1

UNIT - I

Mathematical Foundation for Blockchain

Syllabus

Cryptography : Symmetric Key Cryptography and Asymmetric Key Cryptography, Elliptic Curve Cryptography (ECC), Cryptographic Hash Functions : SHA256, Digital Signature Algorithm (DSA), Merkel Trees.

1.1 Cryptography

1.1.1 Introduction

Cryptography is the study of using mathematics to encrypt and decrypt information or data to be stored or transmitted on insecure networks. Cryptography helps you to store vital information or transmit it across public or private networks (e.g. Internet) so that it cannot be perceived by anyone except the intended recipient.

Definition

Cryptography is defined as storing or transferring encrypted data between sender and receiver on public or private networks such that only intended recipients can read it on receiving it.

1.1.2 Why do we need Cryptography?

We need cryptography to have secure communication. Following are the goals of cryptography :

- **Privacy and Confidentiality :** The message to be transmitted shall be encrypted using a secret key by the sender and similarly it shall be decrypted only by the intended recipient by his/her own secret key.
- **End-point authentication :** Authentication is a mechanism that is used whenever one wants to know exactly who is using or viewing the message. Usually login and password is used to confirm the identity and view the message.
- **Integrity of a message :** It is the mechanism to ensure that the message is not changed or modified while the message is being transmitted between sender and receiver.
- **Non-repudiation :** Sender cannot deny his intentions of transmitting a message later, once it is sent by a sender to receiver, this is known as non-repudiation. Digital signatures are the most commonly used mechanism to achieve and ensure the non-repudiation.
- **Key Exchange :** It is the mechanism by which a sender and receiver can share cryptographic keys among them so that they can be used to encrypt and decrypt the message.

1.2 Types of Cryptography

As indicated in Fig. 1.2.1, types of Cryptography techniques are :

1. Symmetric Encryption
2. Asymmetric Encryption
3. Hash Functions

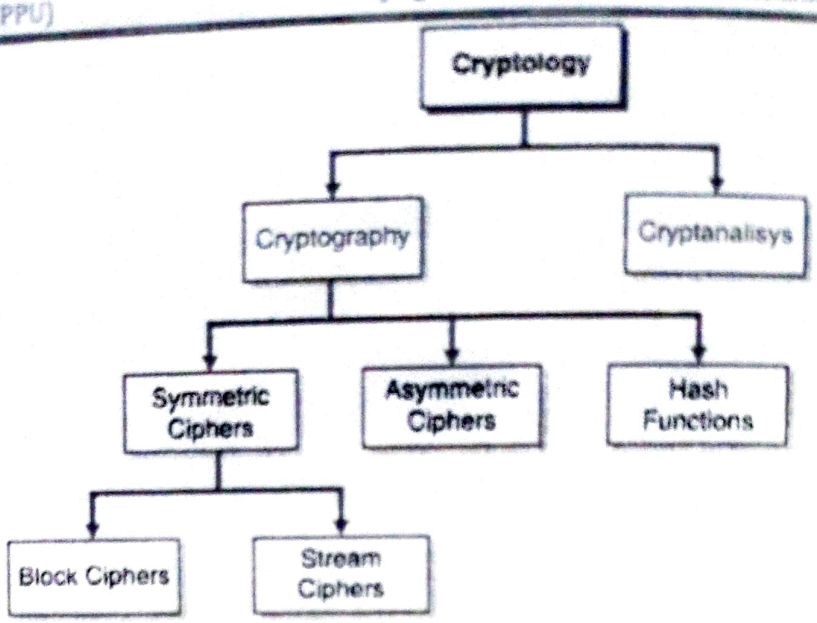

Fig. 1.2.1 : Cryptology

Let us understand the symmetric and asymmetric key cryptography in the following sections

1.2.1 Symmetric Key Cryptography

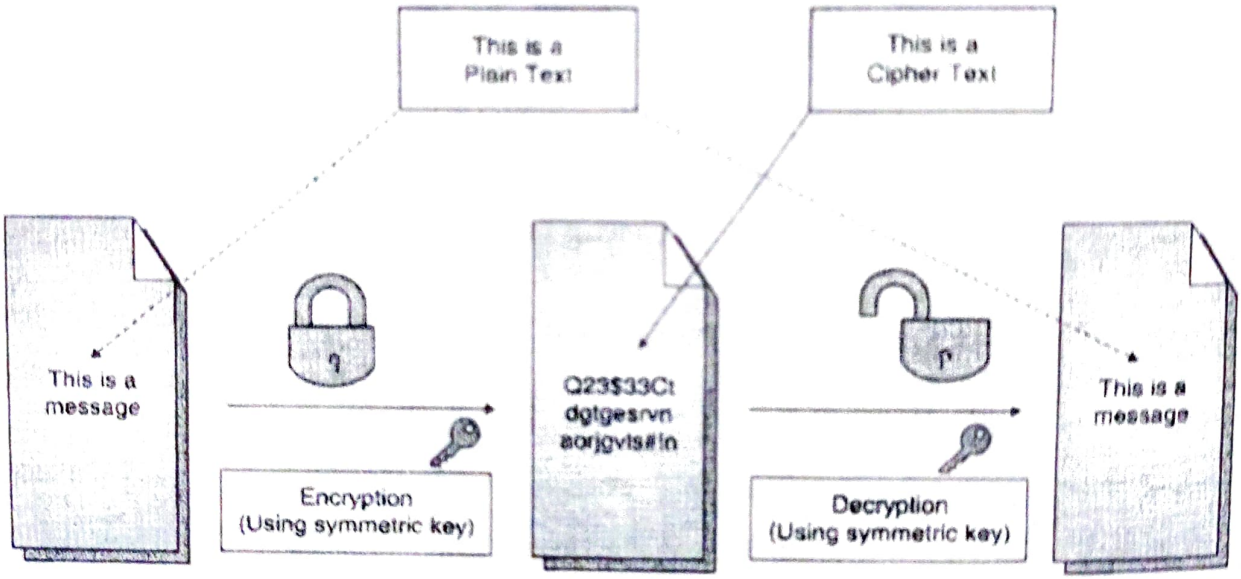

Fig. 1.2.2 : Symmetric key Encryption

- In this type of encryption, sender and receiver share the single key to encrypt and decrypt the message. Here the sender and receiver know the secret key.
- Steps and working of symmetric key encryption is as follows :

1. Sender and receiver share the secret key through some external means.
2. The sender encrypts the message (plain text) using his copy of the key at the sender's end. Here the plain text is encrypted to generate the cipher text.
3. The cipher text is then sent to the receiver over the communication channel.
4. The recipient decrypts the cipher text using his copy of the key at receiver's end.
5. Here the cipher text is decrypted back into the original plain text.

