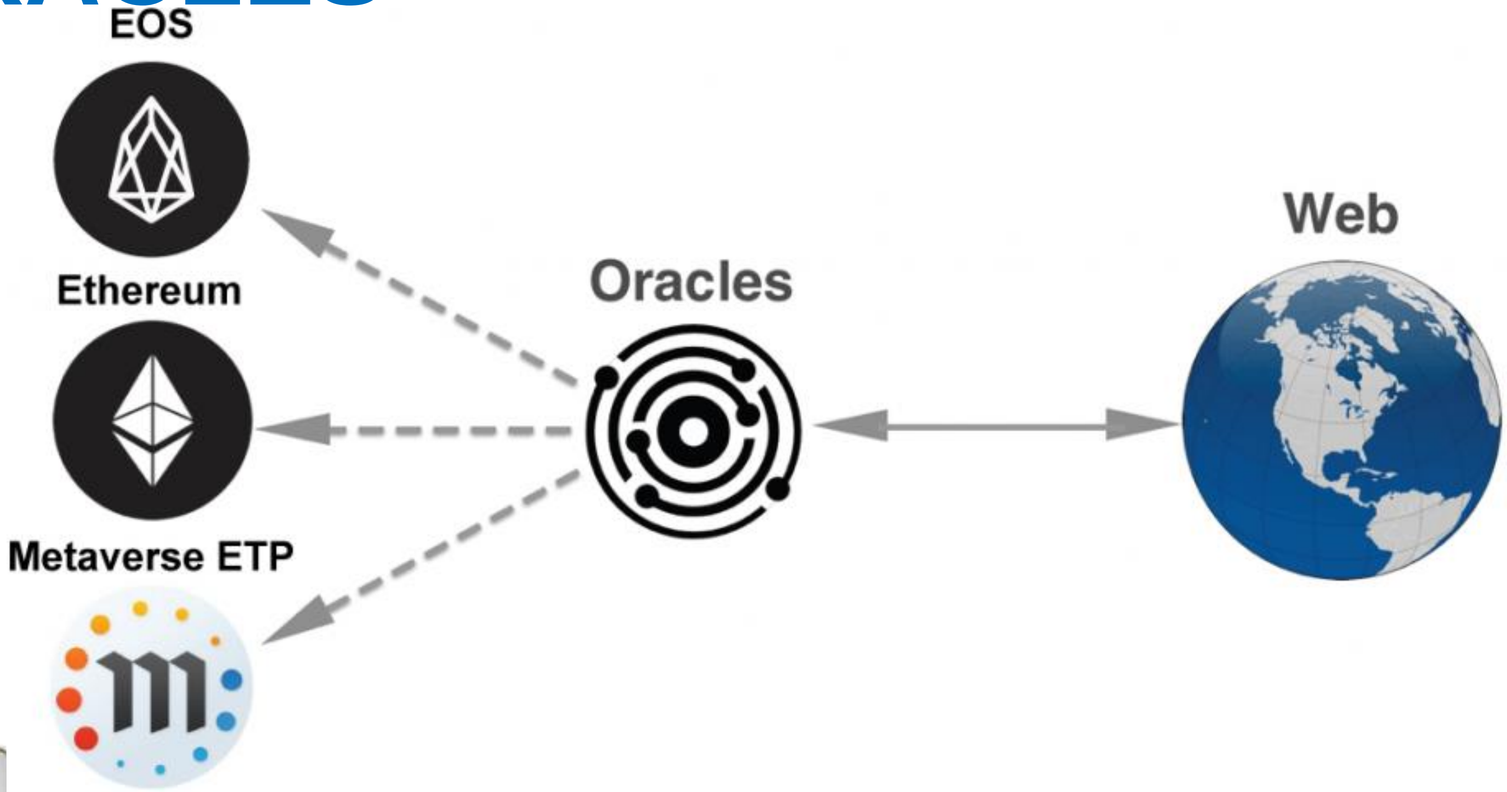
- Hardware
- Software
- Human

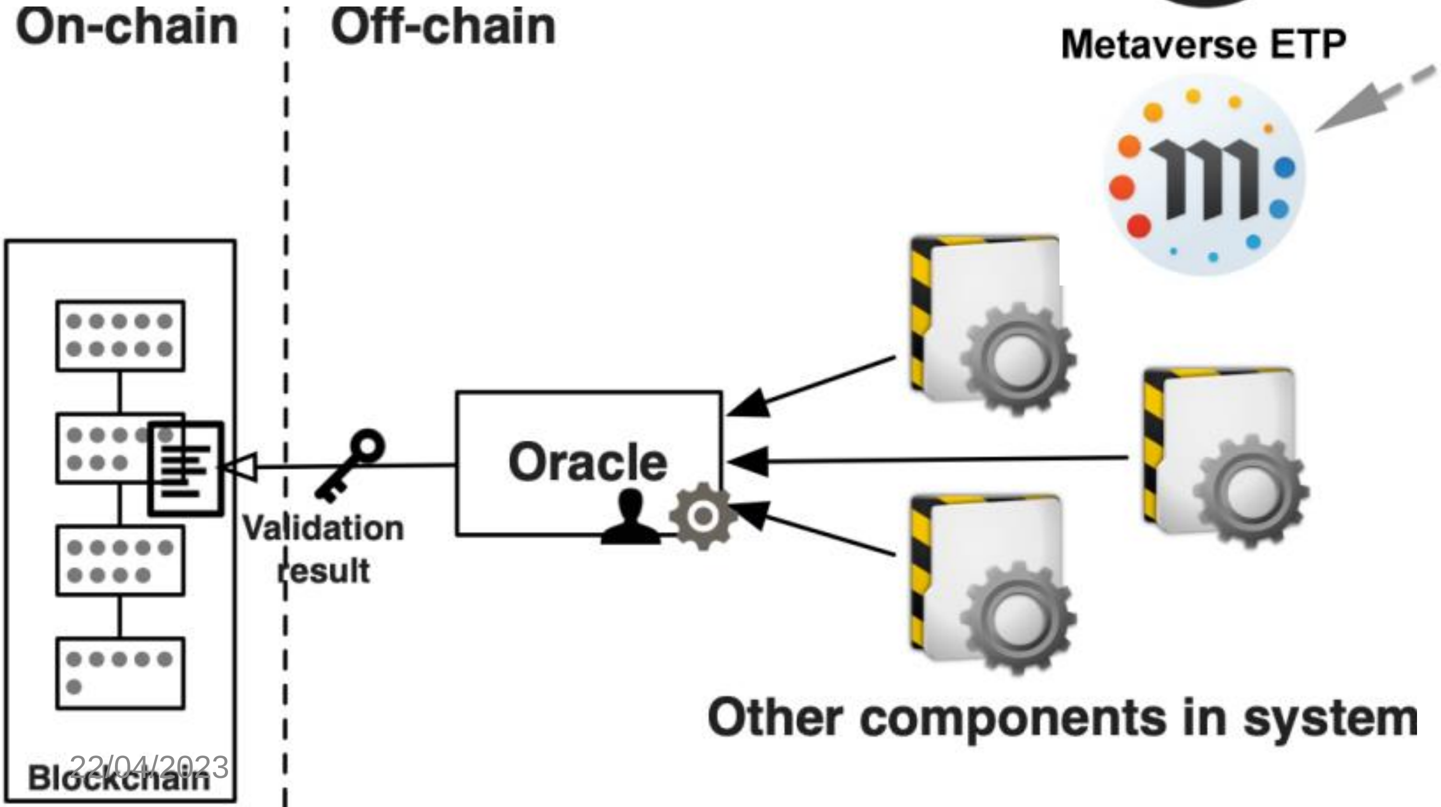


5.5. TYPE OF ORACLES

Data direction:

- input oracle
- output oracle

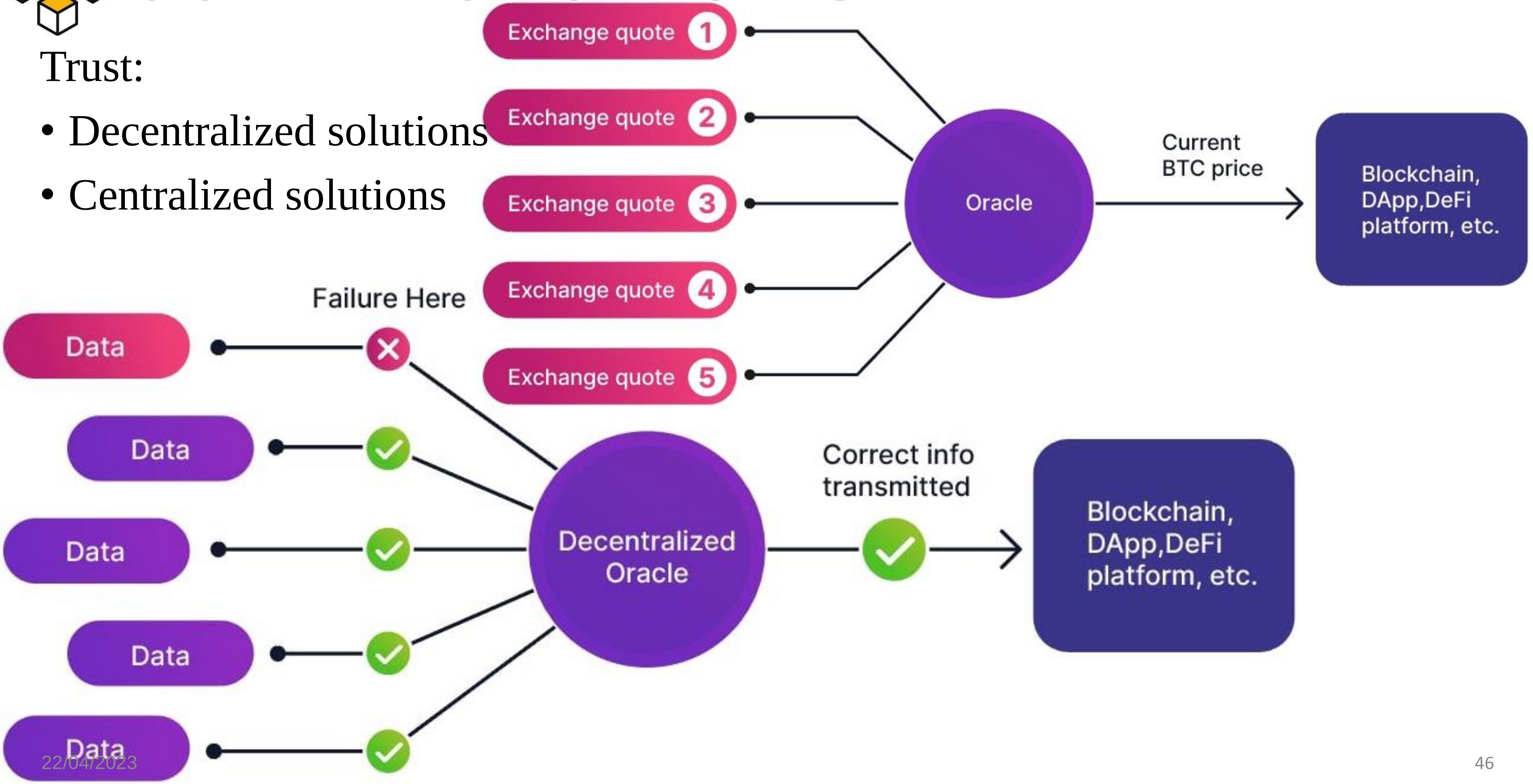




5.5. TYPE OF ORACLES

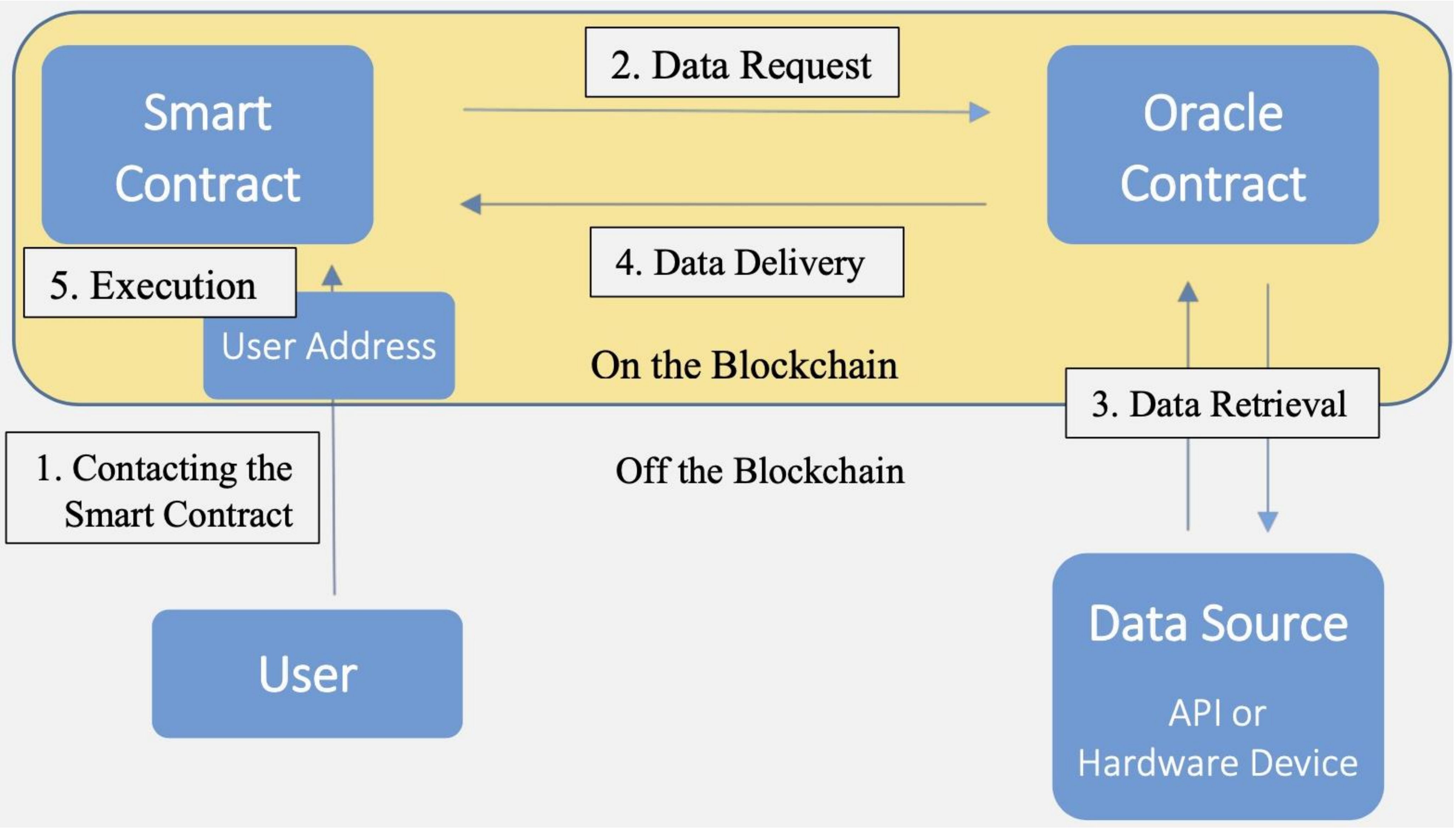
Trust:

