Software Architecture
DIT344

Software Architectures of Blockchain
with a Case Study

Dr. Sam Jobara
jobara@chalmers.se
Chalmers | University of Gothenburg

Software Architectures of Blockchain with Case Study

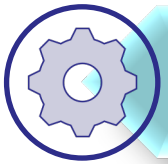
Learning experience for this lecture

- Introduce Blockchain Architecture: Building blocks
- Relate BC Architecture to Architecture styles
- Blockchain Implementations (Use Case)

References

1-Blockchain Across Oracle *by Robert van Molken Published by Packt Publishing, 2018*

2- Architecture-Centric Evaluation of Blockchain-Based Smart Contract E-Voting for National Elections
Olawande Daramola and Darren Thebus, Department of Information Technology, Cape Peninsula University of Technology, April-1 2020; Accepted: 12 May 2020; Informatics: 20 May 2020



Blockchain Terminology

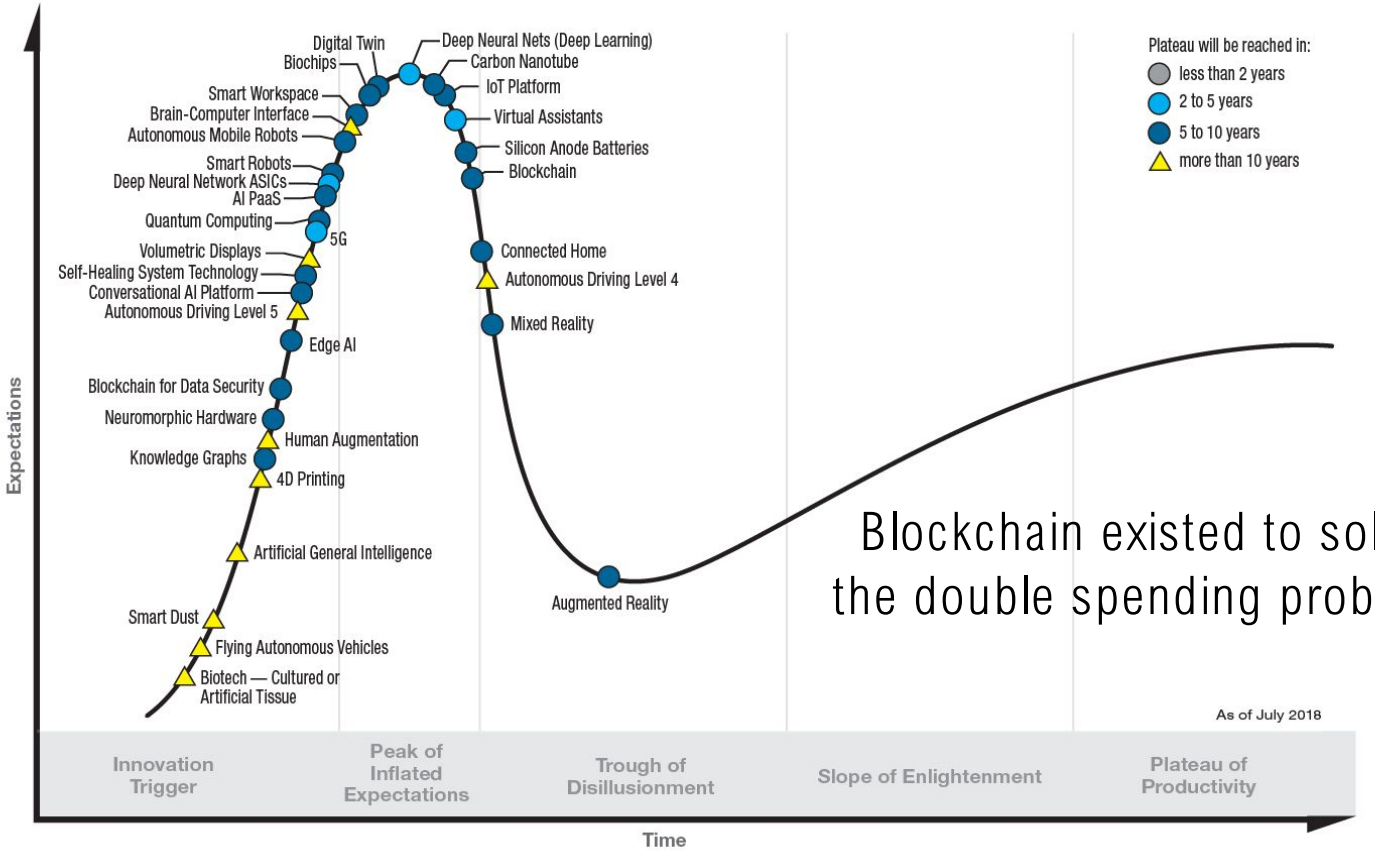
Blockchain Architecture

Blockchain Functions

BC Industrial Cases

E-Voting BC Solution

Hype Cycle for Emerging Technologies, 2018



Blockchain existed to solve the double spending problem

gartner.com/SmarterWithGartner

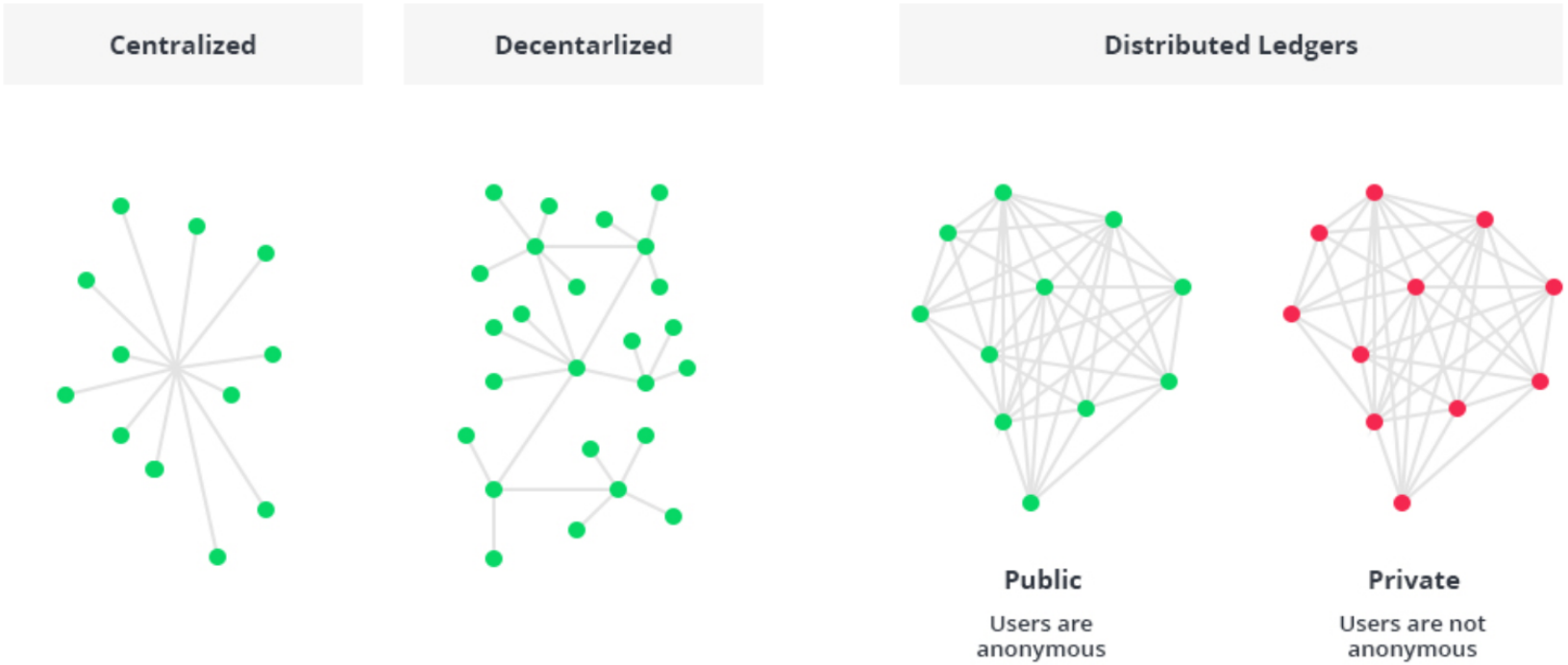
Source: Gartner (August 2018)

Gartner

Blockchain Terminology

P2P Networks

Blockchain is a decentralized, distributed ledger (public or private) of different kinds of transactions arranged into a P2P network. This network consists of many computers, but in a way that the data cannot be altered without the consensus of the whole network (each separate computer).