Similar process can be followed vice - a - versa.

1.2.2 Asymmetric Key Cryptography

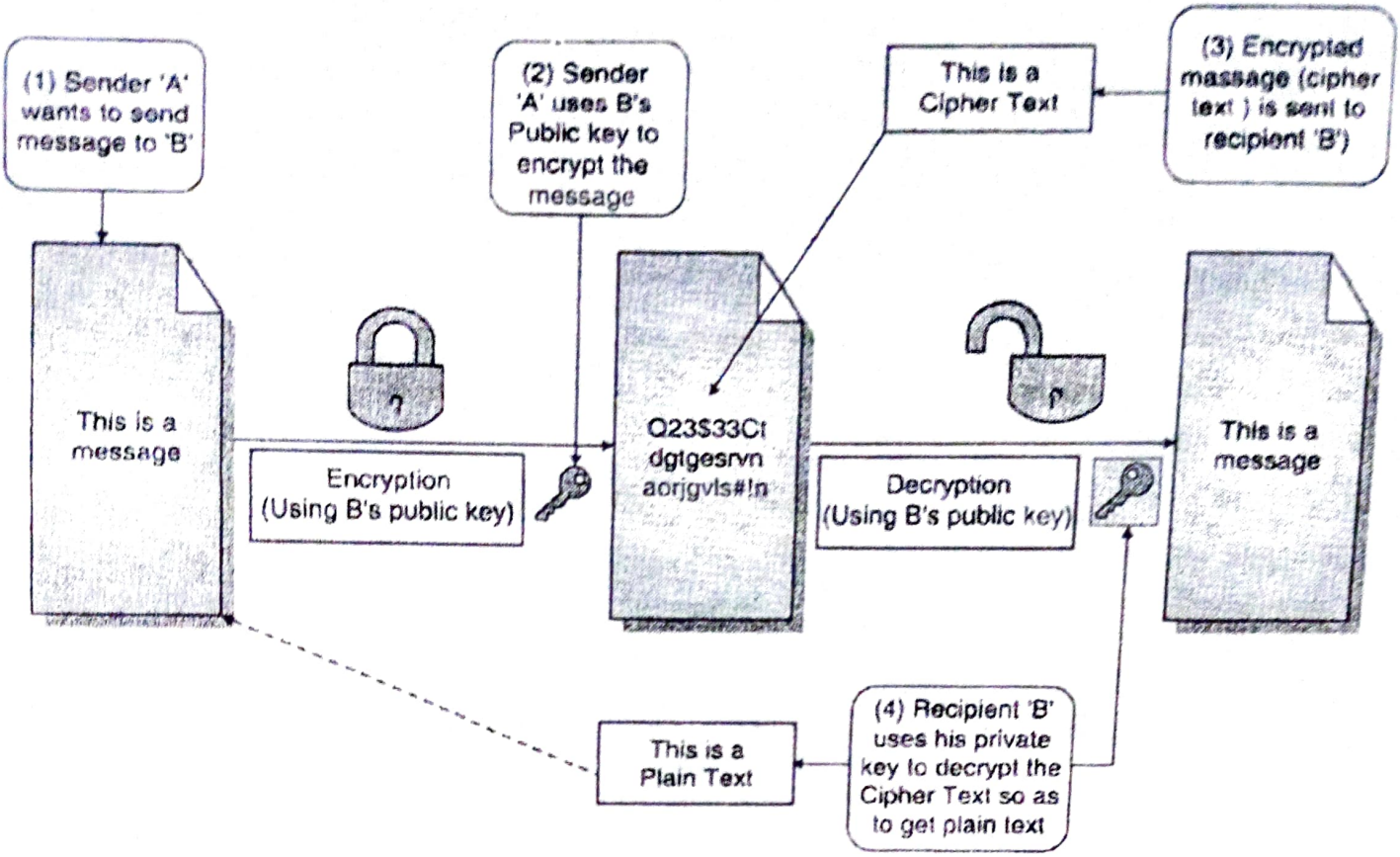

Fig. 1.2.3 : Asymmetric key Encryption

- In this type of encryption, sender and receiver use two different keys that are known as public key and private key to encrypt and decrypt the message. Here the sender and receiver do not share the secret key. As the keys are different it is said as asymmetric key cryptography.
- Steps and working of Asymmetric key encryption is as follows :
- At sender's end,
  1. Sender encrypts the message using the receiver's public key.
  2. The public key of the receiver is publicly available and known to everyone.
  3. Encryption mechanism converts the message(plain text) into a cipher text.
  4. This cipher text can be decrypted only using the receiver's private key.
  5. The cipher text is sent to the receiver over the communication channel.
- At receiver's end,
  1. Receiver decrypts the cipher text using his private key.
  2. The private key of the receiver is known only to the receiver.
  3. Using the public key, it is not possible for anyone to determine the receiver's private key.
  4. After decryption, cipher text converts back into a readable plain text format.

1.3 Elliptic-Curve Cryptography (ECC)

- Elliptic curve cryptography is used to implement public key cryptography. It was discovered by Victor Miller of IBM and Neil Koblitz of the University of Washington in 1985. ECC popularly used an acronym for Elliptic Curve Cryptography. It is based on the latest mathematics and delivers a relatively more secure foundation than the first-generation public key cryptography systems for example RSA.
- In 1985, cryptographic algorithms were proposed based on elliptic curves. An elliptic curve is the set of points that satisfy a specific mathematical equation. They are symmetrical.
- ECC is among the most commonly used implementation techniques for digital signatures in crypto currencies. Both Bitcoin and Ethereum apply the Elliptic Curve Digital Signature Algorithm (ECDSA) specifically in signing transactions. However, ECC is not used only in crypto currencies. It is a standard for encryption that will be used by most web applications going forward due to its shorter key length and efficiency.

Elliptic curve equation is as follows :

$$y^2 = x^2 + ax + b$$

- Any non-vertical line will intersect the curve in three places or fewer.
- Fig. 1.3.1 shows the curve can be plotted on the graph and the line is intersecting at points A, B and C.