- Decentralized solutions
- Centralized solutions



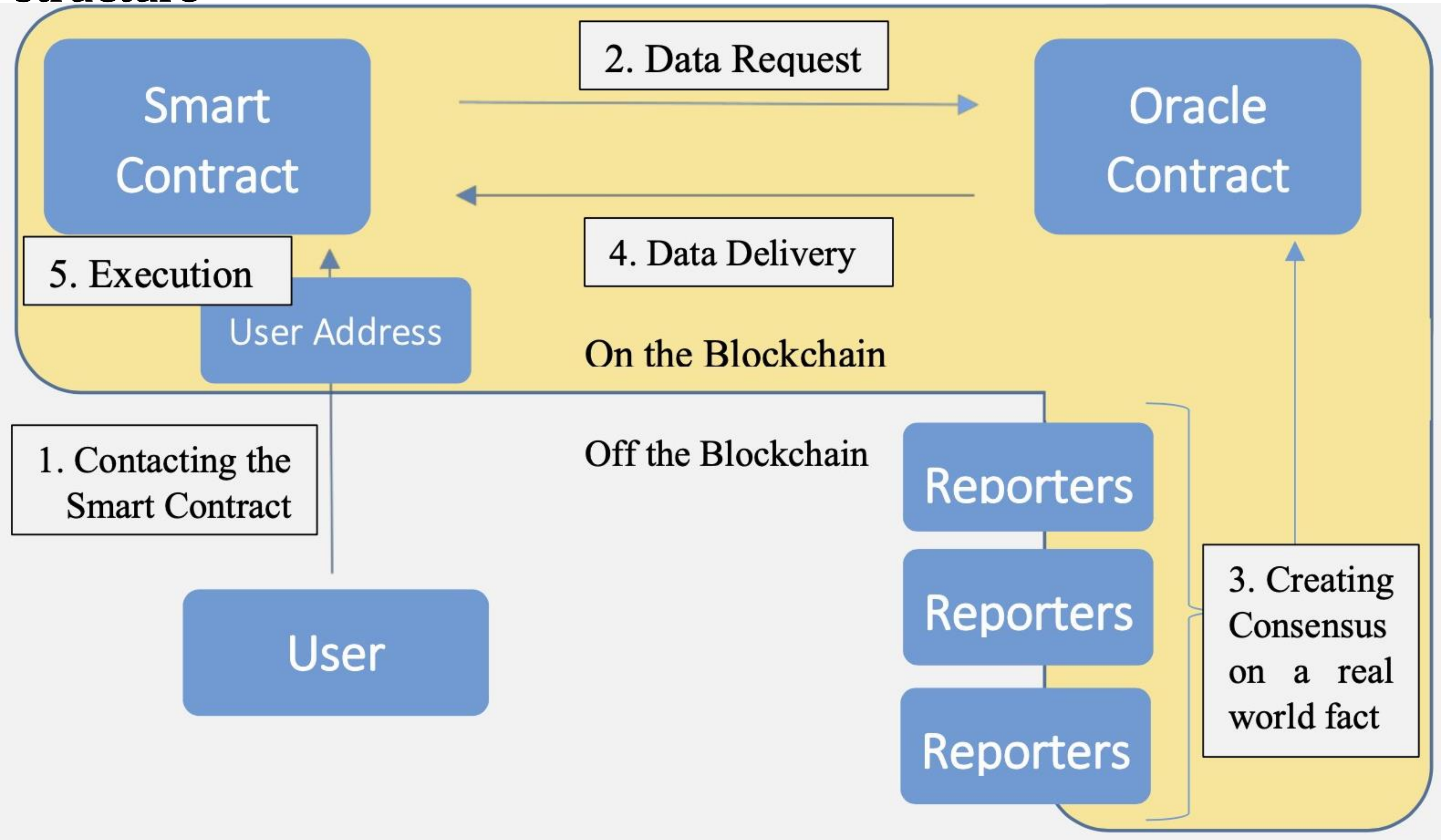
5.6. ORACLE STRUCTURE

- Automated oracle structure



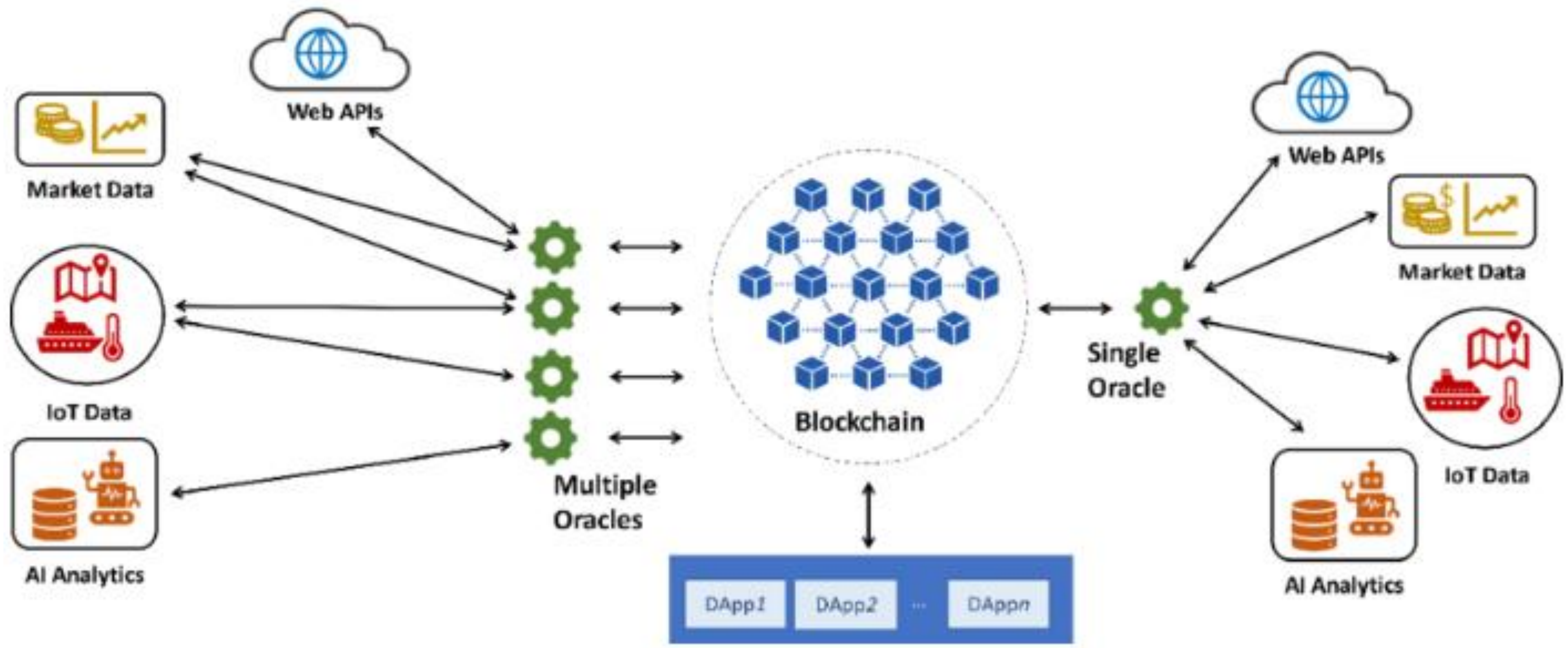
5.6. ORACLE STRUCTURE

- Human oracle structure



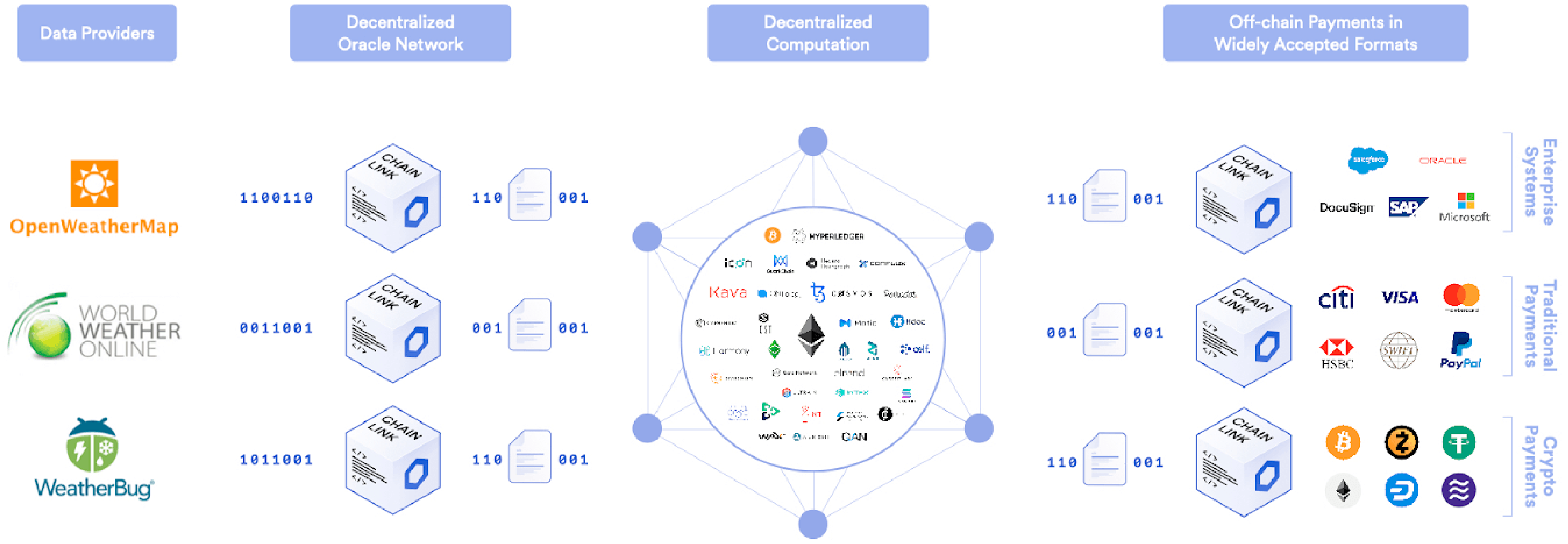
5.7. ORACLE SOLUTIONS

Scalability solutions:



5.7. ORACLE SOLUTIONS

Scalability solutions:



6. BLOCKCHAIN GOVERNANCE

6.1. INTRODUCTIONS

Governance:

- The system, entities are directed and controlled
- Structure, processes for decision making, accountability, control and behaviour at the top of an entity

Principles that dictate governance:

- Rulers: providing governance
- Rules
- Participants

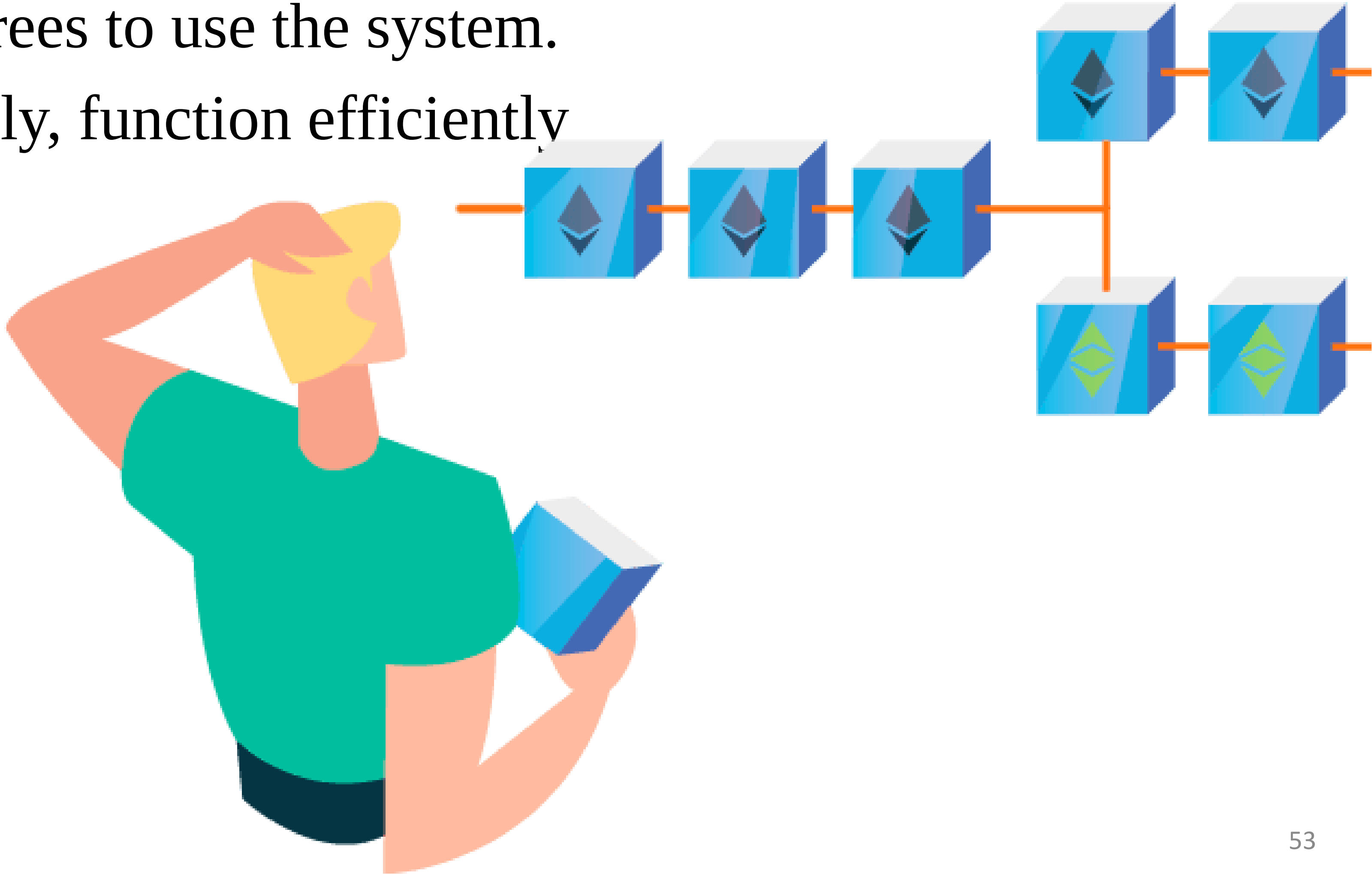
Governance Types: Standard

- Direct Governance:
- Representative Governance

6.1. INTRODUCTIONS

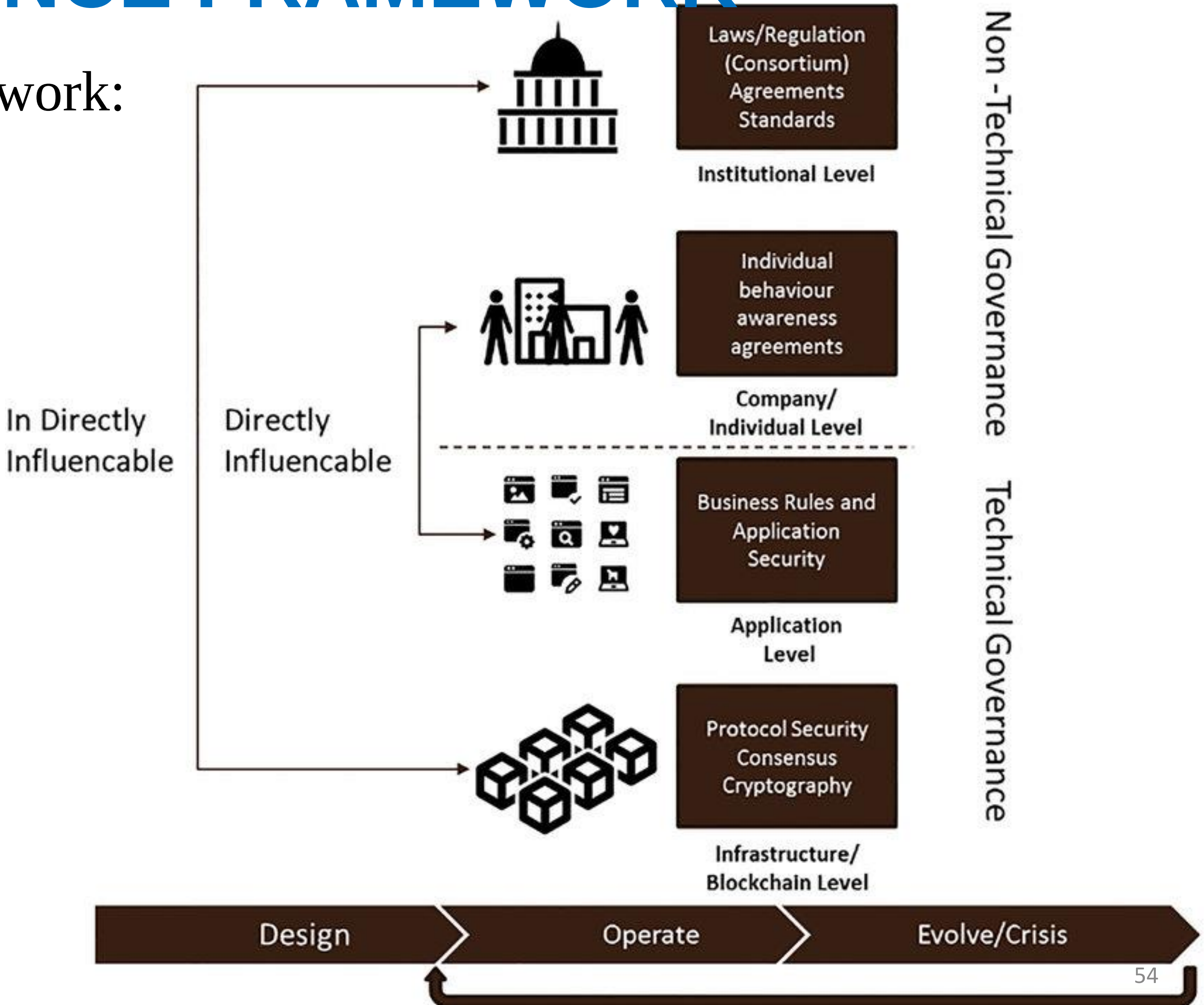
Blockchain governance:

- Mechanisms that decentralized networks adapt and change over time.
- Structure that participant agrees to use the system.
- Makes sure: works seamlessly, function efficiently



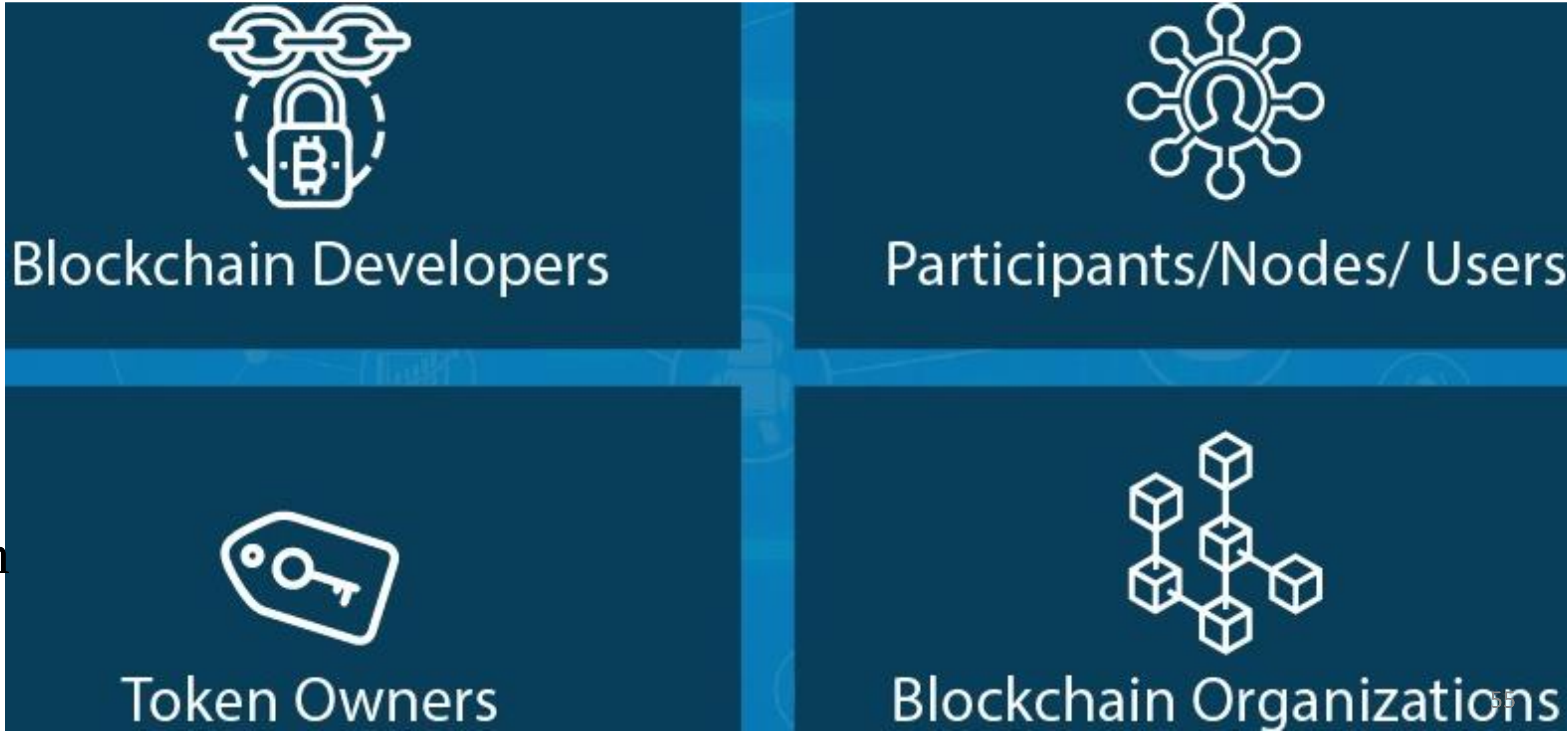
6.2. GOVERNANCE FRAMEWORK

Blockchain governance framework:



6.3. GOVERNANCE PARTICIPANTS

- Organization: founders of a blockchain, important decisions blockchain future. governing the core code of a blockchain
- Developers: maintaining and updating the fundamental code of a blockchain. change any feature in the blockchain
- Node operators: update and maintain the full ledger copy of a blockchain. differ from miners
- Token holders: community of token holders. making a decision concerning the entire ecosystem.

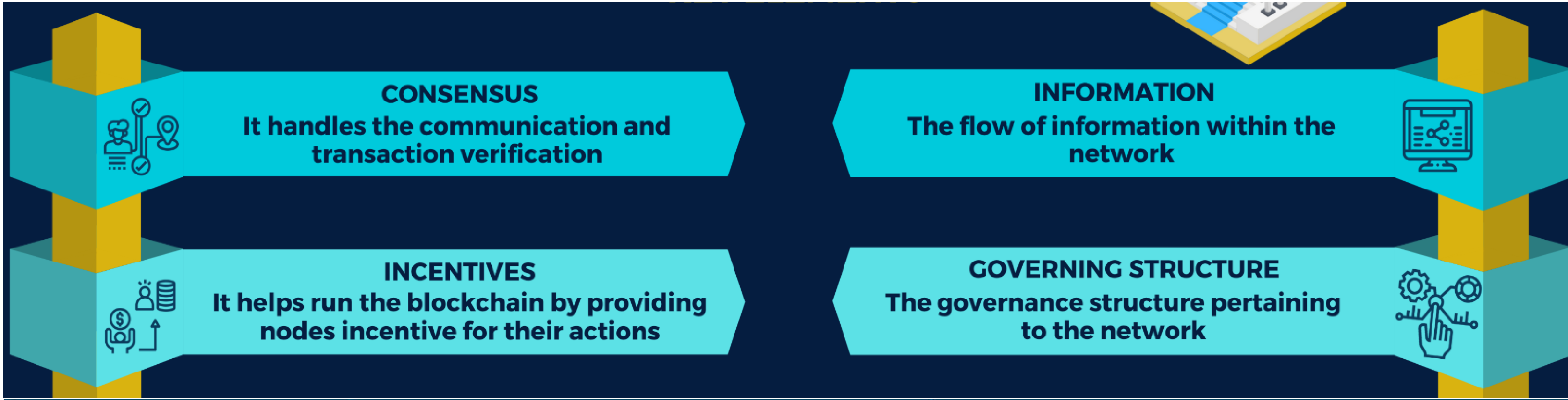


6.3. GOVERNANCE PARTICIPANTS

END-USER	NODE OPERATOR	INFRASTRUCTURE PROVIDER	BLOCKCHAIN-AS-A-SERVICE	SOLUTION PROVIDER
Consumes blockchain services rather than actively participates in the blockchain system. The operator may still provide services that leverage the benefits of blockchain.	Run and operate a node on the distributed ledger to verify records and processes. Could be a miner (cryptocurrency) or participant / steward.	Provide members in the blockchain ecosystem the underlying compute infrastructure to operate nodes and related processes.	Platforms for customers (e.g. developers) to create their own blockchain-enabled solutions. These are likely to include the compute infrastructure.	End-to-end solutions; third parties plug into the blockchain ecosystem and the telco provides governance and/or managed services.
IBM Telcos in India plan to leverage blockchain registries for Do Not Call	sovrin Swisscom, Spark and other CSPs are Sovrin stewards	Azure Platform for simplified deployment onto its compute infrastructure	Deutsche Telekom T-Labs is prototyping a blockchain-as-a-service platform – hashstaX	TELSTRA Blockchain and biometric security smart home solutions