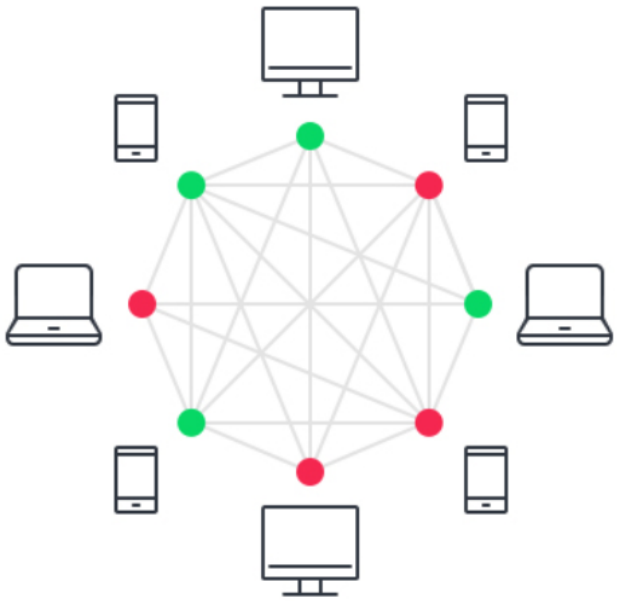
Blockchain Terminology

Types of Nodes

Blockchain nodes are not the same, Validators are able to validate, and issue new blocks (miners) with trustless consensus, while other basic nodes (users) who can initiate transa

Validator Node

Can both initiate/receive and validate transactions



Memeber Node

Can only initiate/receive transactions

Blockchain Terminology

Properties

The following table provides a detailed comparison among these three blockchain systems:

Property	Public blockchain	Consortium blockchain	Private blockchain
Consensus determination	All miners	Selected set of nodes	Within one organization
Read permission	Public	Public or restricted	Public or restricted
Immutability level	Almost impossible to tamper	Could be tampered	Could be tampered
Efficiency (use of resources)	Low	High	High
Centralization	No	Partial	Yes
Consensus process	Permissionless	Needs permission	Needs permission

Blockchain Terminology

Components

These are the core blockchain architecture components:

Node - user or computer within the blockchain architecture (each has an independent copy of the whole blockchain ledger)

Transaction - smallest building block of a blockchain system (records, information, etc.)

Block - a data structure used for keeping a set of transactions which is distributed to all nodes in the network

Chain - a sequence of blocks in a specific order

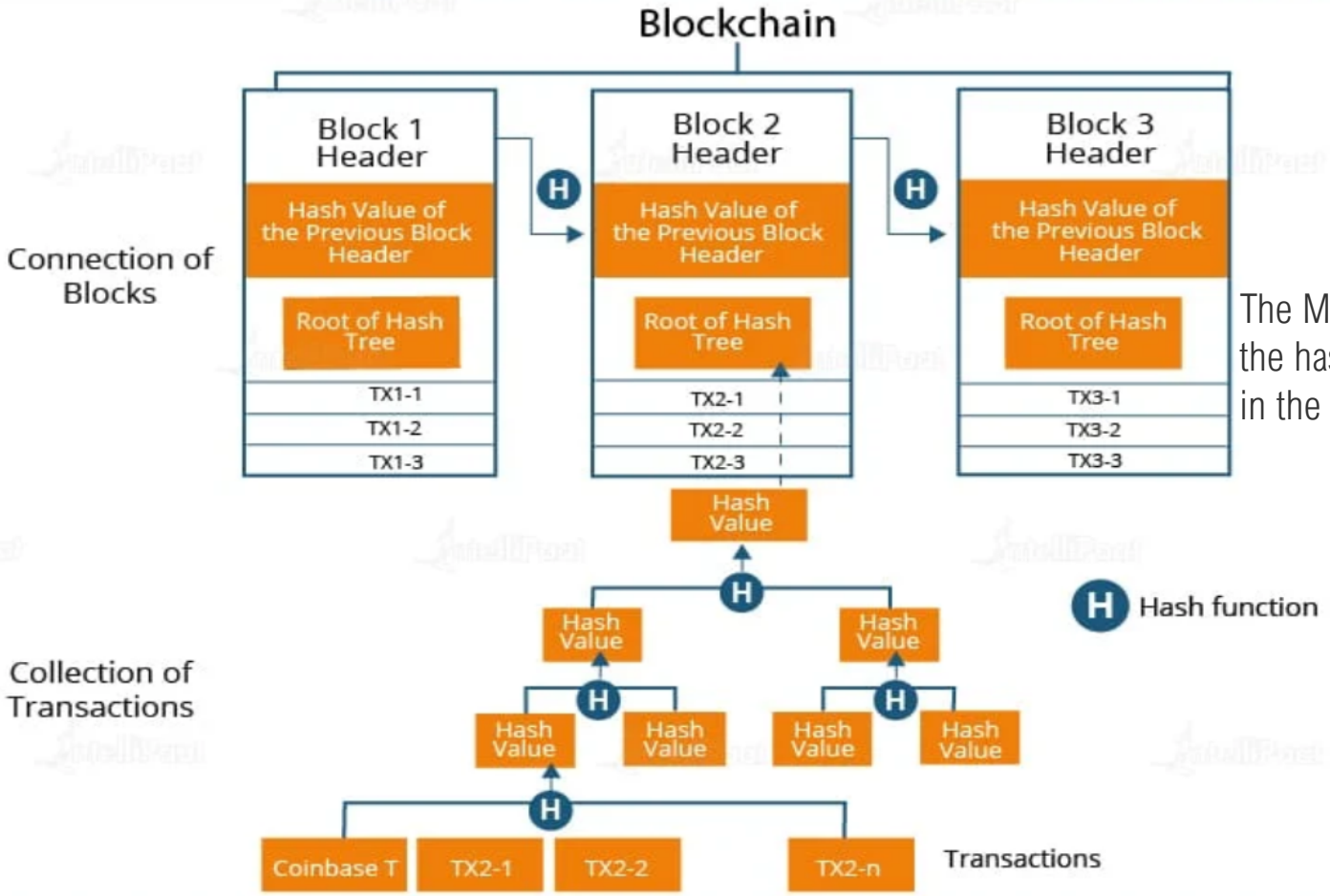
Miners - specific nodes which perform the block verification process before adding anything to the blockchain structure

Consensus (consensus protocol) - a set of rules to carry out blockchain operations

Blockchain Terminology

Blocks Structure

Each blockchain block consists of certain data, the hash of the block, the hash from the previous block, and some Transactions.



The Merkle root is the hash of all the hashes of all the transactions in the block.

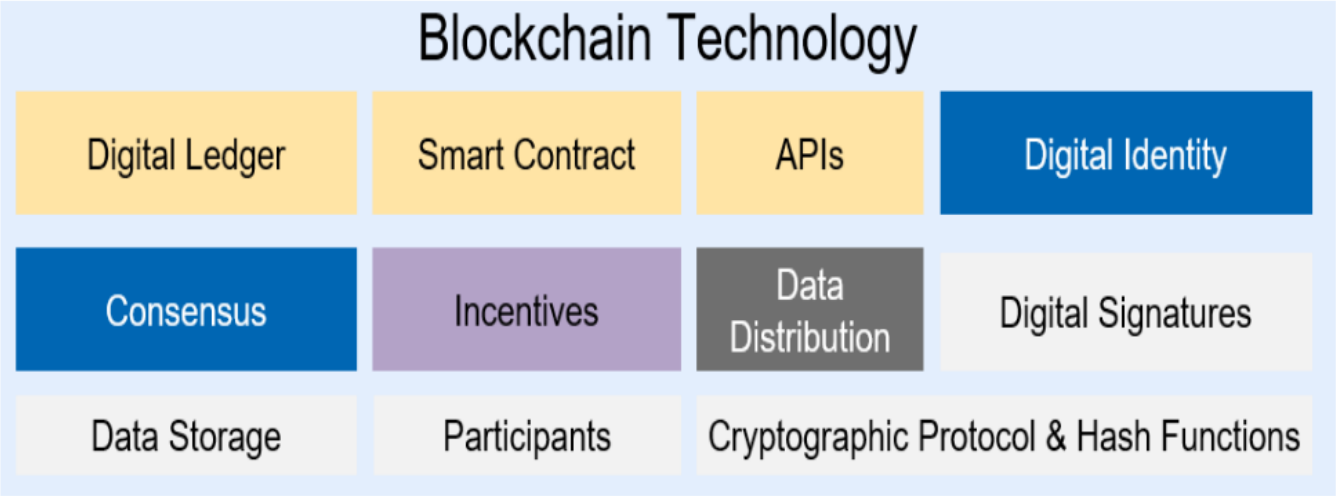
Blockchain Terminology

Distributed ledger technology (DLT)

A blockchain is a digital system of recording transactions of assets in a list that is replicated across available nodes in a P2P network.

The data is distributed to all nodes in a trustless manner (meaning without a trusted third party such as a bank) using a P2P protocol in near real time.