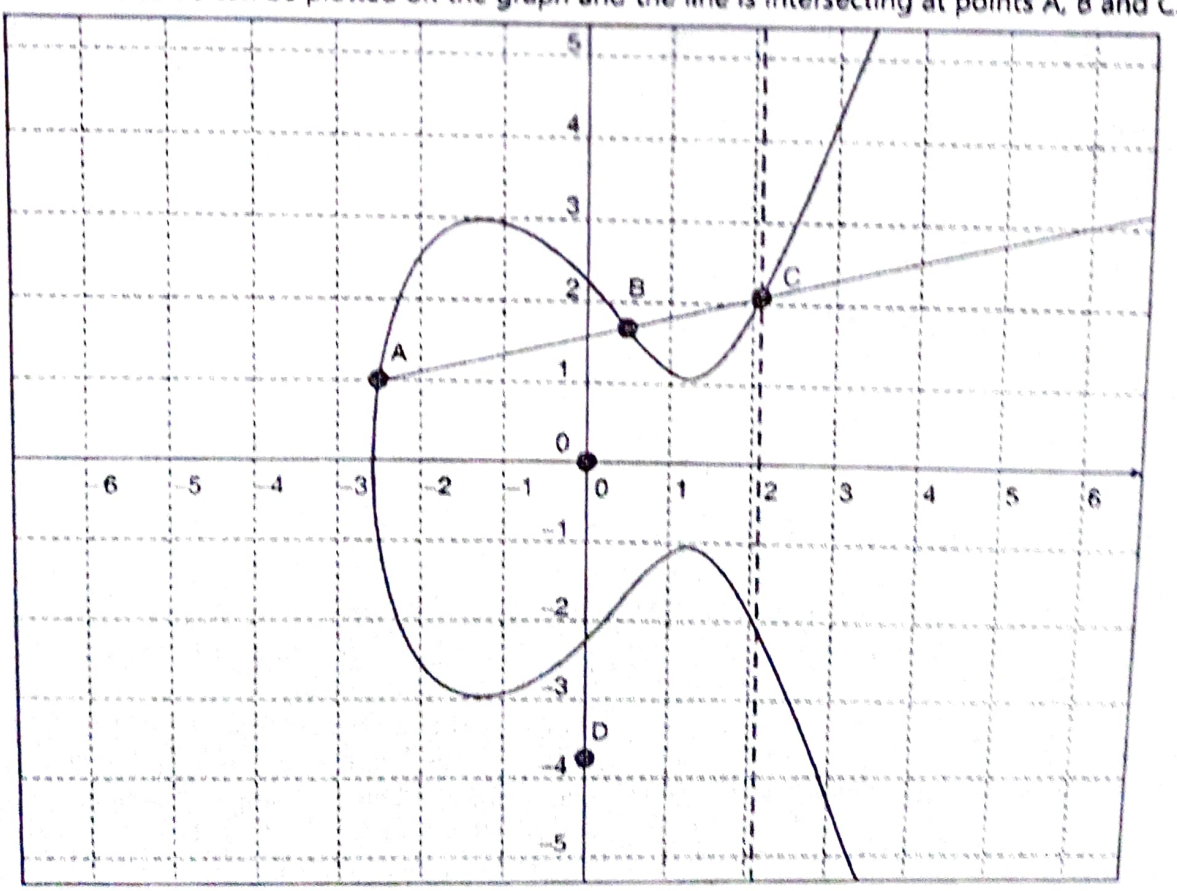

Fig. 1.3.1 : Elliptical Curve Cryptography

- ECC is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. ECC allows smaller keys compared to non-EC cryptography (based on plain Galois fields) to provide equivalent

- ECC is used for key agreement, digital signatures, pseudo-random generators and several integer factorization algorithms.
- Following Table 1.3.1 shows the key length (size) required to provide the equivalent level of security. For example, an elliptic curve cryptography key of 256 bit achieves the same level of security as an RSA of 3072 bit.

Table 1.3.1 : Key length (RSA and ECC)

RSA Key Length (bit)	ECC Key Length (bit)
1024	160
2048	224
3072	256
7680	384
15360	521

Real time use cases under Elliptic Curve Schemes

1. Elliptic Curve Digital Signature Algorithm (ECDSA)
2. Elliptic Curve Diffie-Hellman (ECDH)

(I) Elliptic Curve Digital Signature Algorithm (ECDSA)

The Elliptic Curve Digital Signature Algorithm (ECDSA) is an elliptic-curve variation of the Digital Signature Algorithm (DSA).

ECDSA steps and algorithm :

1. Chose Parameters
   - 1.1 Choose an elliptic curve $E_p(a, b)$: $y^2 = x^3 + ax + b \bmod p$, where $x, y, a, b \in GF(p)$;
   - 1.2 Choose a base point G and get its multiplicative order n, which means $n \times G = 0$, where 0 is the identity element;
   - 1.3 Choose an integer $d \in [1, n-1]$ as the private key, and calculate $Q = d \times G$ as the public key.
2. Sign
   - 2.1 Given a message m → Calculate h(m) and let z be the L_n leftmost bits s of h(m);
   - 2.2 Select $k \in [1, n-1]$ → Calculate a curve point $(x_1, y_1) = k \times G$;
   - 2.3 Calculate $r = x_1 \bmod n$ and $s = k^{-1}(z + rd) \bmod n$;
   - 2.4 The signature is (r, s).

Note that: Q, $E_p(a, b)$, G, n are public, d, k are kept secret.

3. Verify
   - 3.1 Receive a message m and its signature (r,s);
   - 3.2 Calculate h(m) and get z;
   - 3.3 Calculate $u_1 = zs^{-1} \bmod n$ and $u_2 = rs^{-1} \bmod n$;

3.4 Calculate $(x_1, y_1) = u_1 \times G + u_2 \times Q$;

3.5 If $r = x_1 \bmod n$, then the signature is valid.

Correctness : $(x_1, y_1) = u_1 \times G + u_2 d \times G = (zs^{-1} + rs^{-1}d) \times G = (z + rd)\, s^{-1} \times G = k \times G$;

(II) ECDH

- The Diffie-Hellman key exchange algorithm is a mechanism for two people to safely establish a shared secret (Alice and Bob). Elliptic-curve Diffie-Hellman (ECDH) allows two parties to establish the shared secret, each with an elliptic-curve public-private key pair. This shared secret can be used directly as a key or to generate another key.
- Following that, the key, or the derived key, may be used to encrypt subsequent communications with a symmetric-key cypher. It is a Diffie-Hellman protocol variation that employs elliptic-curve cryptography.