6.4. GOVERNANCE RULES

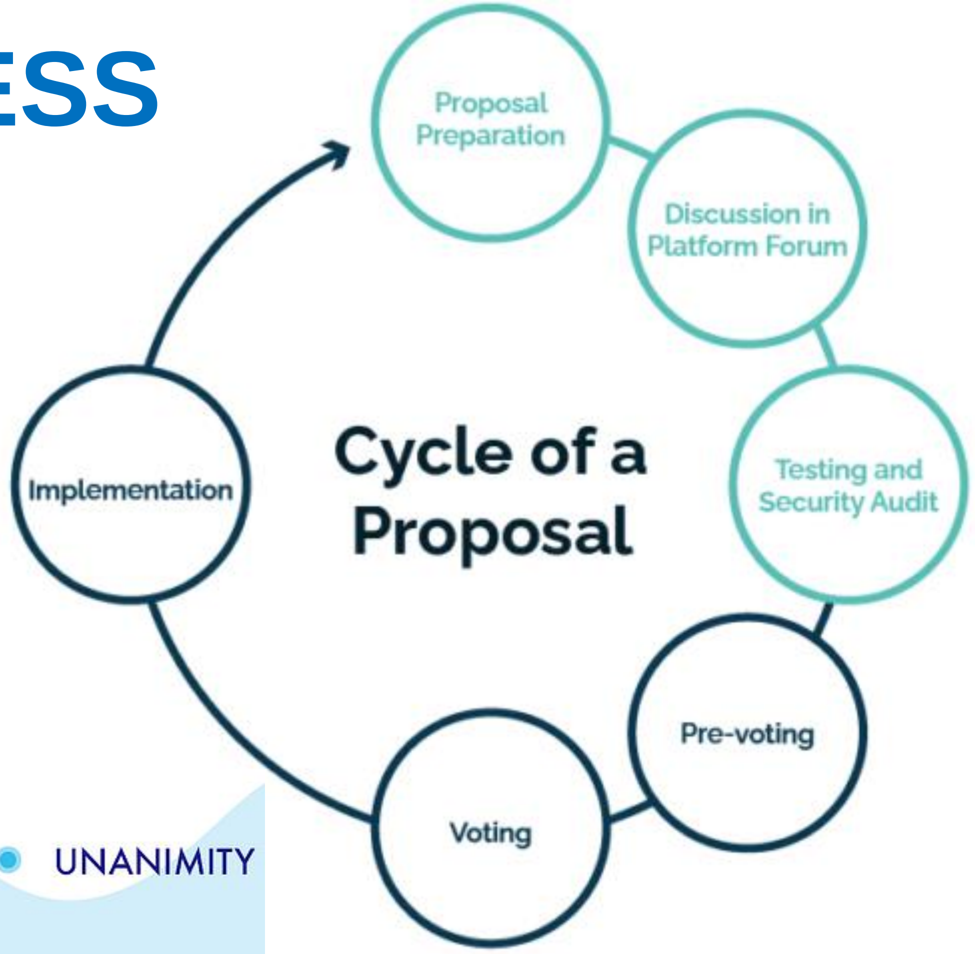
•

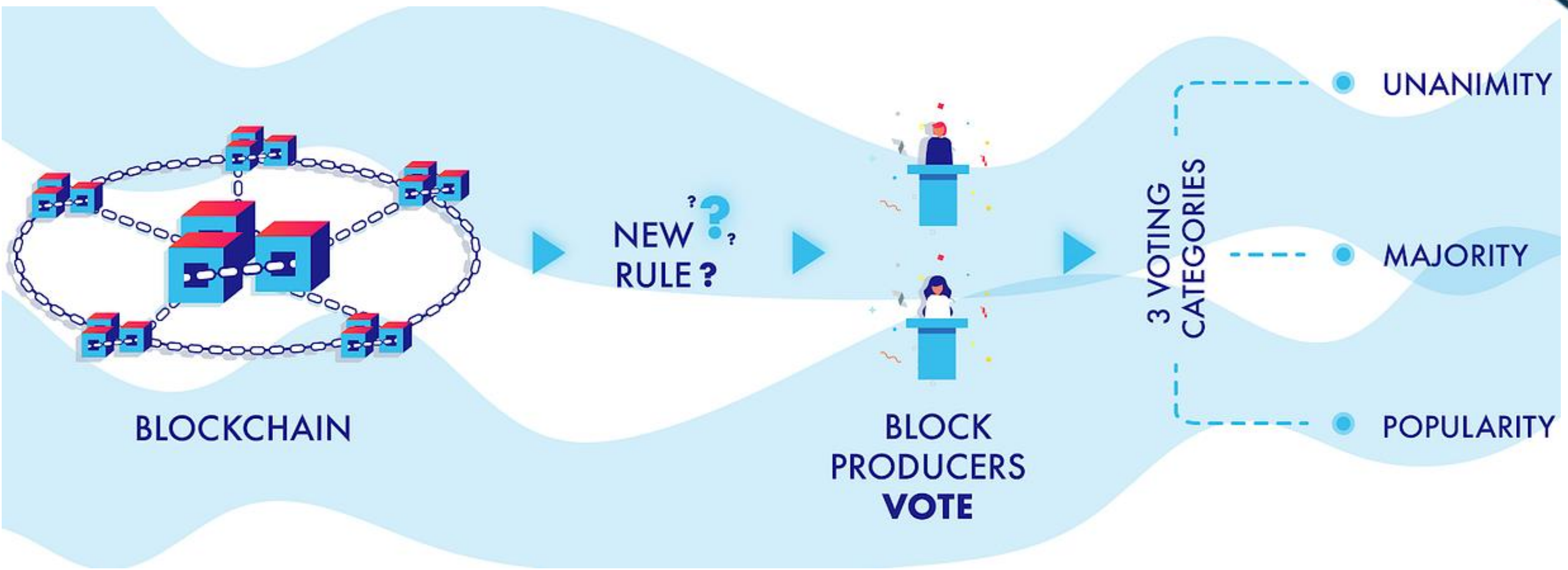


PARAMETERS	OFF CHAIN BLOCKCHAIN GOVERNANCE	ON CHAIN BLOCKCHAIN GOVERNANCE
Consensus	Leaders take decision after reaching a general consensus	Direct decentralized voting mechanism
Incentives	Mining fees as incentives	Disparate incentives
Information	Transparent & trustless	Transparency is higher than Off Chain
Governing Structure	Typically Centralized	Completely Decentralized

6.5. GOVERNANCE PROCESS

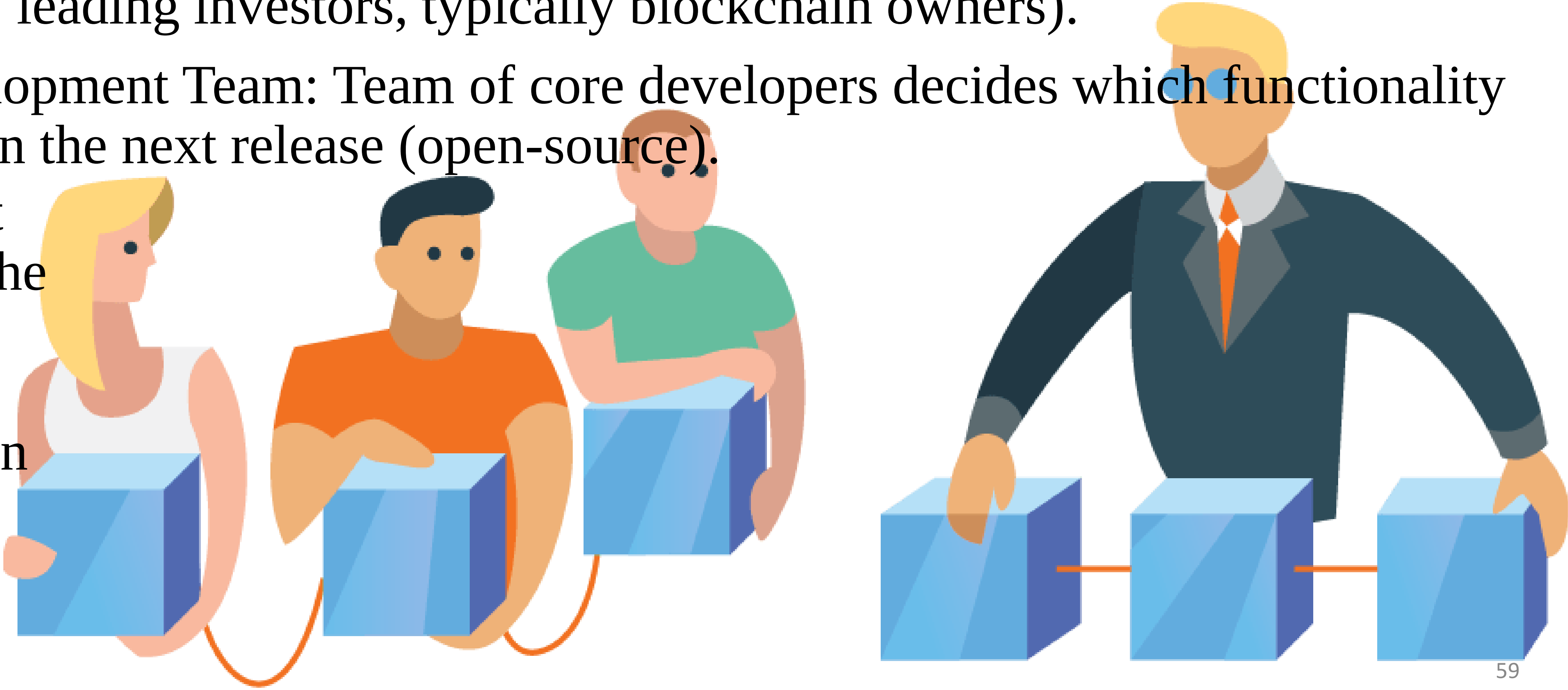
- Proposal Development (off-chain) stage
- Voting Process (on-chain) stage





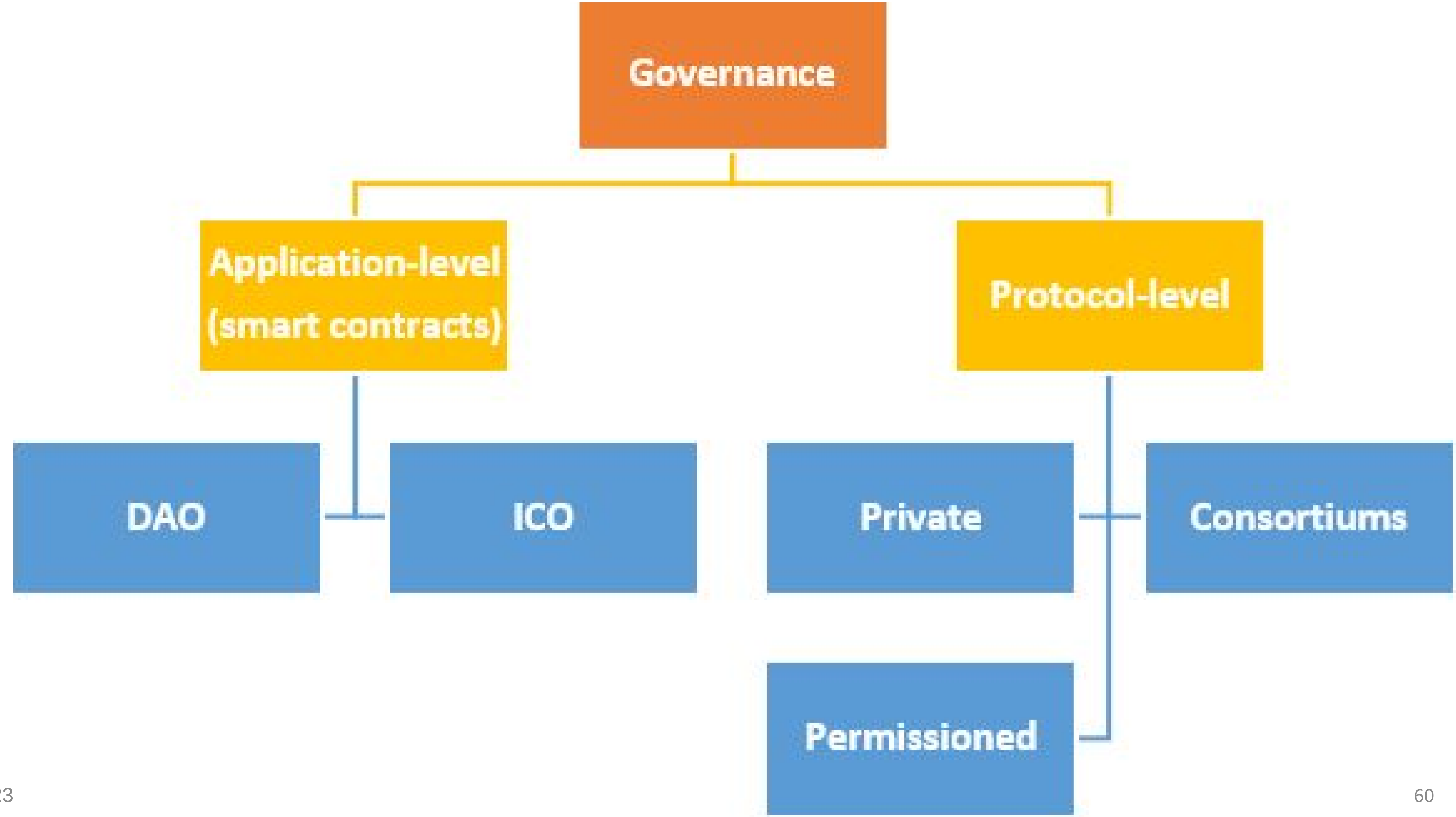
6.6. GOVERNANCE STATEGIES

- On-Chain Governance: rules written in smart contracts. All users make decisions about blockchain development. (direct)
- Open Governance: blockchain users choose team of decision-makers (core developers, leading investors, typically blockchain owners).
- Core Development Team: Team of core developers decides which functionality to include in the next release (open-source).
- Benevolent Dictator: The creator of blockchain has made on all the decisions