The following is an overview of the capabilities that make up blockchain technology:

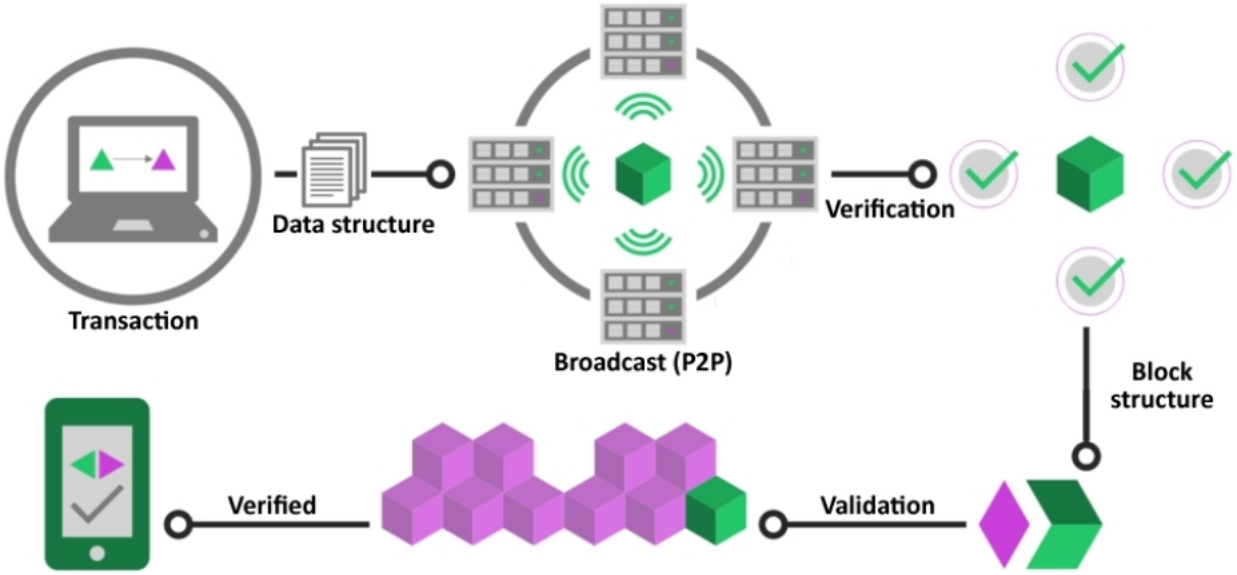


Blockchain architecture

How does a blockchain work?

The transactions that you submit are stored and verified without the involvement of a governing central authority using advanced mathematics, that is, cryptographic hash functions.

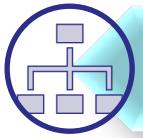
The blockchain not only secures these transactions but also protects their integrity (and anonymity). This demonstrated in the following diagram:



Learning Unit

- Understand Blockchain toplogy, P2P validator vs. member nodes
- Identify Blockchain termonolgy, properties, and the blocks role
- Identify the component of the block, and their respective role
- Understand the difference between the three types of blockchain systems
- Understand the concept of distributed ledgers of blockchains

Blockchain Terminology



Blockchain Architecture

Blockchain Functions

BC Industrial Cases

BC Industrial Cases

E-Voting BC Solution

Blockchain Architecture

BC Types & Evolution

Blockchain stage	Value enabler	Value driver
Blockchain 1.0	Decentralized peer-to-peer consensus	Transaction cost
Blockchain 2.0	Smart contracts	Extra and added services
Blockchain 3.0	Decentralized peer-to-peer applications, storage, and computing service	Organization scope and boundaries
Blockchain 4.0	Decentralized peer-to-peer artificial intelligence	Autonomous agents and decision-making

Public & Closed	**Public & Open**
• Voting • Voting records • Whistleblower	• Currencies • Betting • Video Games
Private & Closed	**Private & Open**
• Construction • National Defence • Law enforcement • Military • Tax Returns	• Supply Chain • Government financial records • Corporate earning statements

Blockchain Architecture

What is the architecture behind the blockchain?

Blockchain is not just a distributed database; it includes advanced software and security techniques to create a network of nodes (peers) that are always in sync.

Each node validates and verifies transactions and blocks redundantly, in order to reach consensus, and it provides a platform to run decentralized applications.

To achieve this, the blockchain or digital ledger technology is built upon a layered architecture.

In most cases, this contains four or five layers, namely the data layer, network layer, consensus layer, incentive layer, and application layer.

Blockchain Architecture

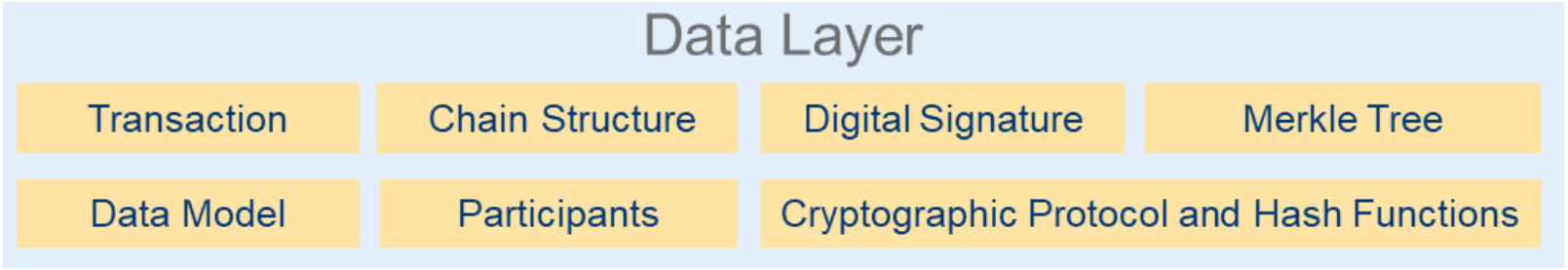
The data layer

At the bottom layer of the stack is the data layer, which deals with the data structure and the physical storage of data in the blockchain.

The diagram below shows the common capabilities that are part of this layer:

The capabilities of the data layer are:

This layer represent most data related artifacts, but not all. Blockchains are all about data and processing of data in terms of IO and deliverables.



Blockchain Architecture

The data layer

The data model can be very simple and contain just one asset, such as a cryptocurrency like Bitcoin, or a more complex model with multiple assets that can even have relationships between them.

An asset(s) can be created or referenced in a *transaction*, which in essence transfers the asset(s) between two parties who wish to exchange the data, for example, processing a payment between two parties, placing an order on an online store, registering an automobile, tracking diamonds around the world, or sharing your digital identity.

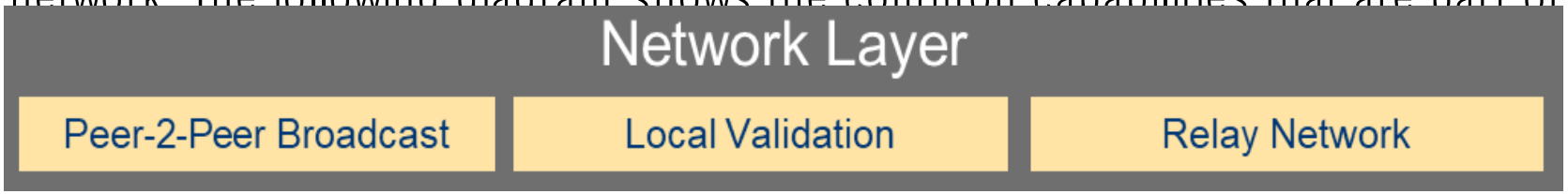
The chain structure is also related to transaction data. It describes the data structure in which individual transactions are combined into a block and how these blocks are chained to each other.

Blockchain Architecture

The network layer

The second layer up on the stack, just above the data layer, is the network layer. This layer deals with the propagation or broadcast of transactions and block data among available peers in the network, the reliability of the network, and local validation of data.

The network layer of a blockchain is similar to *BitTorrent*, and it is also managed by a peer-to-peer network, which is an architecture for distributing data in a network. The following diagram shows the common capabilities that are part of



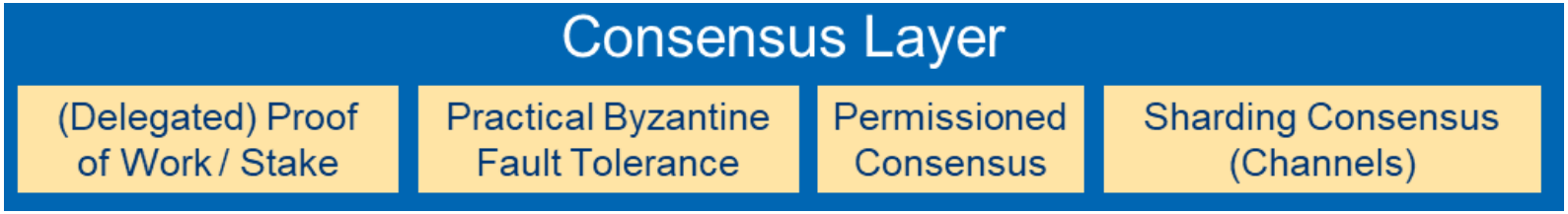
The capabilities of the network nodes varies in attributes based on public vs. private, permissioned or permissionless type.