Working of Elliptic-curve Diffie–Hellman Key Exchange Algorithm (ECDH)

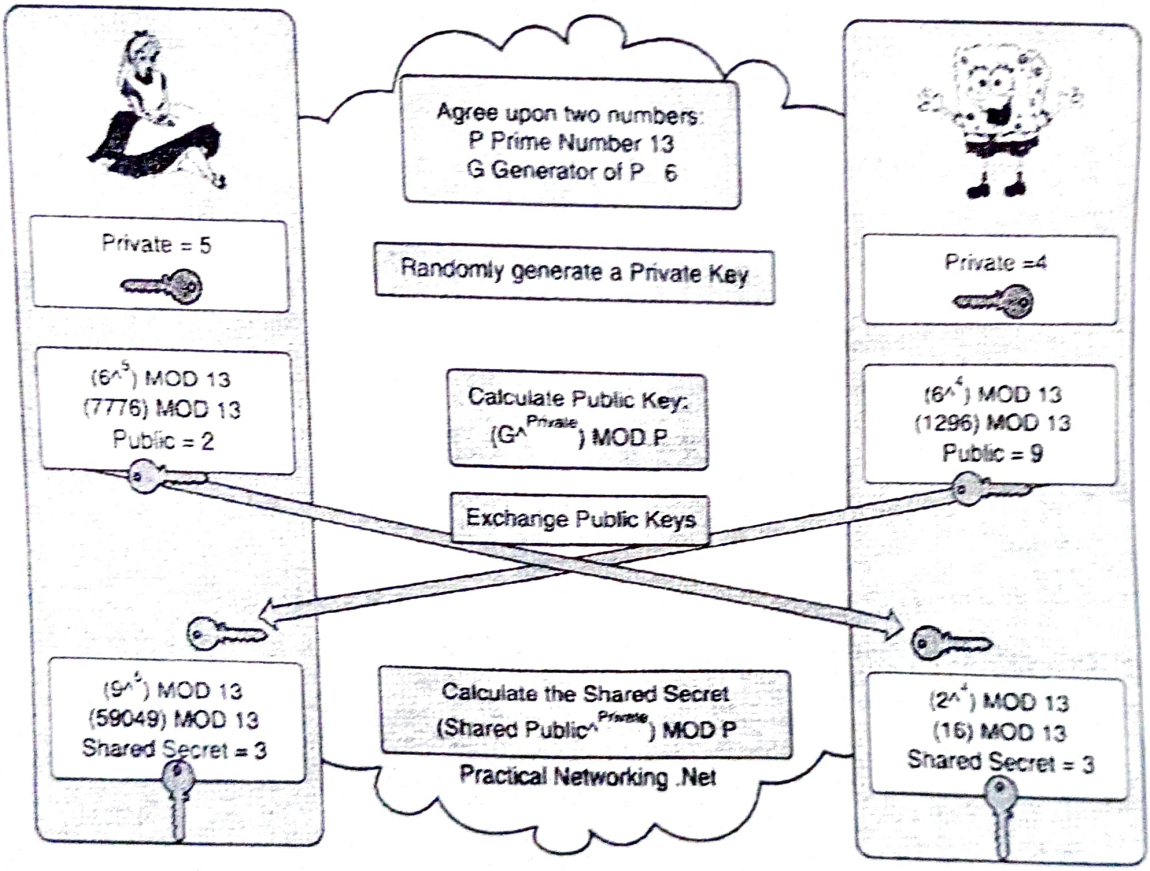

Fig. 1.3.2

1.3.1 Pros and Cons of ECC

Pros of ECC

1. **Shorter Key Length**

   Same level of security as RSA achieved at a much shorter key length

2. **Better Security**

   Secure because of the ECDLP

   Higher security per key-bit than RSA

3. **Higher Performance**

Shorter key-length ensures lesser power requirement – suitable in wireless sensor applications and low power devices

More computation per bit but overall lesser computational expense or complexity due to lesser number of key bits

Cons of ECC

1. **Relatively newer field**

Idea prevails that all the aspects of the topic may not have been explored yet – possibly unknown vulnerabilities

Doesn't have widespread usage

2. **Not perfect**

Attacks still exist that can solve ECC (112 bit key length has been publicly broken)

Well known attacks are the Pollard's Rho attack (complexity $O(\sqrt{n})$), Pohlig's attack, Baby Step,Giant Step etc

1.4 Comparison Symmetric Vs. Asymmetric Key Cryptography

Parameter	Symmetric Key Cryptography	Asymmetric Key Cryptography
Key used	Single key	Public and Private key
Key exchange mechanism	Through some external means	No need to exchange keys
Knowing key	key must not be known to anyone else other than sender and receiver.	Receiver's public key can be known to everyone, but the private key should be known to only the receiver.
Algorithms used	Advanced Encryption Standard (AES) Data Encryption Standard (DES)	RSA Algorithm Diffie-Hellman Key Exchange
Advantages	They are efficient. They take less time to encrypt and decrypt the message.	Robust method. Less susceptible to third-party security breach attempts.
Disadvantages	The number of keys required is very large. In symmetric key cryptography- Each pair of users requires a unique secret key. If N people in the world want to use this technique, then there needs to be N(N-1) / 2 secret keys.	1. It involves high computational requirements. 2. It is slower than symmetric key cryptography.

1.5 Cryptographic Hash Functions

1.5.1 Overview of Hash Function

- A cryptographic hash function is a mathematical function used in cryptography. Typical hash functions take inputs of variable lengths to return outputs of a fixed length. It is a mathematical function that transforms, or "maps" a given set of data into a bit string of fixed size which is known as the "hash value". A cryptographic hash function combines the message-passing capabilities of hash functions with security properties. Hash functions have variable levels of complexity and difficulty.

- Fig. 1.5.2 shows that fixed length output *(n-bit)* is generated when a hash function *h* is applied to input message *m* of arbitrary length. (Fig. 1.5.1)