6.7. BLOCKCHAIN DEVELOPERS

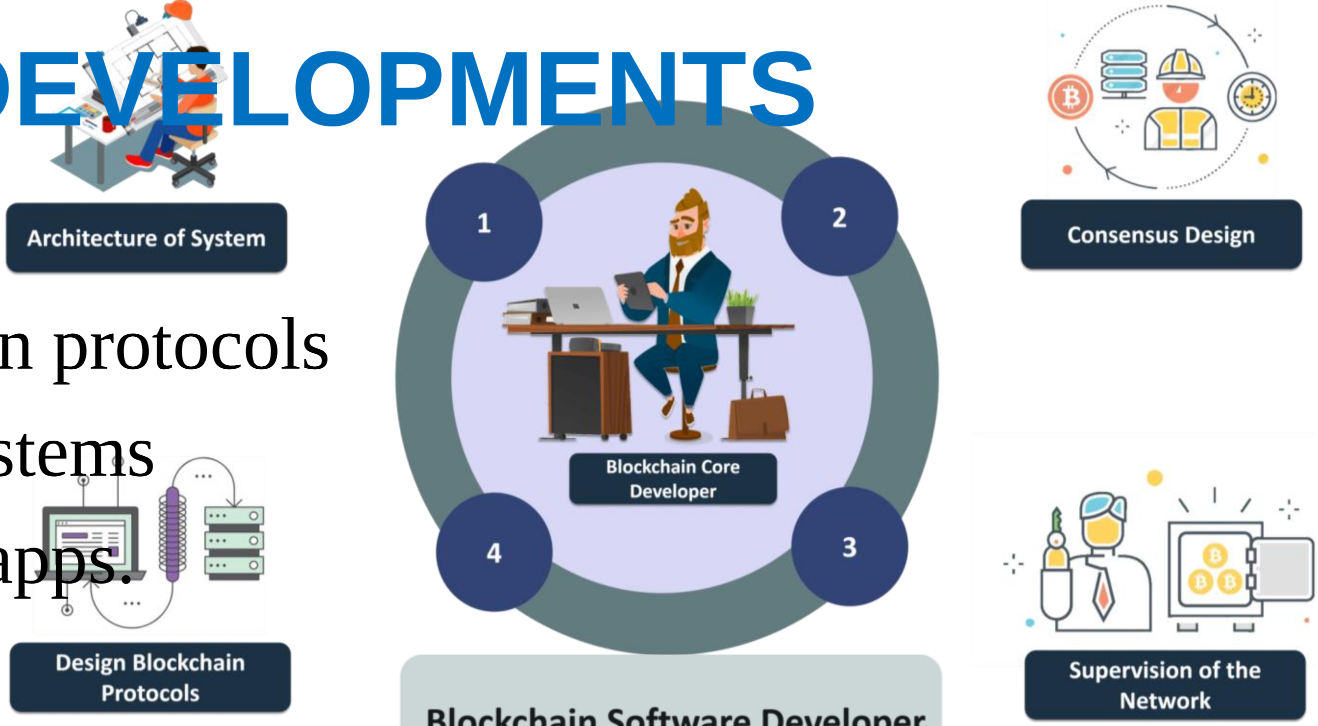
•

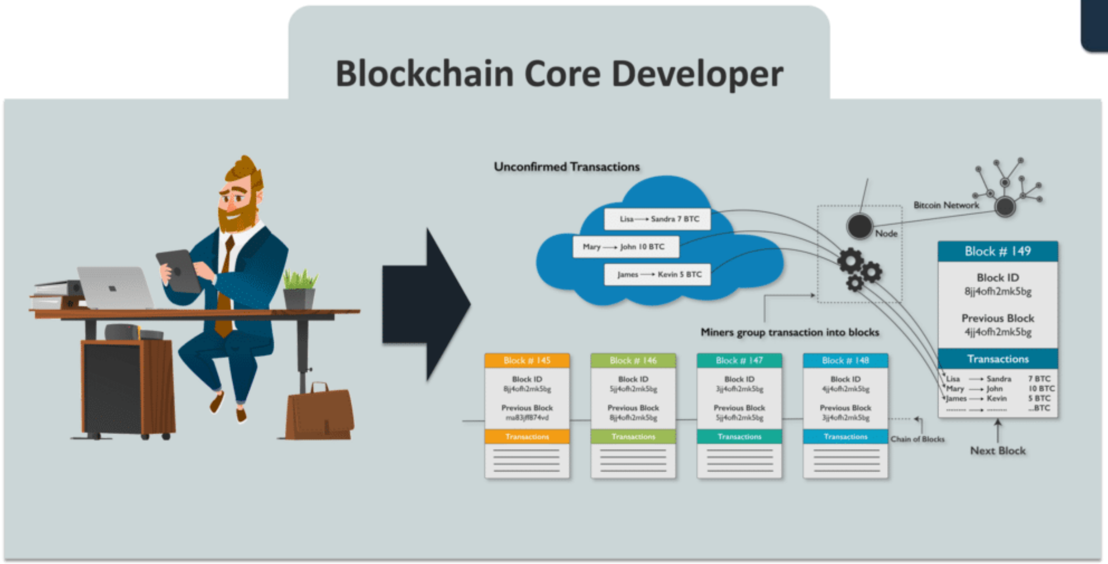


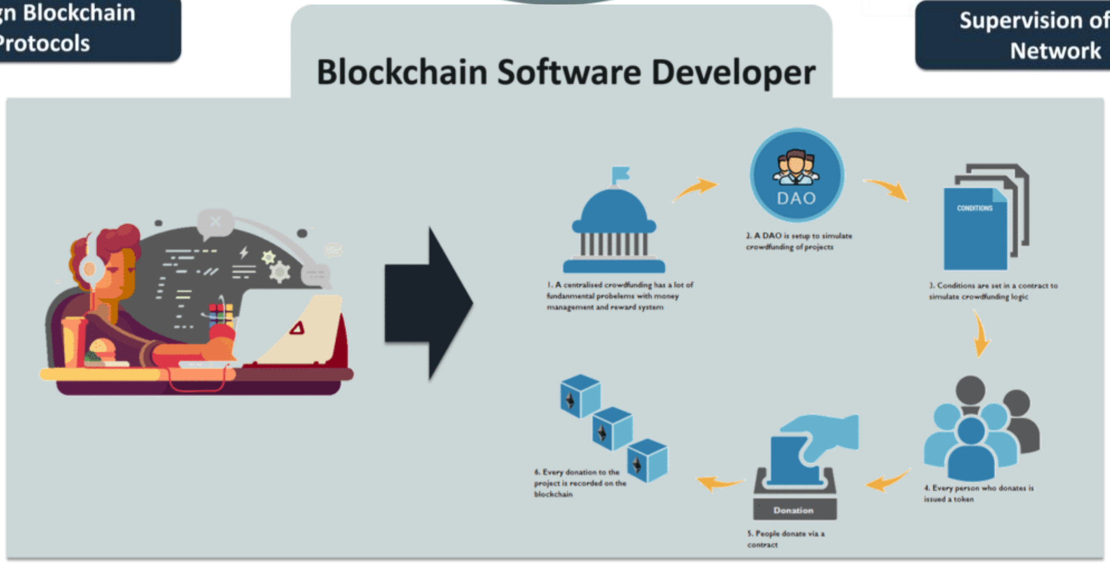
6.7. BLOCKCHAIN DEVELOPMENTS

Blockchain developers:

- Developing and optimizing blockchain protocols
- crafting architecture of blockchain systems
- Developing smart contracts and web apps.





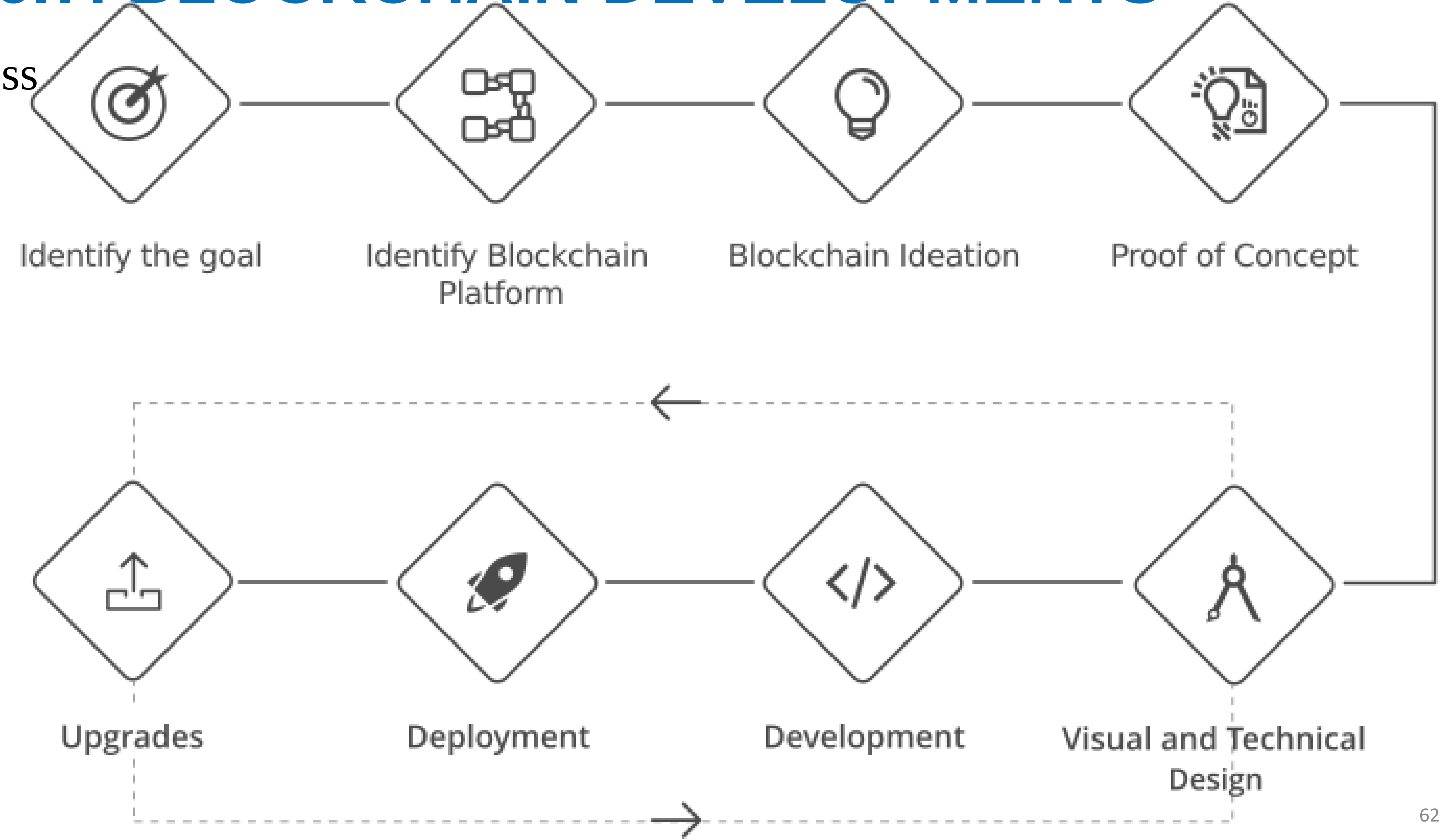


Core blockchain developer handles the design architecture of blockchain technology

Blockchain software developer use this architecture to create blockchain applications