Blockchain Architecture

The consensus layer

This layer deals with the enforcement of network rules that describe what nodes within the network should do to reach consensus about the broadcasted transactions. It also deals with the generation and verification of blocks.

The following diagram shows the common capabilities that are part of this layer:



This layer describes the rules for reaching consensus. The rules that need to be enforced depend on the consensus mechanism that is chosen when the network is initially set up.

Blockchain Architecture

Sophisticated consensus mechanisms

The **Proof of Work** (**PoW**) mechanism is used for consensus.

PoW used in the Bitcoin white paper as it allows for *trustless* and *distributed* consensus.

PoW requires participating nodes to perform an intensive form of calculations (mining)

The mining of transactions is necessary for two reasons:

- Verifying the legitimacy of transactions and record it permanently
- Creating new digital currency to reward first finished miner

Verified blocks of transactions are permanently added to the public blockchain ledger, and with every new block, the puzzle gets a bit more difficult.

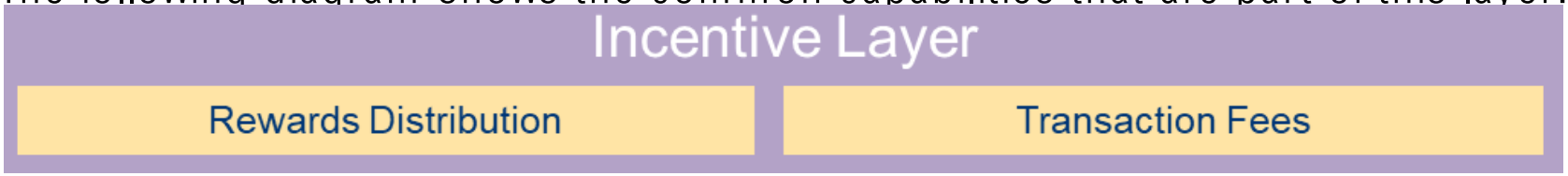
This requires miners to work more efficiently over time, this process consumes lots of power.

Blockchain Architecture

The incentive layer

This 4th layer deals with the distribution of rewards (for mining) that are earned by nodes in the network for the work they do to reach consensus. Whether this layer is implemented or not depends on the consensus mechanism in use.

The following diagram shows the common capabilities that are part of this layer:



It includes capabilities that describe what kinds of incentives are given by the network, when and how incentives can be earned by nodes, and the minimum amount of transaction fees (gas) needed to perform actions on the blockchain.

Blockchain Architecture

The application layer

The fifth (top) layer of the stack is called the application layer. This layer deals with providing the interfaces to access, program, and use the blockchain. The following diagram shows the common capabilities that are part of this layer:

The capabilities of this layer, including the programmable smart contracts and APIs.

The capabilities describe how the digital ledger is implemented and exposed to the world, how smart contracts can be built and run on the blockchain, and how third-party applications can interact with the digital ledger and smart contracts.

Learning Unit

- Understand the five layers of Blockchain architechure toplogy.
- Identify the detailed components and function of each layer
- Understand the consensus process and features of BC nodes

Blockchain Terminology

Blockchain Architecture

Blockchain Functions

BC Industrial Cases

E-Voting BC Solution

Blockchain Functions

Distributed ledgers

- A DL is a database that is synchronized and accessible across different sites and geographies by multiple P2P participants.
- The need for a central authority to keep a check against manipulation is eliminated by the use of a distributed ledger.
- A DL can be described as a ledger of any transactions or contracts maintained in decentralized form across different locations and nodes.
- Cyber attacks and financial fraud are reduced by the use of distributed ledgers.
- All the information on the ledger is securely and accurately stored using cryptography and can be accessed using keys and cryptographic signatures.
- Once the information is stored, it becomes an immutable database, which the rules of the network govern.

Blockchain Functions

Hyperledger Fabric

HYPERLEDGER FABRIC

Hyperledger Fabric is an open-source community enterprise-grade, distributed ledger platform that offers modularity and versatility for a broad set of industry use cases. The modular architecture for Hyperledger Fabric accommodates the diversity of enterprise use cases through plug and play components, such as consensus, privacy and membership services.

The key features of Hyperledger Fabric

- Permissioned architecture
- Highly modular
- Pluggable consensus
- Open smart contract
- Low latency of finality/confirmation
- Flexible approach to data privacy
- Multi-language smart contract support: *Solidity, Golang, Java, Javascript*
- Designed for continuous operations
- Governance and versioning of smart contracts
- Flexible endorsement model for achieving consensus across required organizations

Blockchain Functions

Smart contract

Smart contracts can act as a complement, or substitute, for legal agreements.

They are computer code that directly control some aspects of condition-based transactions.

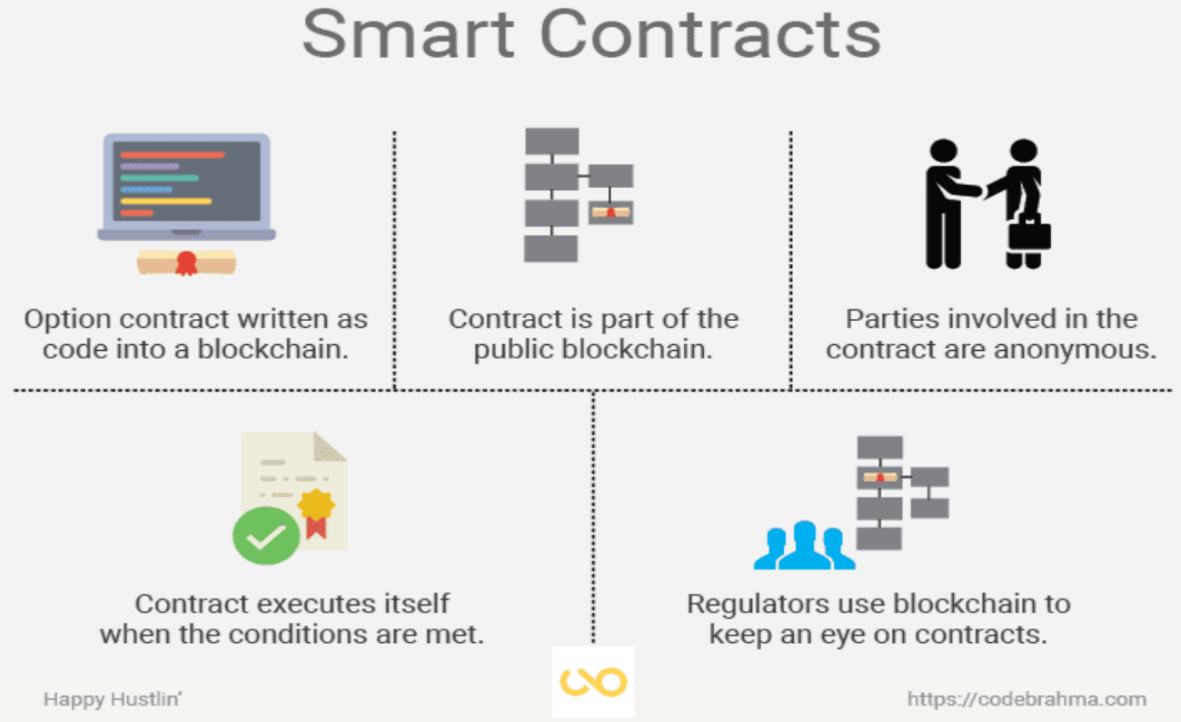
A smart contract also capable of automatically facilitating, executing, and enforcing the negotiation or

agreement.

Smart contracts are immutal

and are enforced by the syst

itself.



Blockchain Functions

Decentralized applications

A capability that is still a very new concept is a decentralized application.

A **decentralized application** (**dApp**) is a blockchain-enabled website that runs independently on every node of the peer-to-peer network, rather than on a single serve.

They are comprised of both a frontend (web) application and a backend application, where the smart contract (backend application) allows it to connect to the blockchain.

For example, a decentralized application includes the data model it uses (participants, assets, and transactions), an authorization and permissions model, smart contracts (backend), and a frontend web application.