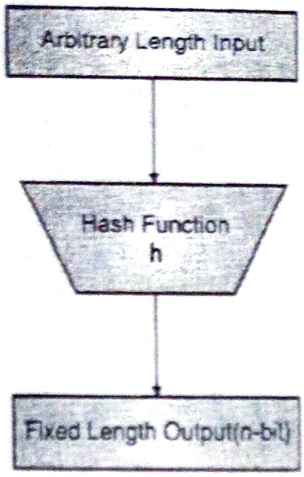

Fig. 1.5.1 : Hash function in cryptography

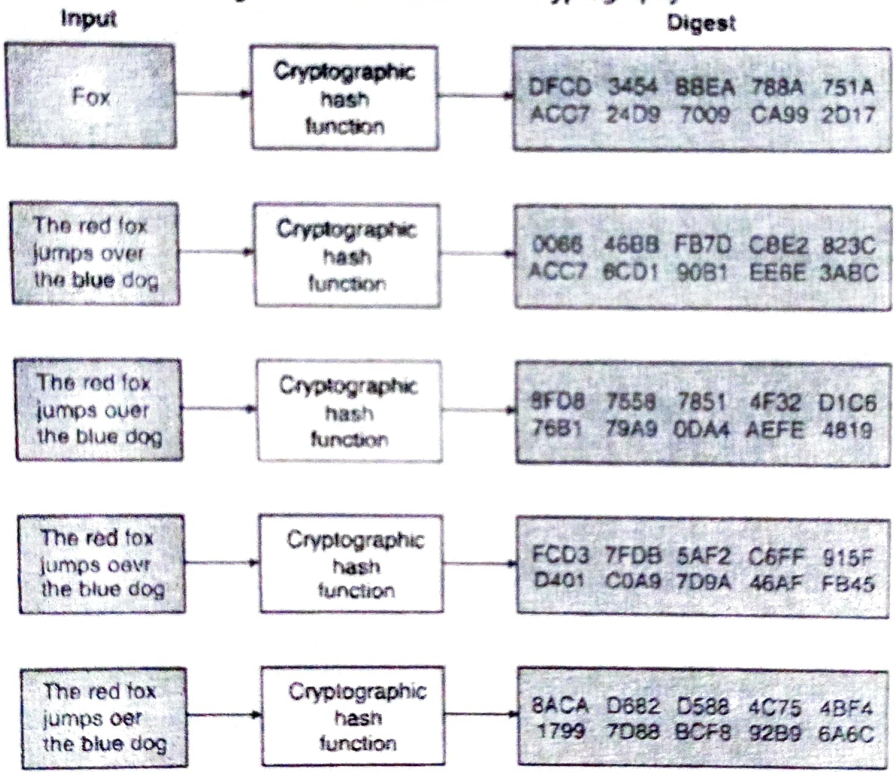

Fig. 1.5.2 : Fixed length output generated from hash function.

1.5.2 Properties of Hash Function

Security level of a cryptographic hash function is defined using following properties :

1. **Preimage resistance :** Given a hash value h, it should be difficult to get any message *m* such that *h = hash(m)*. Functions that lack this property are vulnerable to preimage attacks.
2. **Second pre-image resistance :** Given an input m_1, it should be difficult to find a different input m_2 such that $hash(m_1) = hash(m_2)$. This property is sometimes referred to as weak collision resistance. Functions that lack this property are vulnerable to second-preimage attacks.

3. **Collision resistance :** It should be difficult to find two different messages m_1 and m_2 such that $hash(m_1) = hash(m_2)$. Such a pair is called a cryptographic hash collision.

1.5.3 Applications of Hash Functions

Hash functions are used for :

1. Verifying integrity of message
2. Proof of work in cryptocurrency,
3. password verification and security
4. File, information and data identifier.
5. Signature generation and its verification.

1.6 SHA (Secure Hash Algorithm)

1.6.1 A History of SHA Hashing Algorithms

- Secure Hash Algorithms (SHA) were designed by the National Security Agency (NSA). The U.S. government patented the algorithm, then released it under a royalty-free license so anybody could use it for free. Hashing algorithms are used as one-way functions, meaning that they are designed to be processed one way but are too cumbersome to reverse engineer.
- The SHA-0 algorithm came first in 1993. SHA-1 followed in 1995, and, although it has been cracked, SHA-1 is still in use today. The SHA-2 family, of which SHA 256 is a member, was released in 2001 and includes six hash functions :
  - SHA 224
  - SHA 256
  - SHA 384
  - SHA 512
  - SHA 512/224
  - SHA 512/256
- Each member of the family contains a set of cryptographic hash algorithms that convert data into a unique hash value. In the case of SHA 256, the hash value is 256 bits (equal to 32 bytes). The six hash functions have different numbers of rounds and use different shift amounts and additive constants.

1.6.2 What is the SHA 256 Algorithm?

- SHA-256, which stands for secure hash algorithm 256, is a cryptographic hashing algorithm (or function) that is used for message, file, and data integrity verification. It is part of the SHA-2 family of hash functions and uses a 256-bit key to take a piece of data and convert it into a new, unrecognizable data string of a fixed length. This string of random characters and numbers, called a hash value, is also 256 bits in size.
- SHA 256 is the industry standard used by the U.S. federal government and many other organizations globally.
- Let us consider the following example. Say you write the message "**Good morning**" and apply a SHA-256 hash function to it.

- It will look like this :

  90a90a48e23dcc51ad4a821a301e3440ffeb5e986bd69d7bf347a2ba2da23bd3

- Now, say you decide to do the same with a similar message, "**Good morning!**" It will result in an entirely different string of hexadecimal characters of the same length.
- The following value shows the SHA-256 hash values of two texts that differ by one character :

  c9ebfb6f4b8e880908a737b8d770aa3a518fb6053b327720e8dcc79609c32858

- This example illustrates that adding an exclamation mark changes the entire resulting hash digest.
- As you can see, when just one character is added to the message, the hash value changes completely. The number of characters in the hash values remains the same, regardless of the number of characters in the original text. This helps to hide the size of the original input data because no matter how big or small the input a single word or an entire book it will result in a hash value of the same length.

1.6.3 SHA 256 Terminology

To properly understand SHA 256, we need to grasp the following terms :

1. **Rounds**

   A round is a set of functions repeatedly carried out in the algorithm to scramble the data beyond recognition. There are 64 rounds in the SHA 256 algorithm.

2. **Shift Amount**

   The shift amount is a fixed method used to shuffle bits. In SHA 256, the blocks are divided into eight segments that are 32 bits each. These eight pieces are then shifted in a specific way that scrambles and randomizes the data.

3. **Additive Constants**

   The values added to the blocks are called additive constants. In SHA 256, there are 64 constants used to add to the blocks. These numbers are the cube roots of the first 64 prime numbers. This step uses the first 32 bits of the fractional numbers.

1.6.4 Uses of SHA

We can use SHA 256 in situations where we need the following :

Protecting the Data Integrity

SHA 256 ensures data integrity so that both parties can be sure that the communication is from the person they think it is. The recipient device creates a hash of the original message and compares it to the hash value sent by the sender. If both hash values are equal, the message has not been tampered with during transit.

Verifying Digital Signatures

- A digital signature is a way of signing digital documents, code, or software that is verifiable by the recipient or users. This way, they know whether you created or signed the document or file, or if the item in question was created or altered by someone else.
- A digital signature is created by applying a hash to the file and then using an encryption algorithm to encrypt it via private and public keys. The private key is used when the signature owner signs the document; the public key is used by the recipient to decrypt the message on their end.

- But all of that would mean nothing without verification, which is where SHA-256 comes in. Hashing ensures that the digital signature has not been altered since it was signed. The recipient's system runs the hashing algorithm on its end and uses the public key to decrypt the message. If it matches, then it knows the data is unaltered and authentic.

Verifying Blockchain Transactions

SHA-256 is also used in some popular blockchain applications, most notably the crypto currency Bitcoin. Block headers are integral to blockchains, as they help to "chain" one block of transactions to the next in a specific order. SHA-256 hash helps ensure that no previous blocks are changed without altering the new block's header.