6.7. BLOCKCHAIN DEVELOPMENTS

Process



7. DISTRIBUTED LEDGER TECHNOLOGY

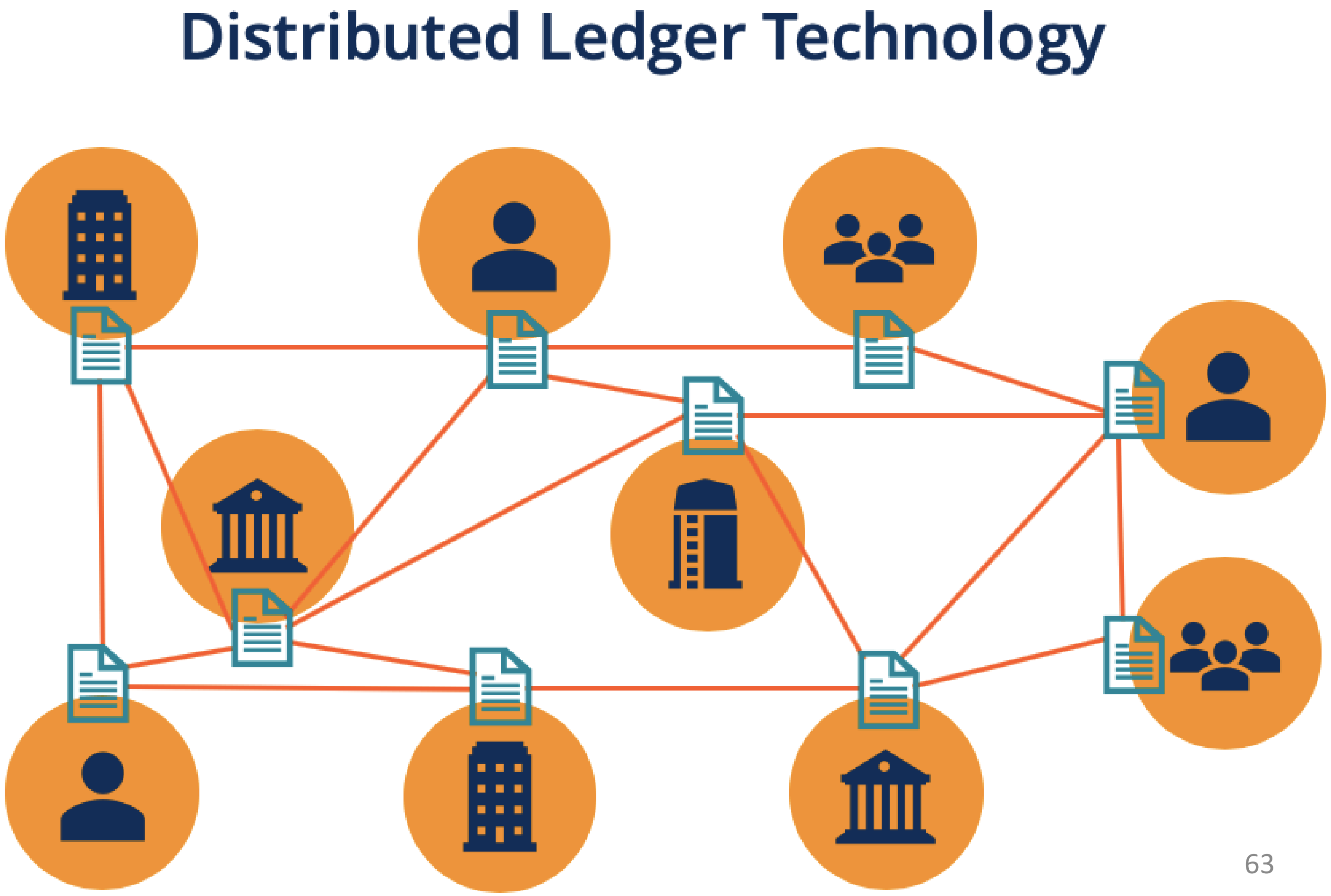
7.1. DISTRIBUTED LEDGER TECHNOLOGY

7.2. DIRECTED ACYCLIC GRAPH

7.3. HASHGRAPH

7.4. HOLOCHAIN

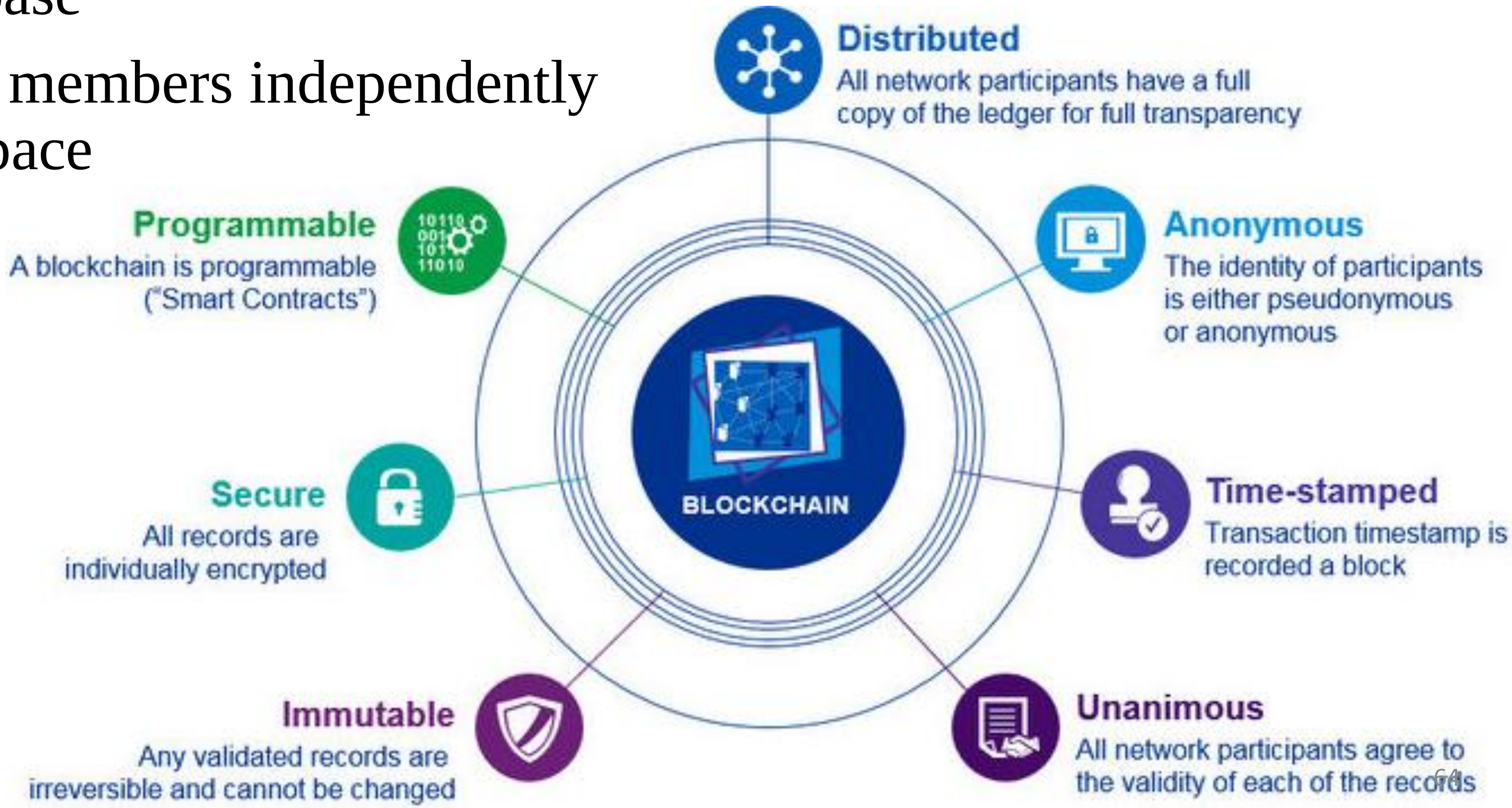
7.5. RADIX



7.1. DISTRIBUTED LEDGER TECHNOLOGY

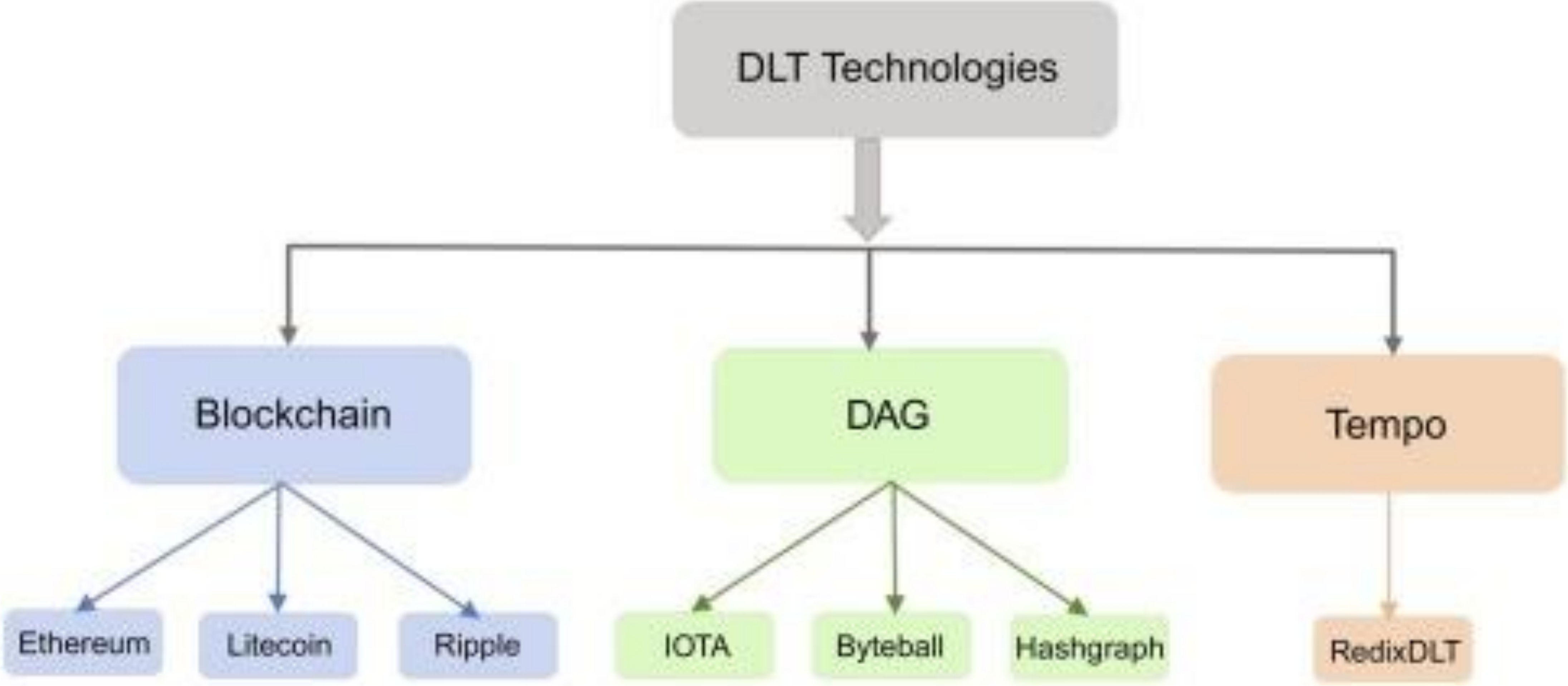
Distributed ledger techonology:

- digital system that records transactions
- form of digital database
- updated and held by members independently in a large network space



7.1. DISTRIBUTED LEDGER TECHNOLOGY

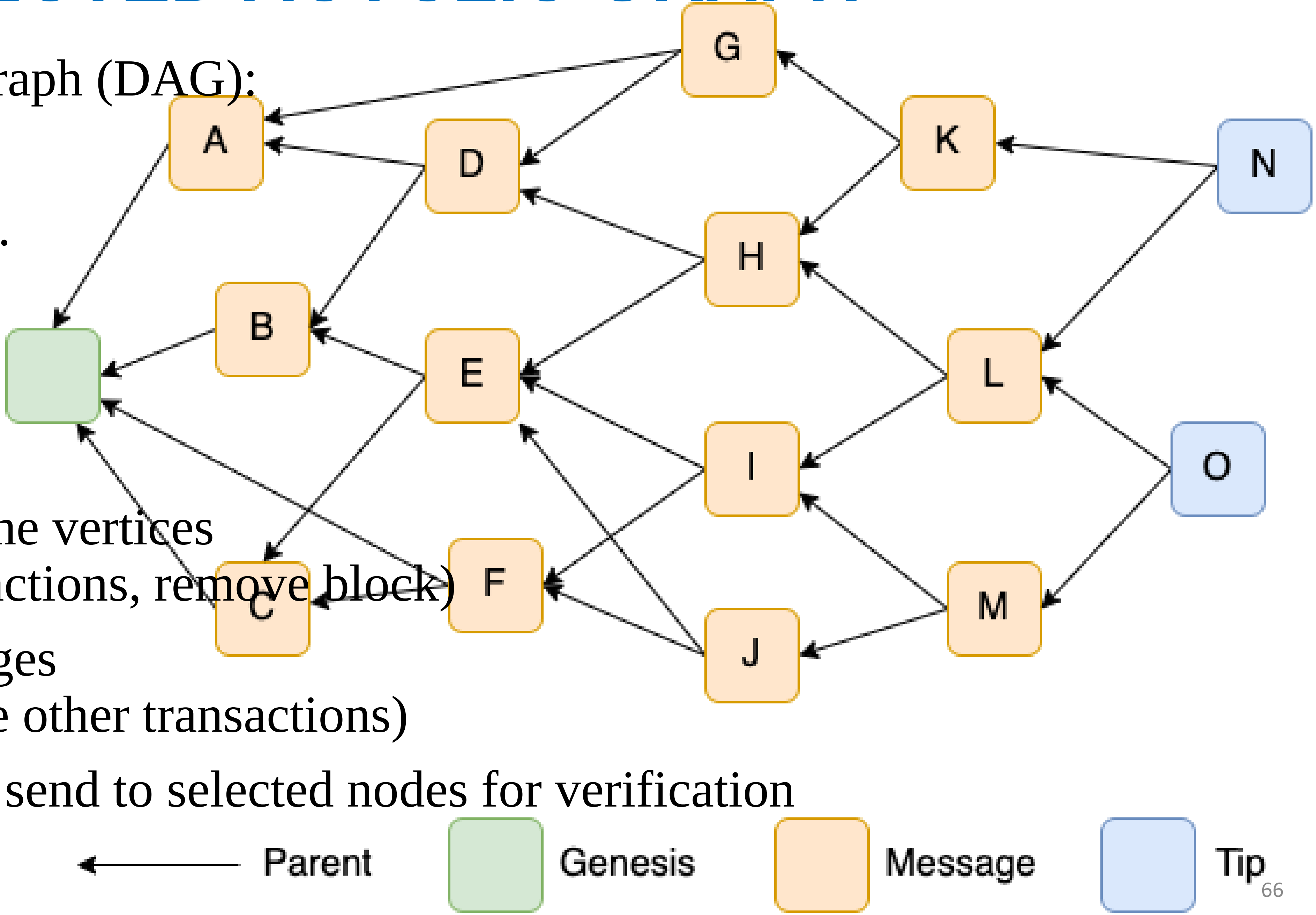
•



7.2. DIRECTED ACYCLIC GRAPH

Directed Acyclic Graph (DAG):

- directed graph
- no directed cycles.



DAG DLT

- Transactions are the vertices
  (network of transactions, remove block)
- Hashes are the edges
  (linked to multiple other transactions)
- Transactions only send to selected nodes for verification

7.2. DIRECTED ACYCLIC GRAPH

Advantages:

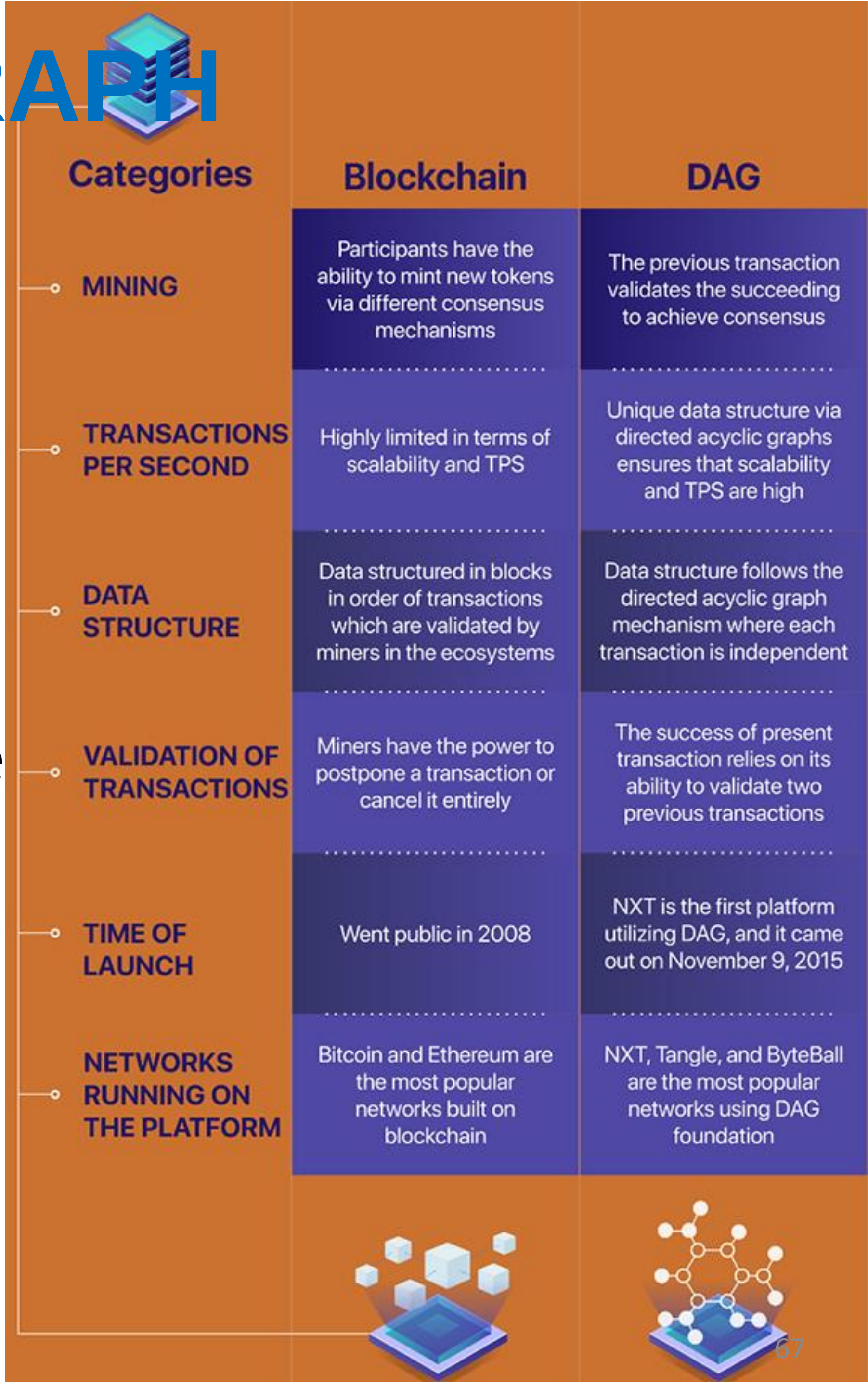
- process transactions instantly (fast)
- Remove miners (fast and free)
- greater scalability

Problems:

- Not limit number of transactions, registry grow too large, need third parties verify transactions
- shorter confirmation times are really irrelevant, because what really matters are the finality guarantees
- not fully decentralized. Hedera Hashgraph, IOTA are centralized.