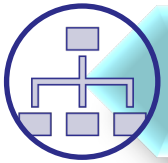
Learning Unit

- Understand the function of Distributed Ledgers
- Understand the function and features of the Hyperledger Fabric
- Understand the smart contract purpose and function
- Understand the Distributed Aps (dApps) purpose and function

Blockchain Terminology

Blockchain Architecture

Blockchain Functions



BC Industrial Cases

E-Voting BC Solution

Blockchain Functions

Blockchain for financial services

Blockchain technology offers a number of key benefits for financial services, such as cost reduction (fewer errors, cutting out the middleman), improved business outcomes (single point of truth), and reduced responsibility and risk (no offline reconciliation) to disrupt their bus

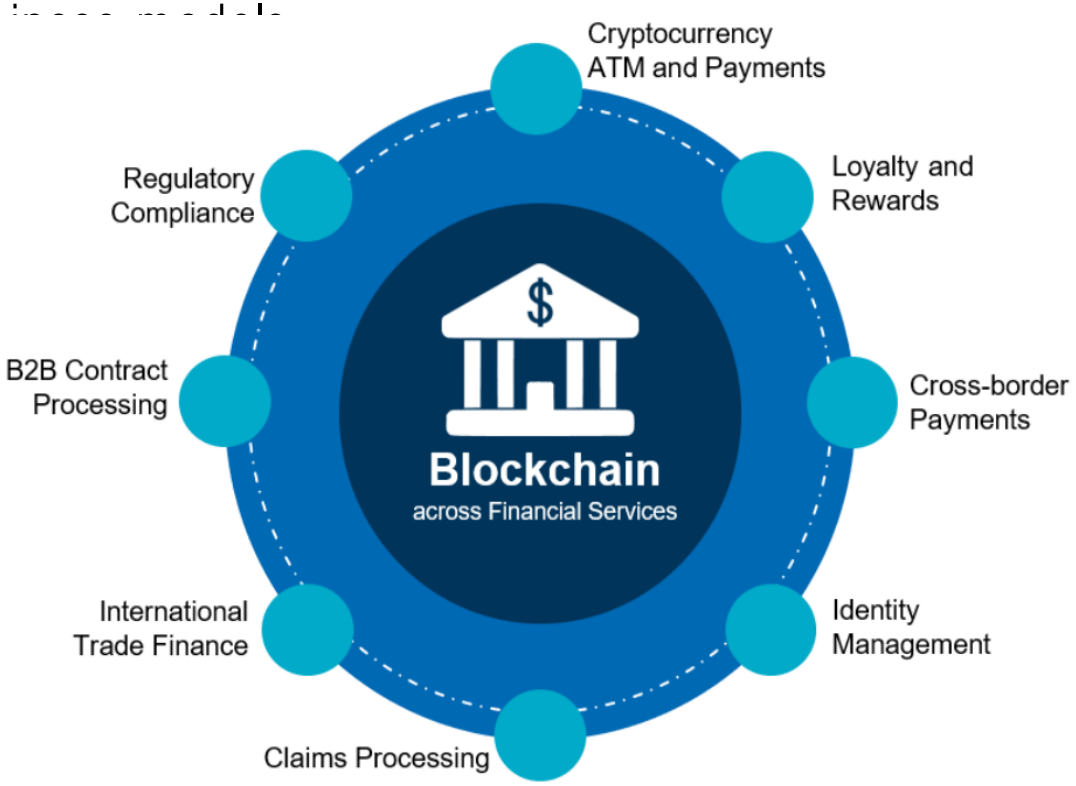
Smart contracts, also act as a sha

application / tool to govern chang

to the underlying ledger and state

database.

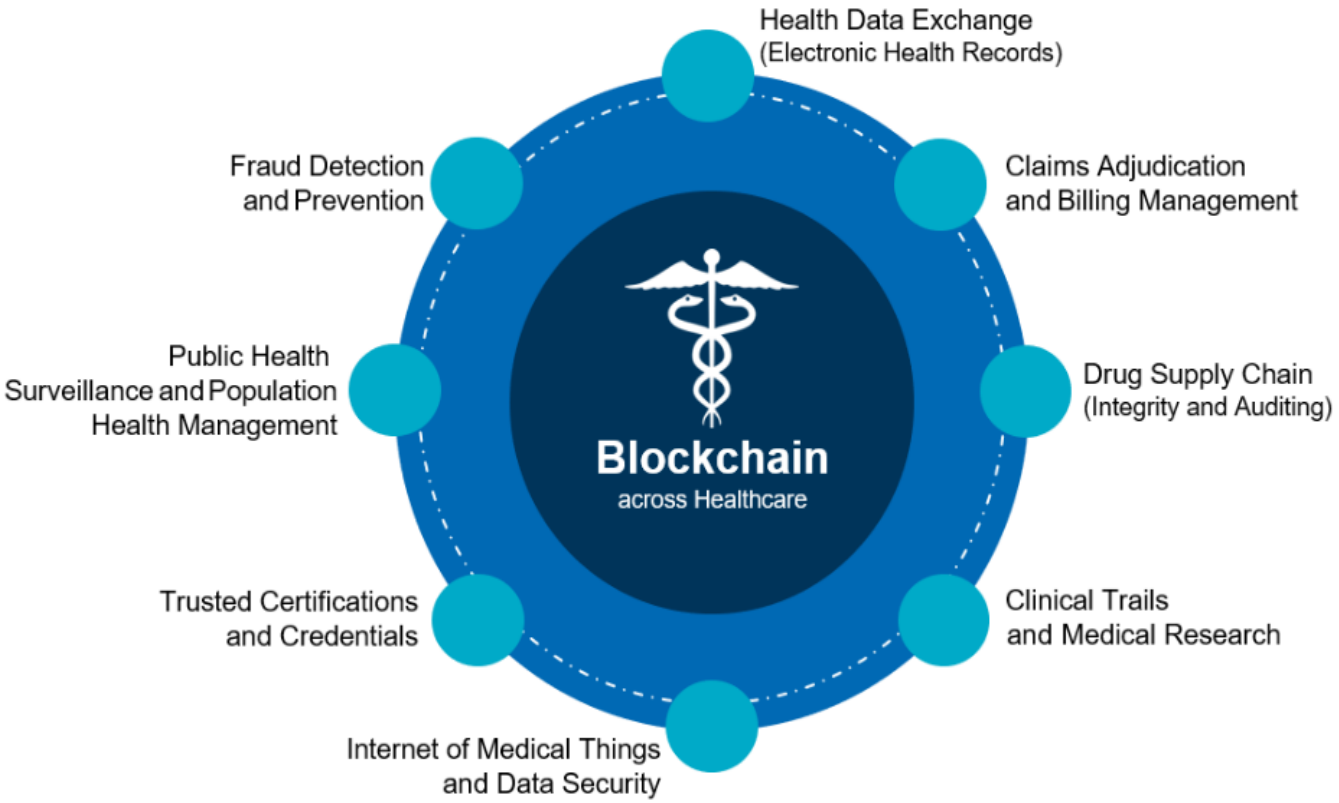
look at the following diagram:



Blockchain Functions

Blockchain for Healthcare Industry

Blockchain has a wide range of use cases in healthcare. The ledger technology facilitates the secure transfer of patient medical records, manages the medicine supply chain and helps healthcare researchers unlock genetic code.



Blockchain Functions

Blockchain for Healthcare Industry

We site her two use cases

MEDICALCHAIN

Industry: Medical Health Record

Location: London, England

What they do: blockchain maintains the integrity of health records while establishing a single point of truth. Doctors, hospitals and laboratories can all request patient information that has a record of origin and protects the patient's identity from outside sources.

Blockchain application: blockchain-based platform maintains a record of origin and protects patient identity.

BLOCKPHARMA

Industry: Pharmaceuticals, Supply Chain

Location: Paris, France

What they do: offers a solution to drug traceability and counterfeiting. By scanning the supply chain and verifying all points of shipment, the company's app lets patients know if they are taking falsified medicines With the help of a blockchain-based SCM system.

Blockchain application: Through its app, its blockchain-based system can help prevent counterfeit medicines.

Blockchain Functions

Social Media Platforms

The social media landscape is currently controlled by a small number of large corporations that have all the power to decide what people can and cannot say on social media.

Moreover, social media giants monetize user data by selling it to advertisers in exchange for providing their "free" service, giving their users little to no control over their data.