1.6.5 Use Cases of SHA 256

SHA 256 is one of the most reliable algorithms for authentication and message integrity verification. It is used with many different authentication and encryption protocols and processes, including :

- **SSL/TLS** - Secure socket layer (SSL) and transport layer security (TLS) are encryption protocols that maintain data integrity and confidentiality while it is in transit.
- **SSH** - Secure Shell (SSH) protocol creates a secure channel between two devices for data transfer.
- **IPsec** - Internet protocol security (IPsec) is a collection of protocols designed to secure data transfer across different IP networks.
- **PGP** - Pretty good privacy (PGP) is an encryption algorithm used to sign, encrypt, and decrypt emails, files, directories, or a disk partition.
- **S/MIME** - Secure/multipurpose internet mail extensions (S/MIME) is an algorithm to secure the integrity and confidentiality of emails.
- **Blockchain** - In a blockchain, preceding hash values are used to calculate the hash value of the current block.

1.6.6 How Does the SHA 256 Algorithm Work?

SHA 256 is very complex. We will try to understand it from a broad perspective. FIPS-180 describes SHA algorithms as involving two essential stages :

- **Pre-processing** - This is where the message is padded, broken down into smaller blocks, and initialization values are set.
- **Hash computation** - This process involves a series of operations that result in a series of hash values. The 256-bit resulting hash digest we get at the very end is generated by computing these various hash values.
- SHA 256 follows the steps given below :

1. First, data is converted into binary. Binary code uses 0s and 1s to store information. For example, the letter 'a' is written as '01000001' in this basic computer language.
2. The binary data is divided into blocks of 512 bits. If the block is smaller than 512, it will be expanded to that size by adding bits of "padding". If it is larger, it will be broken into blocks of 512 bits. (If the last block is not exactly 512 bits, padding is added to the last block to make it 512 bits.)

3. The message is further divided into smaller blocks that are 32 bits each.
4. Sixty-four iterations (rounds) of compression functions are performed, wherein the hash values generated above are rotated in a specific pattern and additional data gets added.
5. New hash values are created from the output of the previous operations.
6. In the last round, one final 256-bit hash value is produced this hash digest is the end product of SHA 256.

1.7 Introduction to Digital Signature Algorithms

DSA (Digital Signature Algorithm) incorporates the algebraic properties of discrete logarithm problems and modular exponentiations for generating an electronic signature for various applications. It was proposed in 1991 and was adopted as a Federal Information Processing Standard by NIST (National Institute of Standards and Technology) in 1994.

1.7.1 Digital Signature Algorithm (DSA)

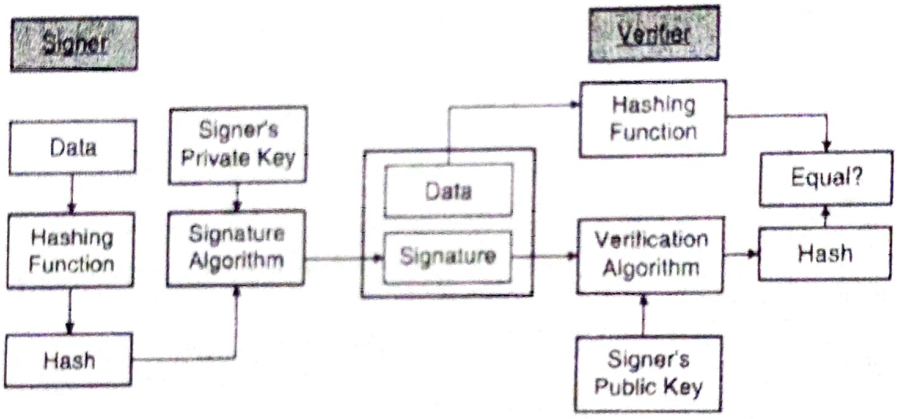

Fig. 1.7.1 : Digital Signature Algorithm

- DSA stands for Digital Signature Algorithm. It is used for digital signature and its verification. It is based on the mathematical concept of modular exponentiation and discrete logarithm. It was developed by National Institute of Standards and Technology (NIST) in 1991.
- It involves four operations :
  1. Key Generation
  2. Key Distribution
  3. Signing
  4. Signature Verification
- Steps at signer's end :
  1. Message or data is hashed using a hashing function to create a digest.
  2. Sender uses his private key to sign the generated digest in step 1 using the signature algorithm.

- Steps at Recipient's end :
  - Recipient on receiving the data and signature runs a verifier algorithm and used sender's public key.
  - If the generated hash value from above step matches with the hash value which was generated from the earlier hashing then recipient is satisfied with the received data and signature. Else if matching fails, then the signature of the sender is not valid.
  - The digital signature process can be divided into 2 parts :

1. **Signature Generation :**
   1. Generating a pair of public keys and providing a key by the sender of the message.
   2. Generating the message digest from the message using a hash function.
   3. Generating the digital signature from the message digest with the private key.
   4. Sending the message, the digital signature, and the public key to the receiver.
2. **Signature Verification :**
   1. Generating the message digest from the message using the same hash function.
   2. Verifying the digital signature with message digest using the public key.
   3. There are 2 popular algorithms that are used to generate and verify digital signature using public keys:
   4. RSA (Rivest, Shamir and Adleman) algorithm developed by Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman in 1976.
   5. DSA (Digital Signature Algorithm) algorithm developed by the US government in 1991.

- The benefits DSA offers are :
  1. **Non-Repudiation :** after signature verification, the sender cannot claim to have not sent the data.
  2. **Integrity** : data modification during transmission prevents final verification or message decryption.
  3. **Message Authentication :** right private/public keys combination help verify sender origin.

1.7.2 Pros of using Digital Signature Algorithm

1. Fast signature computation
2. Requires less storage space for the entire process
3. Freely available (Patent-free) for cost-free global use.
4. Small signature length
5. Observation in real-time
6. Non-invasive
7. DSA is accepted globally for legal compliance.
8. Time-efficient (low time consumption in comparison to processes of physical signing etc.)

1.7.3 Cons of using Digital Signature Algorithm

1. The process does not include key exchange capabilities.
2. The underlying cryptography must be new to ensure its strength.
3. The standardization of computer hardware and software vendors on RSA may cause problems due to DSA's second authentication standard.
4. The complex remainder operations require a lot of time for computation and hence signature verification.
5. It only ensures authentication, not confidentiality, as the algorithm does not encrypt the data.

1.8 Merkle Tree

1.8.1 Concept and Overview

- Merkle trees, also known as Binary hash trees, are a prevalent sort of data structure in computer science.
- In bitcoin and other cryptocurrencies, they're used to encrypt blockchain data more efficiently and securely.
- It is a mathematical data structure made up of hashes of various data blocks that summarize all the transactions in a block.
- It also enables quick and secure content verification across big datasets and verifies the consistency and content of the data.