Conclusion:

- could be interesting if decentralization is not a priority.
- not sufficiently tested and may be insecure.



Categories	Blockchain	DAG
MINING	Participants have the ability to mint new tokens via different consensus mechanisms	The previous transaction validates the succeeding to achieve consensus
TRANSACTIONS PER SECOND	Highly limited in terms of scalability and TPS	Unique data structure via directed acyclic graphs ensures that scalability and TPS are high
DATA STRUCTURE	Data structured in blocks in order of transactions which are validated by miners in the ecosystems	Data structure follows the directed acyclic graph mechanism where each transaction is independent
VALIDATION OF TRANSACTIONS	Miners have the power to postpone a transaction or cancel it entirely	The success of present transaction relies on its ability to validate two previous transactions
TIME OF LAUNCH	Went public in 2008	NXT is the first platform utilizing DAG, and it came out on November 9, 2015
NETWORKS RUNNING ON THE PLATFORM	Bitcoin and Ethereum are the most popular networks built on blockchain	NXT, Tangle, and ByteBall are the most popular networks using DAG foundation

7.2. DIRECTED ACYCLIC GRAPH

DAG – DLT platforms:

- Taraxa
- IOTA
- Nano
- Corda
- Hashgraph

7.3. HASHGRAPH

Hedera Hashgraph:

- enterprise DAG-DLT
- data structure and consensus algorithm
- provides a new platform for distributed consensus

Protocols:

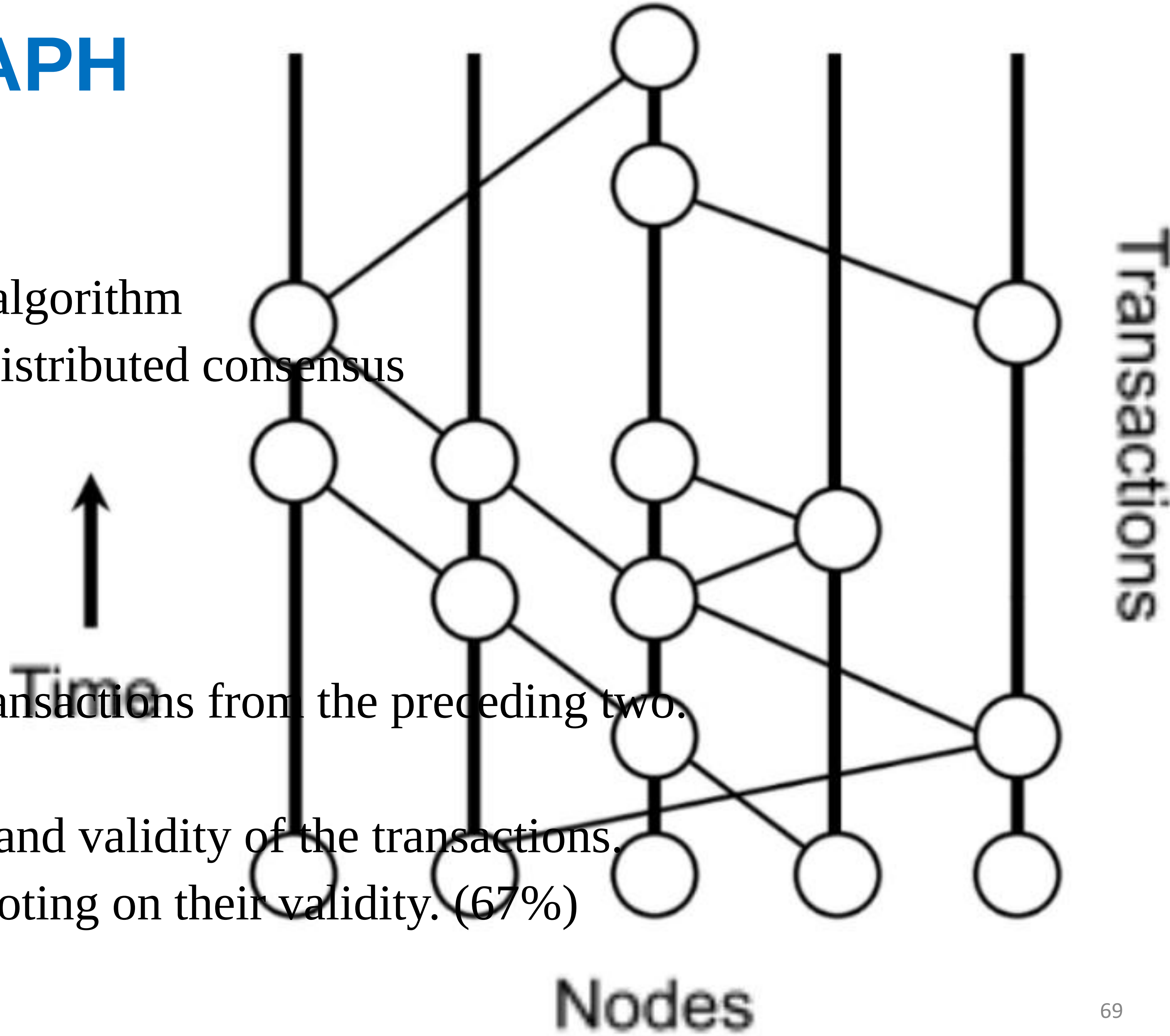
- Gossip about Gossip
- Virtual Voting

Transactions:

- events keep the hashes and transactions from the preceding two.

Network:

- voting algorithm for security and validity of the transactions.
- Nodes witness transactions, voting on their validity. (67%)



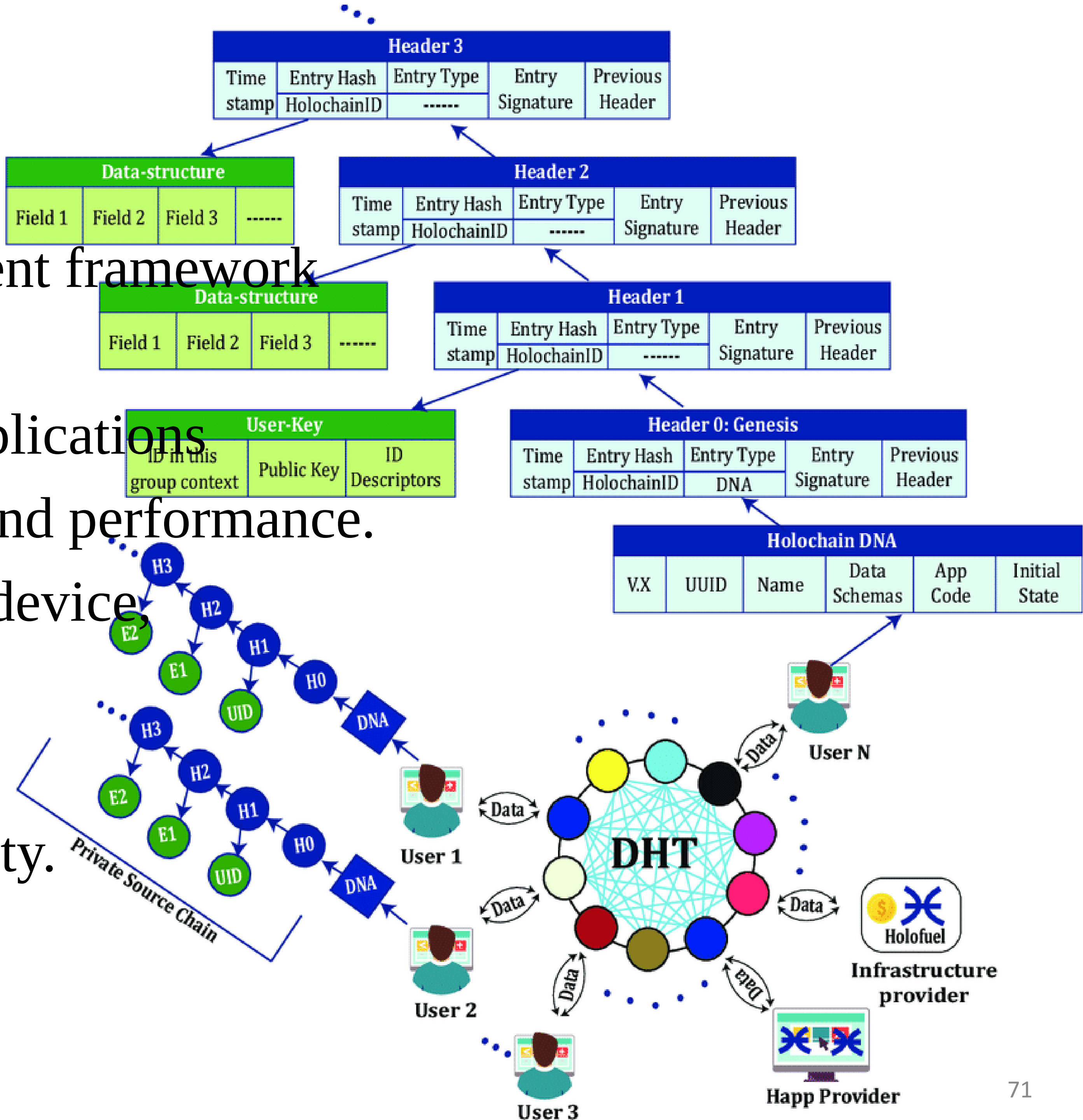
7.3. HASHGRAPH

	Blockchain	Hashgraph
Copyright	Open Source	Patented
Consensus	Proof of Work, Proof of Stake, Practical Byzantine Fault Tolerance, Proof of Elapsed Time	Virtual Voting
Security Mechanism	Cryptographic Hashing	Asynchronous Byzantine Fault Tolerance
Applications	Bitcoin, Ethereum, Hyperledger Blockchain Projects, EOS	Swirlds
Speed	100 to 1000 transactions based on the protocol implementation like ethereum, hyperledger etc.	500,000 transactions per second
Asynchronous Byzantine Fault Tolerance	Few implementations in Hyperledger are Byzantine Fault tolerance ready.	100% compliance

7.4. HOLOCHAIN

Holochain:

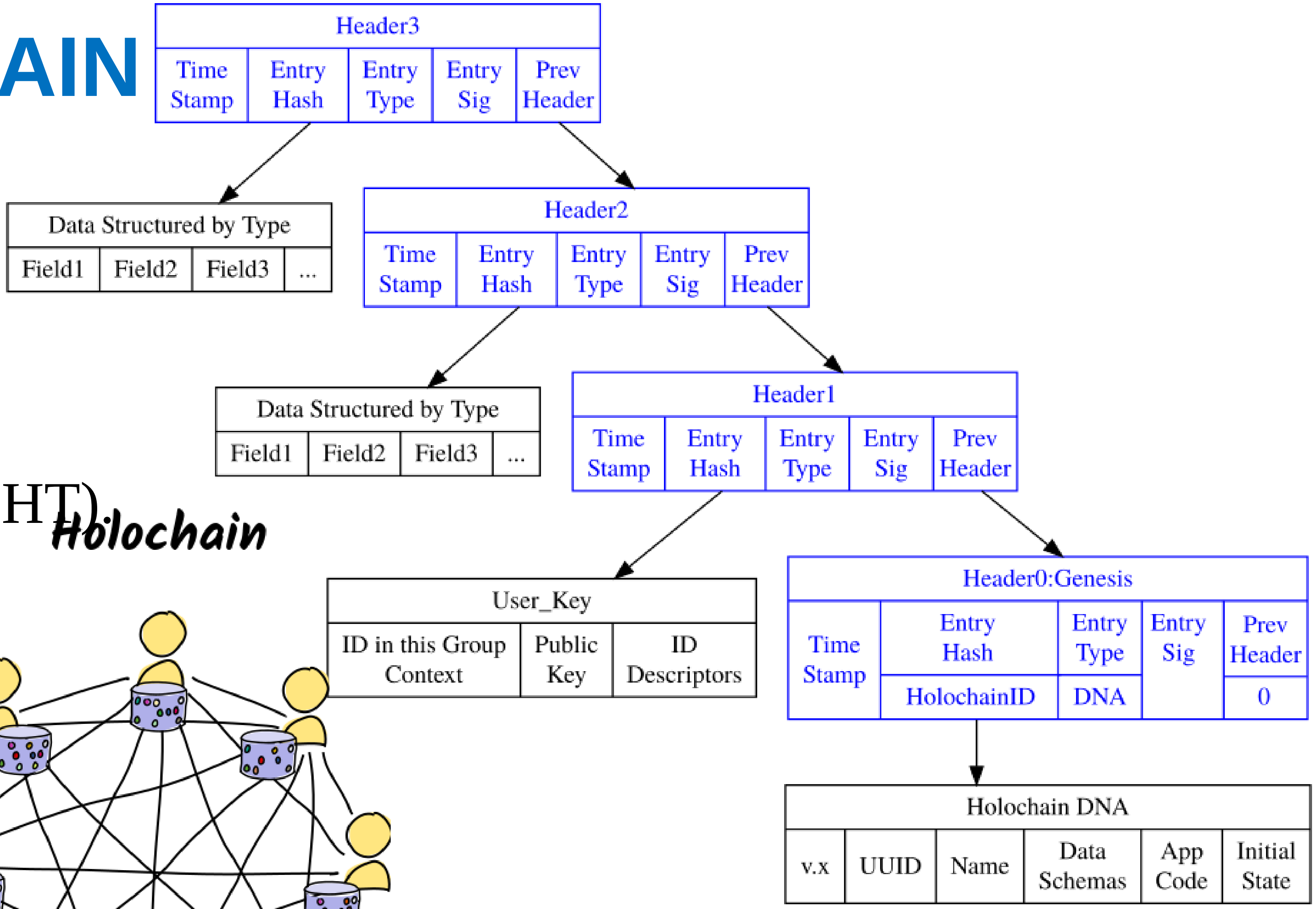
- open-source application development framework
- peer-to-peer networking protocol
- allows to create truly serverless applications
- high levels of security, reliability, and performance.
- user runs application on their own device,
- creates and stores their own data
- talks directly to other users
- cryptography and peer accountability.