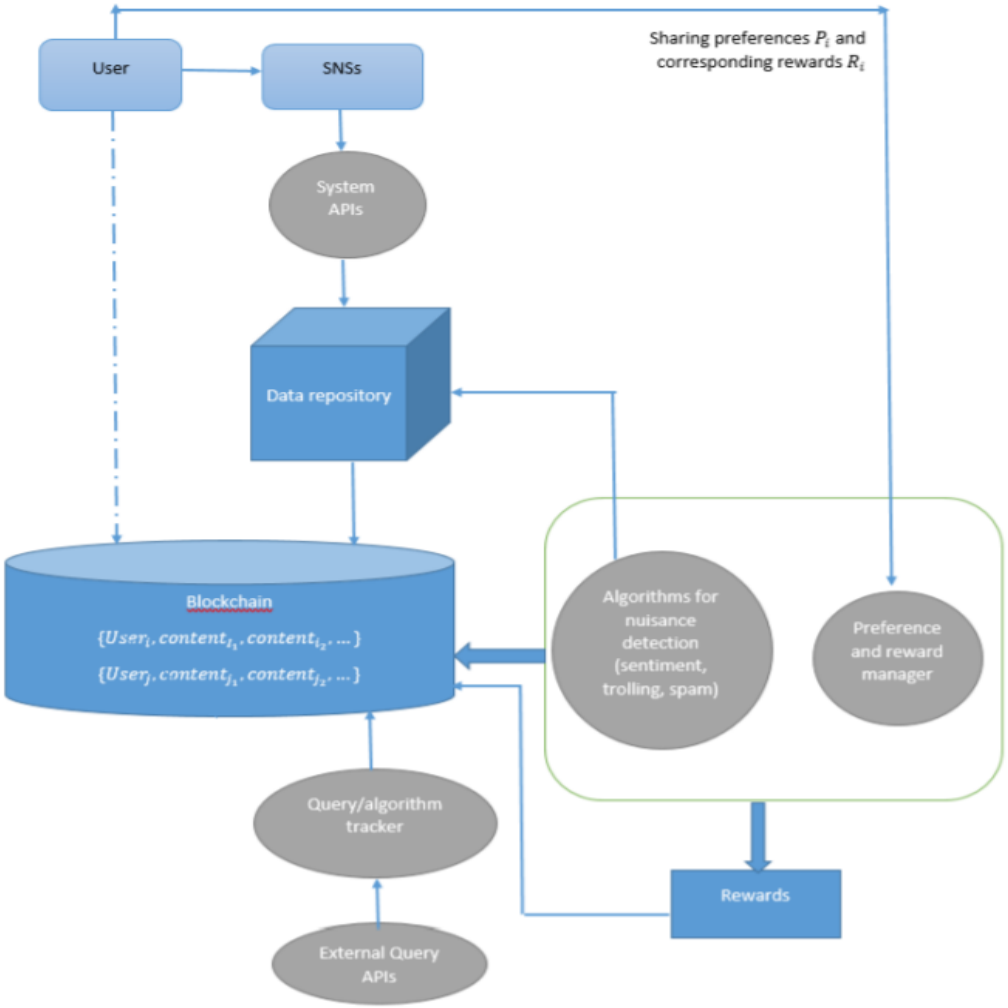
As a result, a number of blockchain startups have been launched to decentralize social media and to return content and data ownership back to the user and away from large centralized organizations.

Blockchain Functions
Social Media Platforms

Blockchain based Social Media Platforms are characterized by the following:

- No Single-Point of Failure.
- No Censorship of data or platform Rewards for Valuable Content.
- A content creator can be rewarded for valuable content authenticity.

Examples of decentralized social medias are: SteemIt, Lit, HyberSpace, Sapien, and SocialSpace*



Blockchain System architecture and data model of decentralized Social Media

* When Blockchain meets Online Social Networks
Barbara Guidi University of Pisa, Department of Computer Science, Largo Bruno February 2020

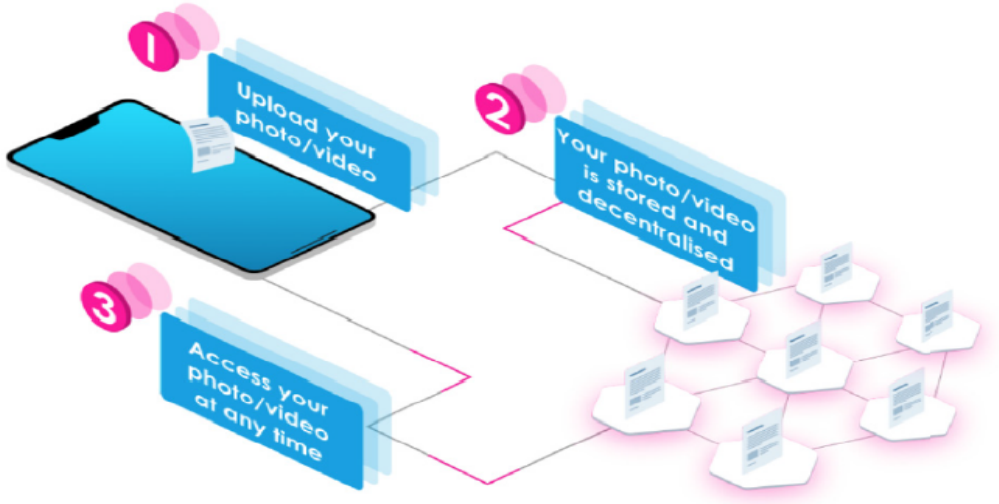
Blockchain Functions

Social Media Platforms

SocialX is fully decentralized social media platform.

All media (photos and videos) and data (messages, posts, etc.) are stored in a decentralized manner. The principal goal of the platform is to face the problem of fake accounts, fake followers, and fake votes.

Indeed, the decision power is given to communities, which can decide what content is valuable. SocialX reads user data as a raw file, then store it in distributed file system which is hosted on Ethereum node cluster. Blockchain nodes are used to save transactional operations and to operate smart contracts.



Learning Unit

- Understand the use cases and industry applications of Blockchains
- Realize the deployment benifits of BC in Financial, Medical, and Social media fields.

Blockchain Terminology

Blockchain Architecture

Blockchain Functions

BC Industrial Cases

E-Voting BC Solution

Industrial Case

Blockchain-Based Smart Contract E-Voting for National Elections*

Objective

A highly secured E-voting is one of the valid use cases of blockchain technology. This research involves elicitation of the e-voting requirements, formulation of a blockchain e-voting architecture, an architecture-based evaluation, the analysis of the results, and a report of the findings.

* Architecture-Centric Evaluation of Blockchain-Based Smart Contract E-Voting for National Elections
Olawande Daramola, and Darren Thebus, Informatics, May 2020
Department of Information Technology, Cape Peninsula University of Technology, Cape Town

Smart Contract E-Voting Requirements

The following are the system key stakeholders requirements:

1- Trust: All stakeholders must have confidence and trust the SCE outcome

2- Transparency: System supports the casting of votes and tally of votes by all stakeholders, as well as allow them to verify this easily.

3- Verifiability: The system must enable voters to check that their votes were cast and recorded as valid votes.

4- Auditability: Support any process that may necessitate the rechecking and recounting votes in the event of electoral disputes.

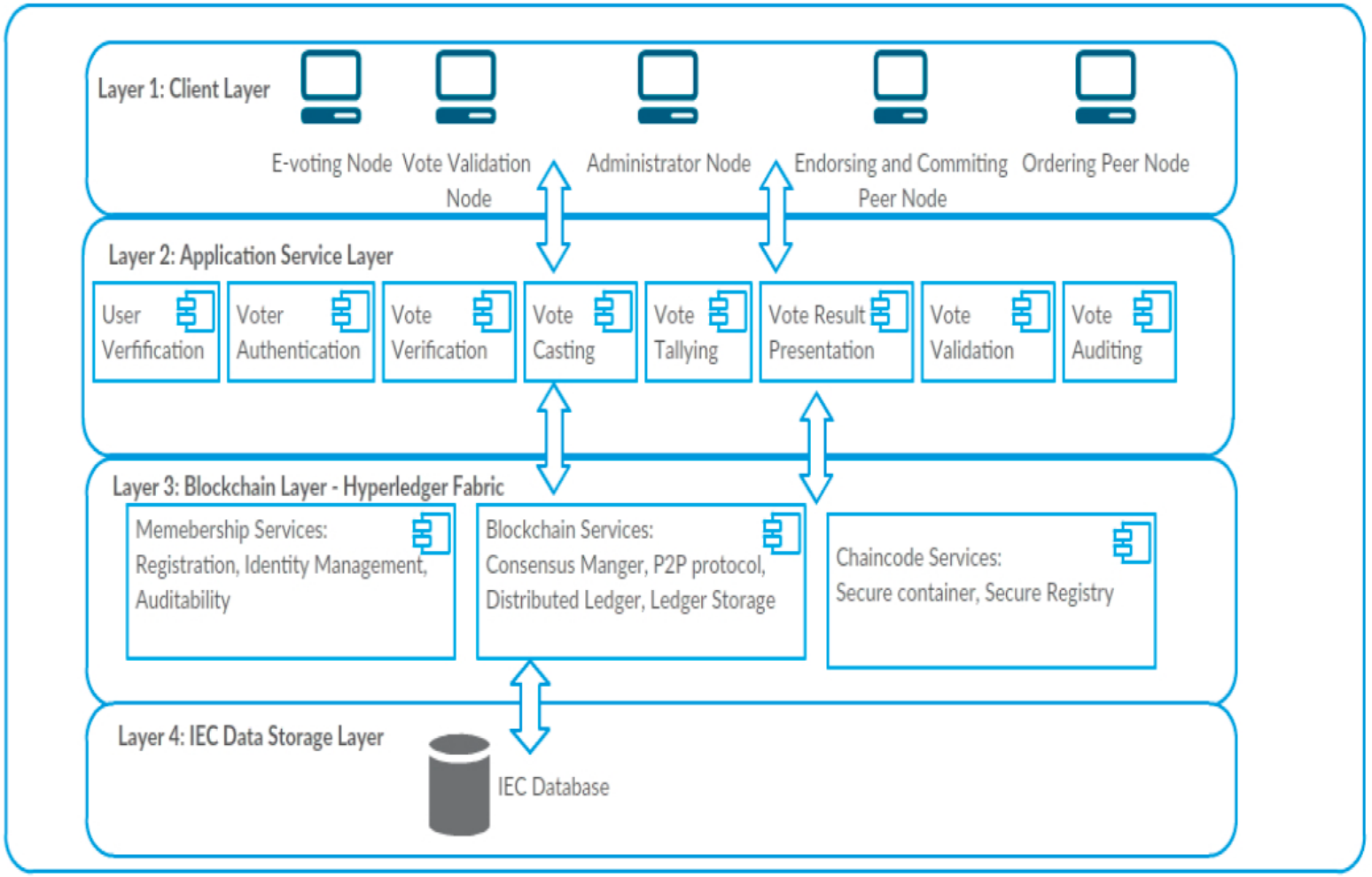
5- Availability: Backup system ensuring almost zero-down time.