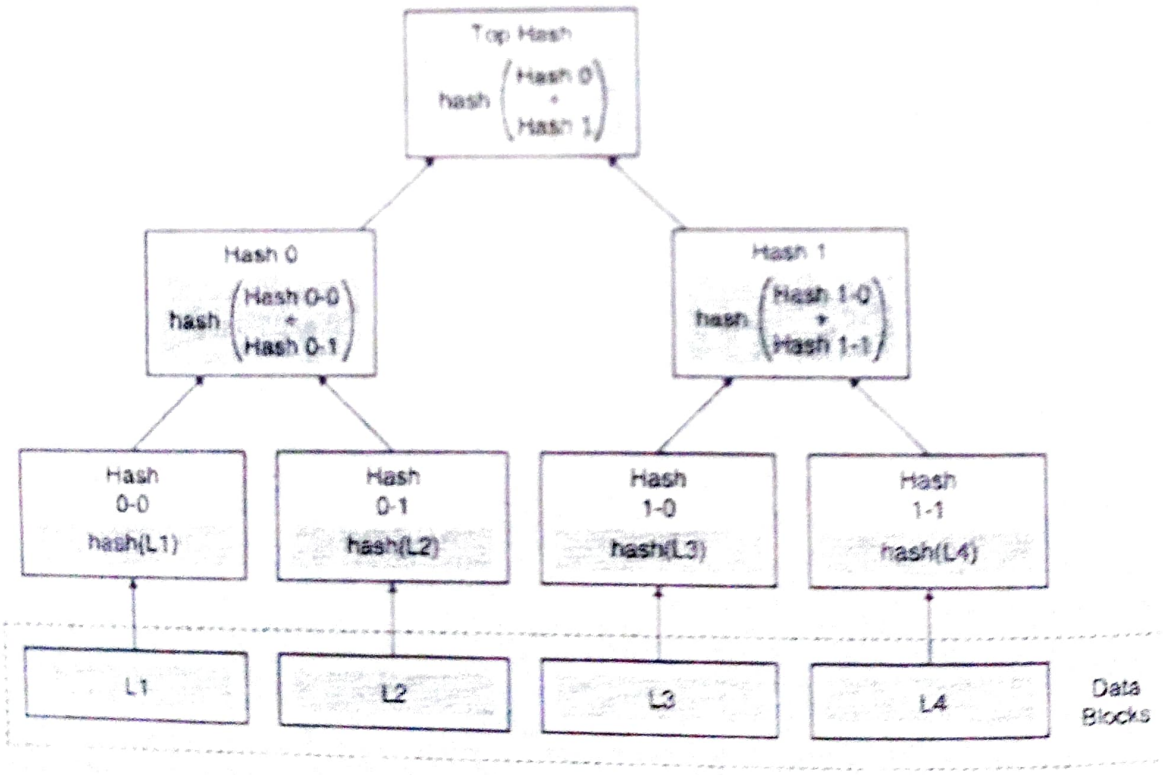

Fig. 1.8.1 : Markle Tree (Binary Hash Tree)

1.8.2 What is a Merkle Root?

- A Merkle root is a simple mathematical method for confirming the facts on a Merkle tree.

undamaged, and unaltered.

- They play a very crucial role in the computation required to keep cryptocurrencies like bitcoin and ether running.

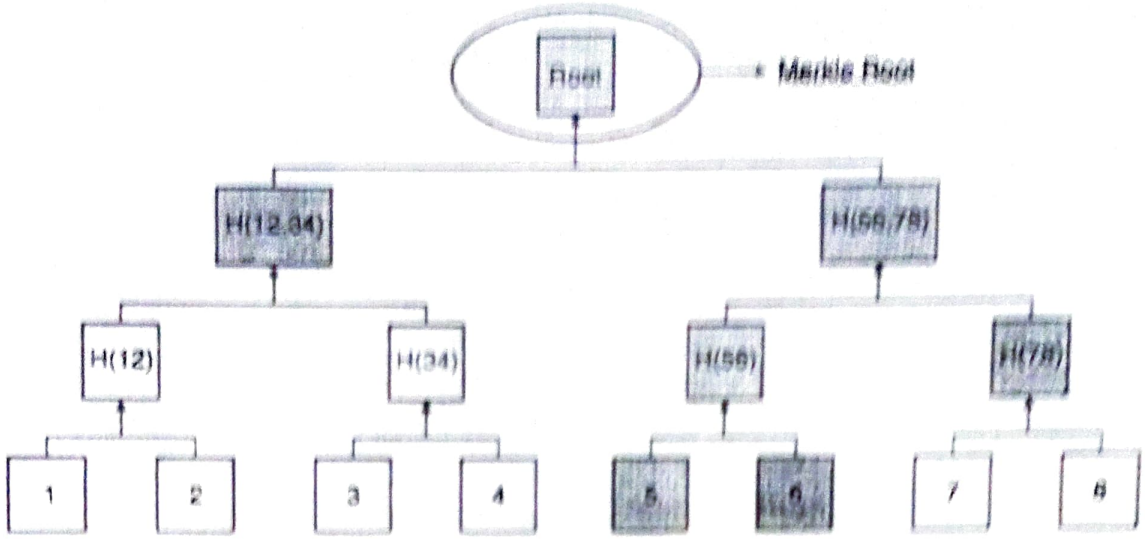

Fig. 1.8.2 : Merkle Root

1.8.3 Working of Merkle Trees

- A Merkle tree totals all transactions in a block and generates a digital fingerprint of the entire set of operations, allowing the user to verify whether it includes a transaction in the block.
  - Merkle trees are made by hashing pairs of nodes repeatedly until only one hash remains; this hash is known as the Merkle Root or the Root Hash.
  - They are built from the bottom, using Transaction IDs, which are hashes of individual transactions.
  - Each non-leaf node is a hash of its previous hash, and every leaf node is a hash of transactional data.
- Now, look at a little example of a Merkle Tree in Blockchain to help you understand the concept.
- Consider the following scenario: A, B, C, and D are four transactions, all executed on the same block. Each transaction is then hashed, leaving you with :
  - Hash A
  - Hash B
  - Hash C
  - Hash D
- The hashes are paired together, resulting in :
  - Hash AB

    and
  - Hash CD

- And therefore, your Merkle Root is formed by combining these two hashes : Hash ABCD.