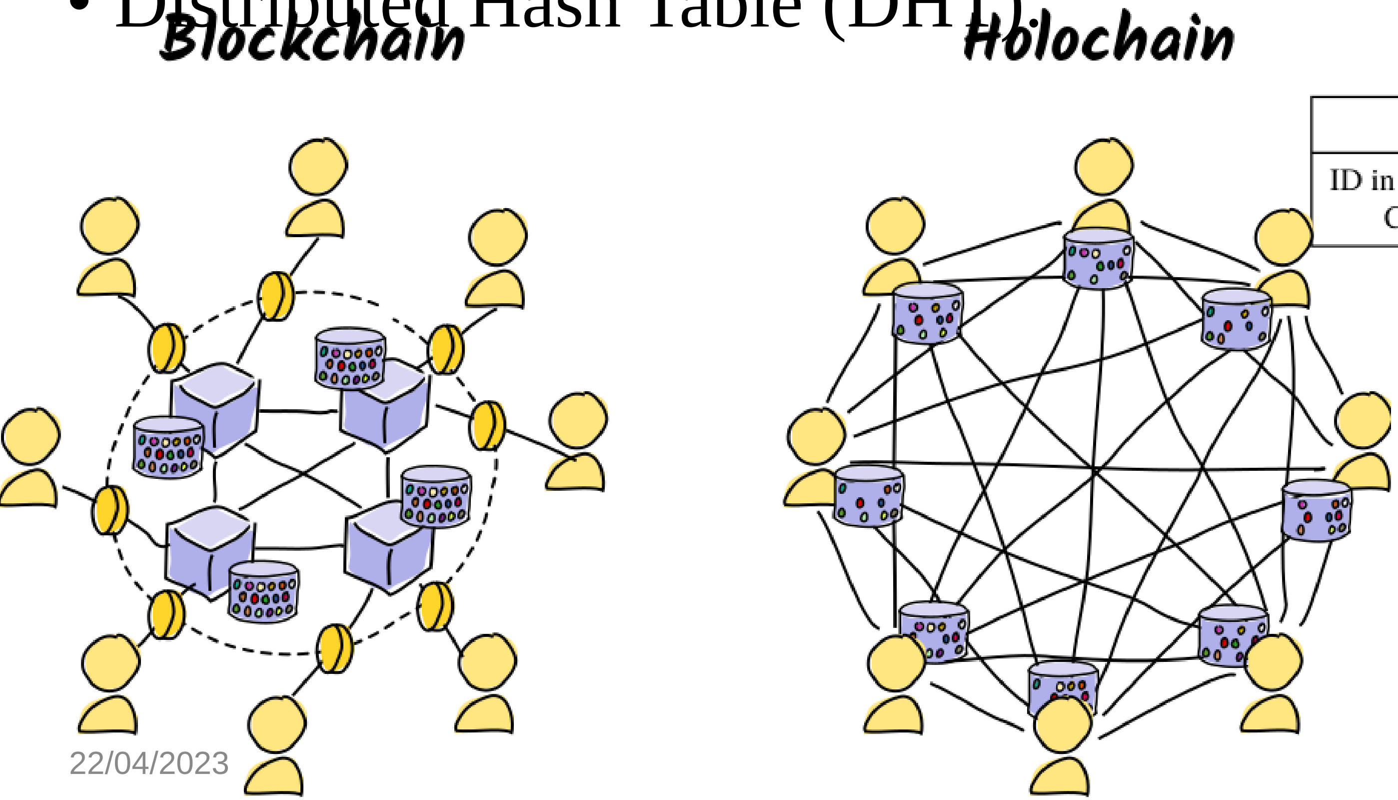


7.4. HOLOCHAIN

Holochain is composed of different technology:

- Hashchains
- Cryptographic singing
- Distributed Hash Table (DHT)





7.4. HOLOCHAIN

Holochain dApp Architecture:

- Shared storage (DHT)
- Application (Nucleus)
- Source Hash Chain

	blockchain	Holochain
Hash-chain approach	Data-centric, a single global data set - one shared reality across all nodes.	Agent-centric, allows nodes to act independently, or in tight coordination only with counterparties, and then share independently evolving data realities that come to agreement over time.
Energy Use	Bitcoin consumes more than 0.1% of the world's electricity[3][4] to power less than 0.0001% of the world's money.	Since no mining is required, no specialized processors[5] are needed, making it feasible to run full nodes on low-power computers or cell phones.
Transaction Volume	Neo currently processes +1000 transactions per second. Bitcoin and Ethereum considerably less at a handful per second.	Expected to surpass financial exchange backbones like the Visa network which has a max of 56,000 transactions per second.
Scalability	Even ignoring proof-of-work, there are serious scalability limits on synchronizing a global ledger across many nodes.[6]	With a sharded DHT, the transaction load per node gets lighter as the network grows[7].
Platform	Can now only run effectively with special mining rigs or wasteful staking algorithms.	Can run on a Raspberry Pi or a mobile phone.
Computational efficiency of architecture (not 1 machine)	O(n*m) for validating transactions on blockchain as a whole distributed architecture.	O(n/m*log m) for validating transactions.
Consensus Effects	Core consensus algorithms centralize power (make the rich richer). Proof-of-Work results in infinitely growing computational overhead for finite data set.	No mining or staking. No consensus. Not vulnerable to majority attacks. You only have to trust the code on your own node and can validate counterparty's history directly.

7.5. RADIX

Radix is:

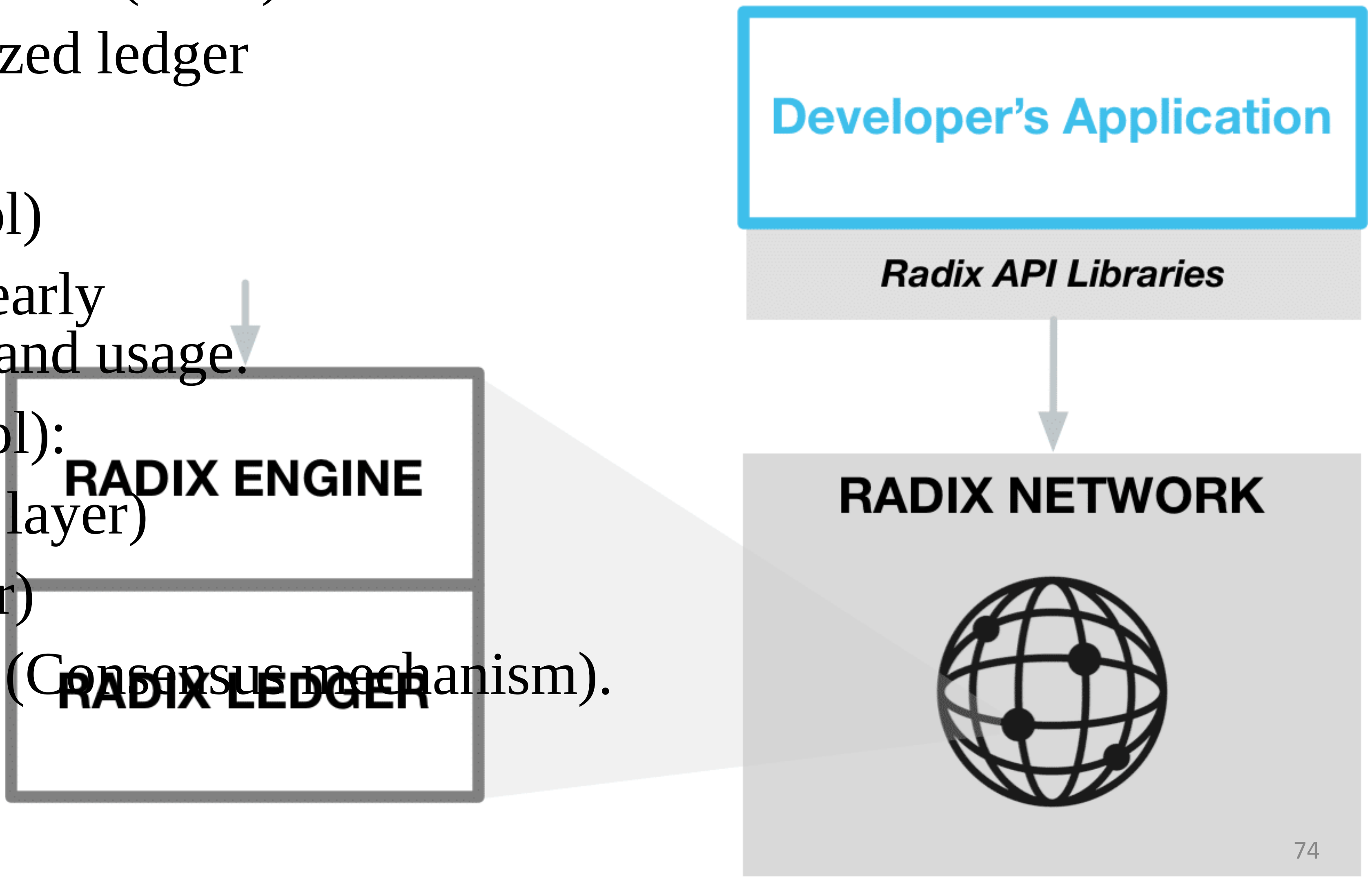
- layer-1 decentralized protocol (DeFi).
- public trustless decentralized ledger
- not yet open-source

Tempo (Consensus Protocol)

- can make Radix scale linearly with expanding platform and usage.

Radix components (protocol):

- Radix Engine (Execution layer)
- Cerberus (consensus layer)
- Delegated Proof of Stake (Consensus mechanism).



7.5. RADIX

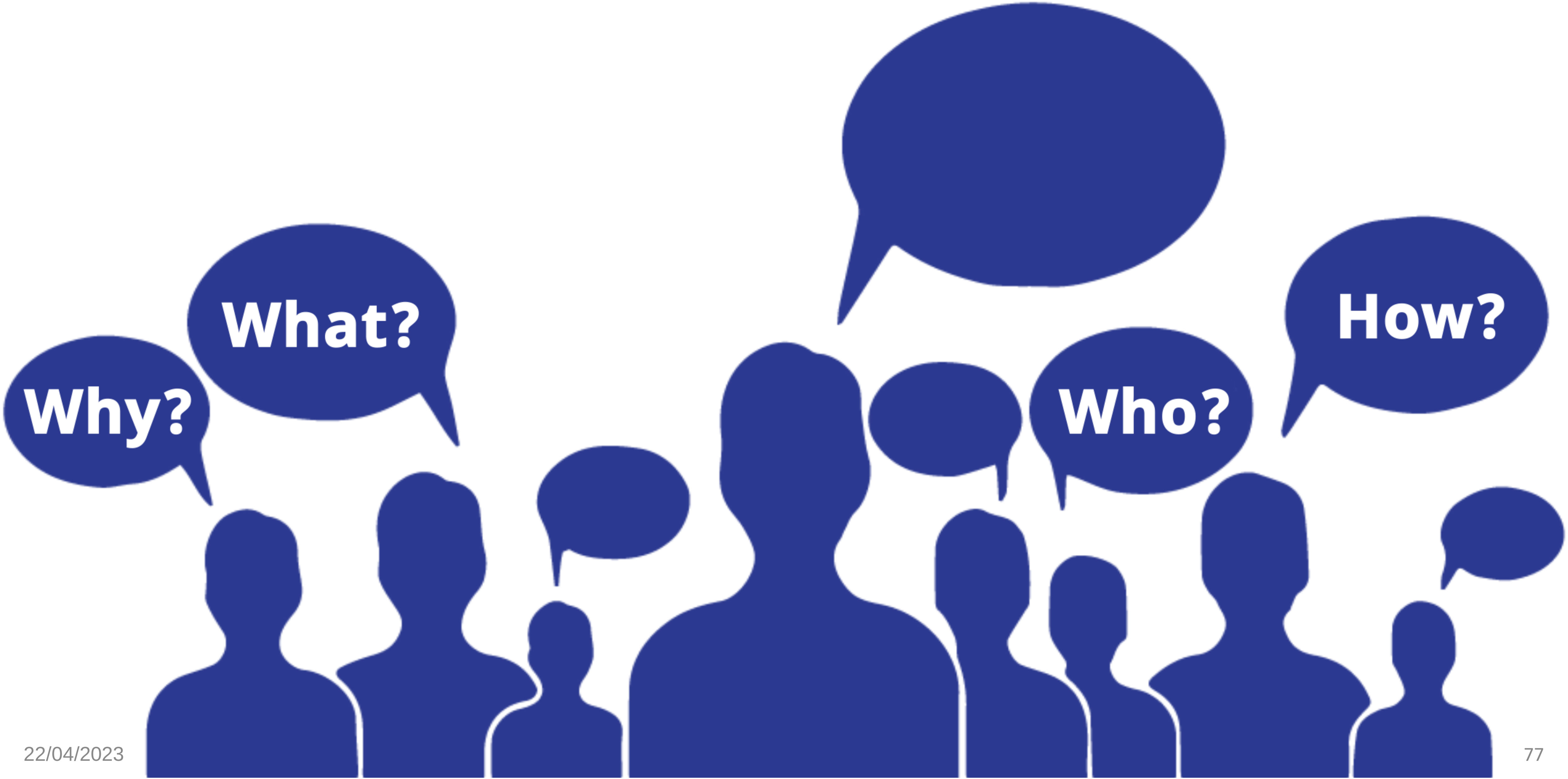

Smart Money Apps
Radix is building an open, interconnected platform where the full range of powerful DeFi applications will be built securely and safely.
Stable Coins
Collateralized Lending
Perpetual Futures
Decentralized Exchanges
Wallets & Dashboards
Money Markets
Yield Farming
Options & Derivatives
NFTs
Gaming
DeFi Insurance
Portfolio Management

7. SUMMARY

Blockchain openness:

- Infrastructure: Storage, Processing, Communications
- Trilemma: Decentralization, Scalability, Security.
- Interoperability: blockhchain system (crosschain), other system (oracle)
- Governance: make decision to improve blockchain.

8. DISCUSSION



FINISH

Thank You