6- Performance: Ensure that all operations are handled speedily and efficiently. the identity of voters, and the choices made during voting.

7- Socio-political factors: The e-voting system should not be vulnerable to socio-political manipulations that can compromise the integrity of the voting process.

BANES Architecture

Based on the identified requirements, a Blockchain-based Architecture for National E-voting system (BANES) a Layered Architecture was proposed, as shown below in Fig.1



Blockchain style within Layered Architecture Style

Figure 1. A schematic view of the blockchain-based architecture for national e-voting system (BANES).

BANES Architecture

Client Layer

This layer contains the various electronic devices and systems with which users interact with the blockchain e-voting system. These devices are the peer nodes of the e-voting blockchain that interact via smart contracts, referred to as "chaincode" in the Hyperledger Fabric.

(i) E-Voting nodes: enable voters authentication and casting of votes, and to ensure that all blockchain transactions are recorded.

(ii) Administrator nodes: used to configure blockchain network channels, assign roles to the nodes of the blockchain, and grant permissions.

(iii) Public nodes: enable public view-only to transactions of the e-voting blockchain.

(iv) Vote validation: responsible for vote validation. They are also used to ensure the authenticity of transactions that are included in a block.

(v) Committing nodes: These are the nodes that validate and commit new blocks to the blockchain.

BANES Architecture

Application Service Layer

Consists of a set of services that are available in the e-voting system. The level of access control and the defined permissions level determines the type of services that a node can access in the blockchain.

Blockchain Layer

It is composed of the Hyperledger Fabric V2.0, which is a modular blockchain architecture framework that facilitates blockchain information system solutions.

It supports the creation of permissioned blockchain networks that have in-built properties such as security, and privacy protection.

The Hyperledger Fabric has "ordering nodes" which ensures consistency of the blockchain by ensuring that only ordered blocks of an endorsed transaction are made available to the committing peer nodes before they are added to the blockchain.

As we stated before,

BANES Architecture

As stated before, the use of Hyperledger fabric is motivated by:

- Permissioned architecture
- Highly modular
- Pluggable consensus
- Open smart contract
- Low latency of finality/confirmation
- Flexible approach to data privacy

IEC* Data Storage Layer

It contains the relevant databases that store information on the profile of registered voters. This database is used as the basis to authenticate and authorize voters to vote.

* ISO/IEC 27040 is to provide security guidance for storage systems and ecosystems as well as for protection of data in these systems.

BANES Architecture

BANES is premised on two key concepts, which are smart card technology and the zero-knowledge protocol.

Smart Card Technology for Voter Authentication.
This is used to eliminate impersonation and to ensure that only valid voters can vote. During the voter registration exercise before election time, a smart card shall be given to each voter by the central electoral authority. The smart card will contain the voter's public key for identification, which will be combined with a personal identification number (PIN) for voters' authentication.

Zero-knowledge Protocol for Voter Authentication and privacy protection.
The zero-knowledge protocol was applied in the BANES to ensure that when an authorized voter casts a vote, the blockchain knows that a valid vote has been cast and nothing more. The identity of the voter and voter's choice is not revealed.

BANES Architecture

A Process View of the Blockchain Architecture for (BANES)

It is assumed that the casting of votes will take place at designated polling units to protect voters from being coerced to vote in certain ways by politicians and their agents. With the
BANES, the e-voting procedure will follow the procedure below as in Fig.2, 3, &4:

(i) The voter inserts the personal smart card into the voting node and supplies a password.

(ii) Authentication and authorization of the votes take place via the IEC database.

(iii) If successful, a digital ballot is generated by the IEC system. A digital ballot consists of a set of candidate public keys and a unique ballot ID.

(iv) Voter submits a vote for the preferred candidate.

(v) The ballot ID is assigned to the preferred candidate through their public key. The transaction is authenticated by using the digital signature of the private key.

(vi) The transaction is sent to all nodes and stored on the blockchain.

A Process View of the Blockchain Architecture for E-Voting (BANES)

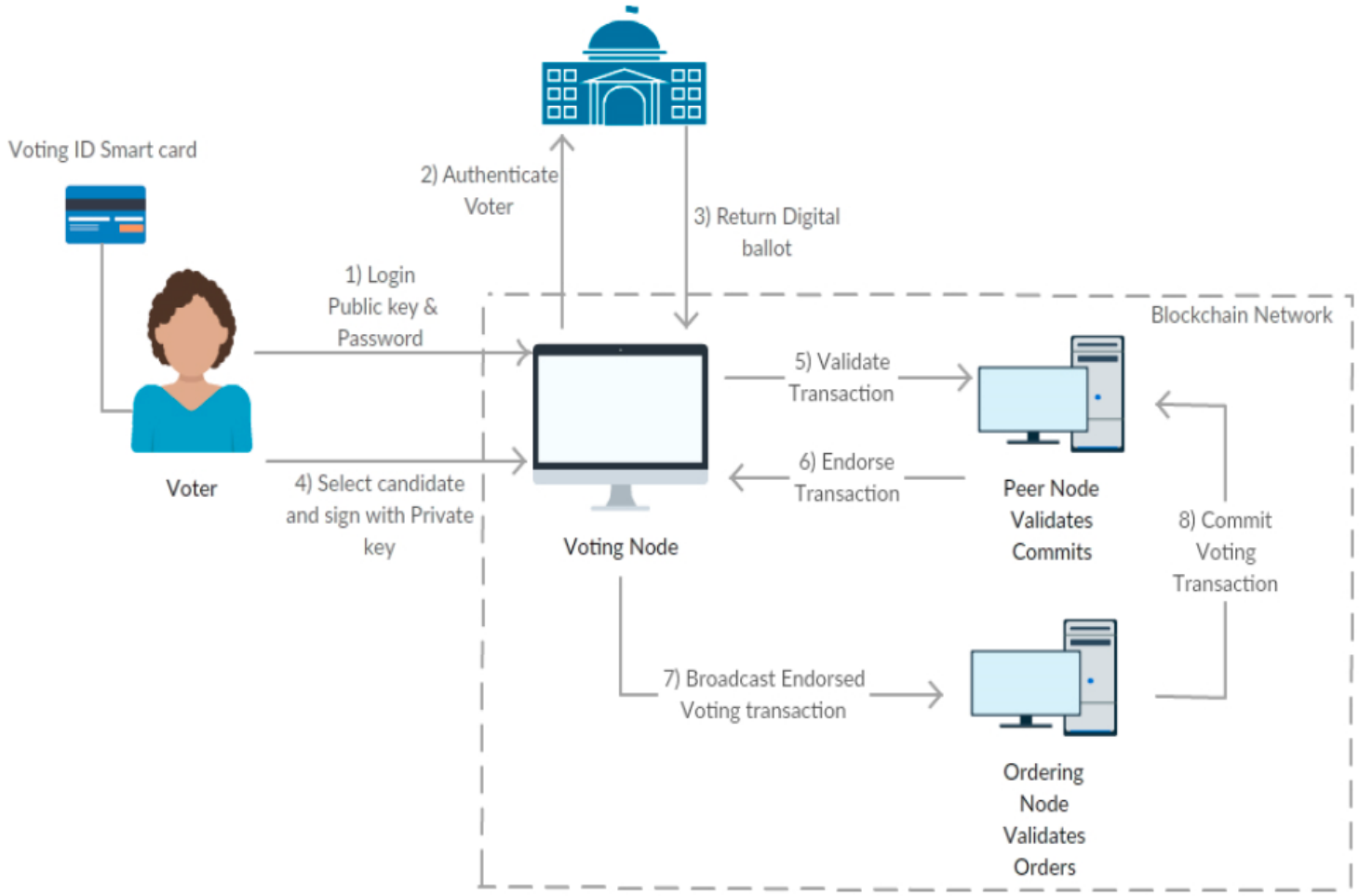


Figure 2. A high-level view of the e-voting process using the BANES.

Processing View- UML activity diagram of BANES

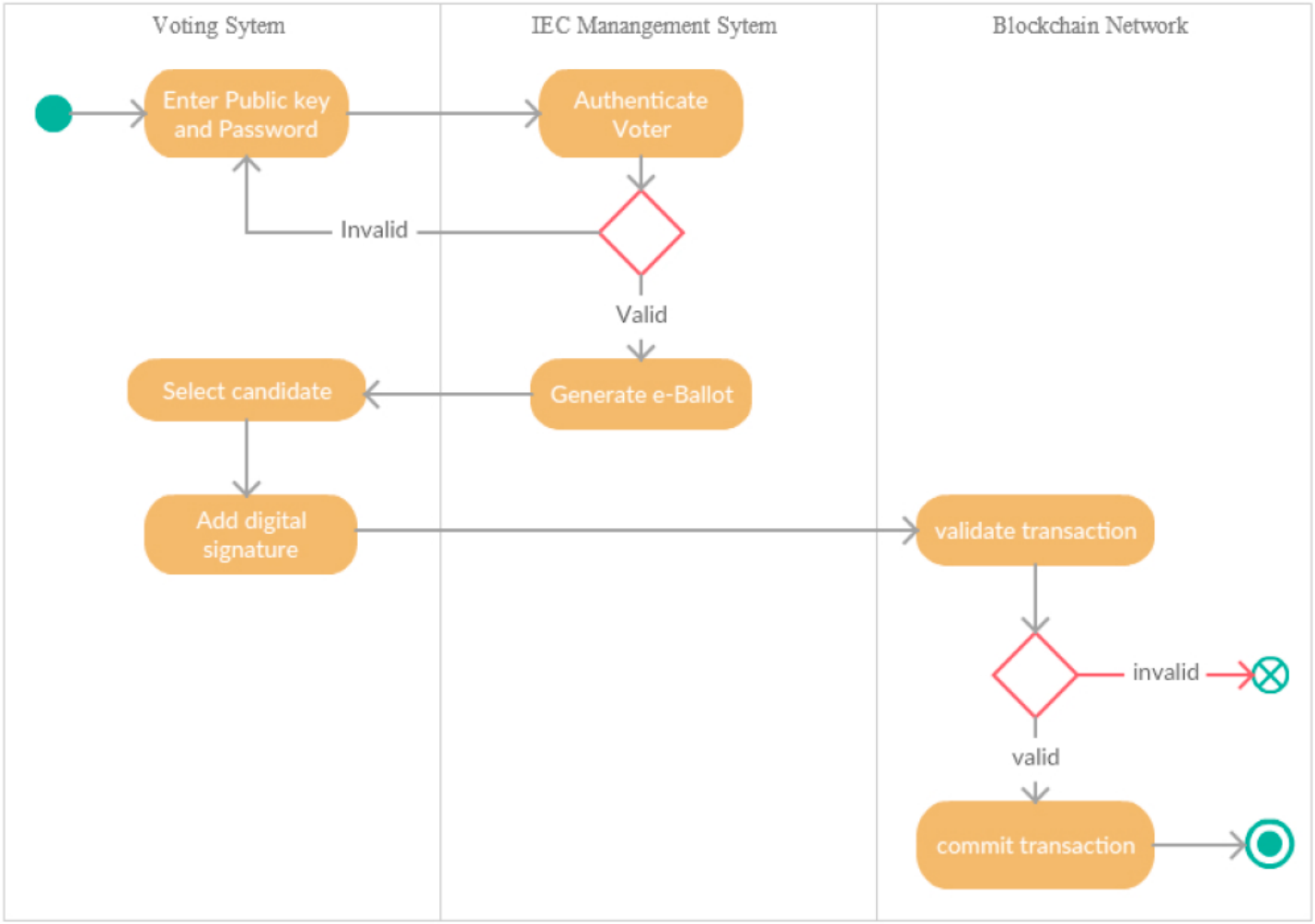


Figure 3. Process View—UML activity diagram of BANES.

Development View- UML component diagram of BANES

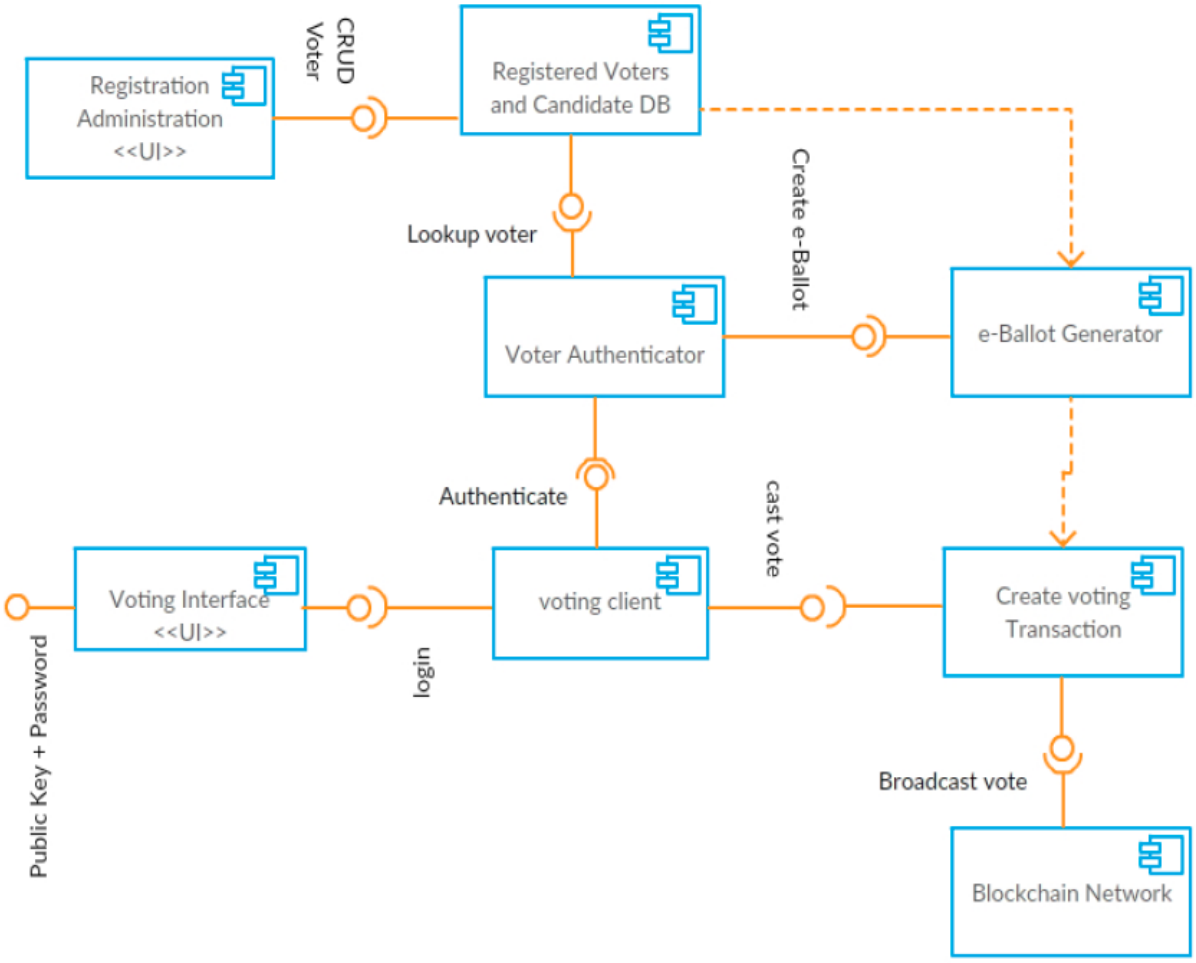


Figure 4. Development View—UML component diagram of BANES.

BANES Architecture
A Process View of the Blockchain Architecture for (BANES)

The architectural approaches that have been adopted and the rationale for adopting them were explained. The summary of quality attributes, the approach, and component that seek to address them is presented as follows:

- Voters Smart card—Identification, Authentication, and Authorization
- Zero-knowledge protocol—Security
- Hyperledger Fabric Blockchain:
  - Public/private key Encryption—Security, Functional Suitability
  - Modular Architecture—Reliability
  - REST* API's—Reliability, Security, Performance
  - Permissioned Blockchain network—Security, Functional Suitability
  - Isolation of system services—Performance, Security
  - Distributed processing—Security
  - Decentralized processing and storage—Functional Suitability, Reliability
  - Immutability—Security, Functional Suitability

*

Learning Unit

- Appreciate the value for BC in solving the challenges of e-voting
- Realize the importance of stakeholders requirement and assessment role in the development of the BANES system
- Understand the 4-layers architecture design of the BANES system
- Familiarize yourself with the UML process and development views of BANES design.

Remember the Hype Curve