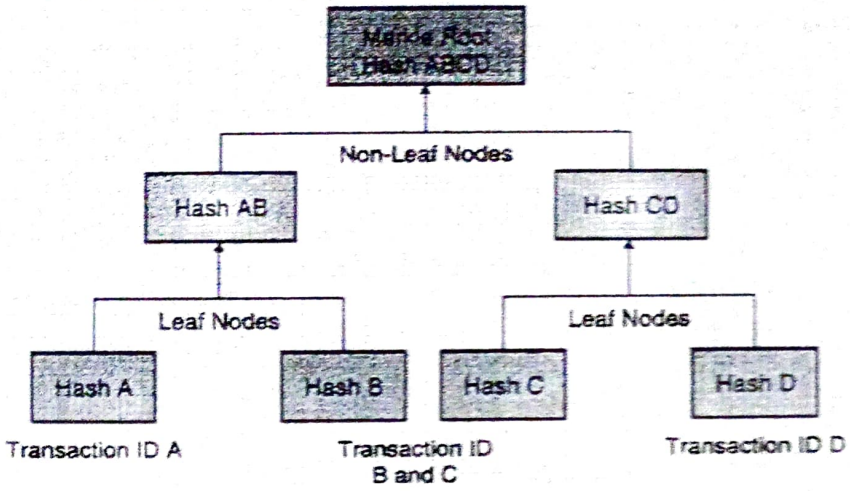

Fig. 1.8.4 : Markle Tree (combine hashes)

1.8.4 Benefits of Merkle Tree in Blockchain

Merkle trees provide four significant advantages :

1. **Validate the data's integrity :** It can be used to validate the data's integrity effectively.
2. **Takes little disk space :** Compared to other data structures, the Merkle tree takes up very little disk space.
3. **Tiny information across networks :** Merkle trees can be broken down into small pieces of data for verification.
4. **Efficient Verification :** The data format is efficient and verifying the data's integrity takes only a few moments.

1.8.5 Use-Cases of Merkle Tree in Blockchain

1. Git, a distributed version control system, is one of the most widely used. It is used to handle projects by programmers from all around the world.
2. Interplanetary File System, a peer-to-peer distributed protocol, is another suitable implementation. It is also open source, allowing computers to join and use a centralized file system.
3. It is part of the technique that generates verifiable certificate transparency logs.
4. Amazon DynamoDB and Apache Cassandra use it during the data replication process. These No-SQL distributed databases use Merkle trees to control discrepancies.

Review Questions

Q.1 Define cryptography & its types.

Q.2 Compare cryptographic methods- asymmetric & symmetric.

Q.3 Steps for RSA & advantages of it.

Q.4 Write in brief about AES & DES (pros and cons).

Q.5 Write in brief about ECC, equation of ECC, Why ECC is preferred over RSA.

Q.6 Compare RSA & ECC.

2 Features of Blockchain

UNIT - II

Syllabus

History, Centralized Vs. Decentralized Systems, Layers of Blockchain : Application Layer, Execution Layer, Semantic Layer, Propagation Layer, Consensus Layer, Why is Block chain important? Limitations of Centralized Systems, Blockchain Adoption So Far.

2.1 Introduction of Blockchain

- Blockchain technology is a distributed, decentralized, immutable ledger that stores the record of ownership of digital assets in a business network. To put it another way, a blockchain is a distributed ledger or database that is shared among the nodes of a computer network.
- It is a mechanism enabling the secure transfer of assets without any need of intermediary. As the internet is a technology that facilitates the digital flow of information, in the same way blockchain is a technology that facilitates the digital exchange of units of value. Anything from currencies to land titles to votes can be tokenized, accumulated, and exchanged on a blockchain network.

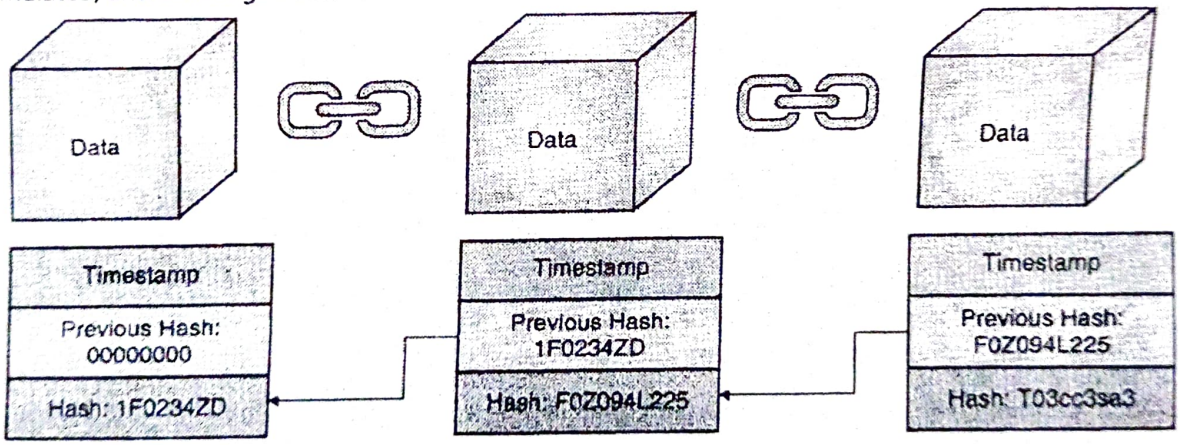

Fig. 2.1.1

- The majority of participants verify every transaction of the system. BitCoin is an example of the blockchain and is the most popular cryptocurrency.
- One significant difference between a typical database and a blockchain is the way the data is structured. A blockchain gathers information together in groups, known as blocks, which hold sets of information. All Blocks have certain storage capacities and, when filled, are closed and linked to the previously filled block, creating a data chain known as the blockchain. All new information that follows that newly added block is compiled into a freshly formed block which will also be added to the chain once filled.

Note :

- Blockchain is not Bitcoin; rather the technology behind Bitcoin is known as Blockchain.
- Bitcoin is the digital cryptocurrency or token on the other hand blockchain is the distributed ledger to keep track of transactions of those digital tokens.
- Blockchain can be used for a number of applications, one of them as Bitcoin.

2.1.1 The History of Blockchain

1991 : The research scientists Stuart Haber and W. Scott Stornetta presented a computationally practical solution for time-stamping digital documents so that they could not be tampered. The concept of a cryptographically secured chain of blocks was used by them to design a system to store the time-stamped documents. To create a 'secured chain of blocks', Merkle Trees are used. It stored a series of data records, and each data record connected to the one prior to it.

1998 : The Scientist, Nick Szabo works on "Bit Gold" decentralized digital currency.

2004 : A computer scientist and cryptographic activist named Hal Finney introduced a system called Reusable Proof Of Work (RPoW) for digital cash. It was a game changing step in the history of cryptocurrencies. The RPoW system worked by obtaining a non-fungible or a non-exchangeable Hashcash based proof of work token in return. He created an RSA-signed token that further could be transferred from one person to another. This System helps others to solve the problem of Double Spending by keeping the ownership of tokens registered on a trusted server.

2008 : Satoshi Nakamoto, wrote a paper named "Bitcoin : A Peer-to-Peer Electronic Cash System". He introduced the term chain of blocks and explained the concept of distributed blockchains. Satoshi Nakamoto's actual identity is unknown. He was active in the Bitcoin developer community till 2011. A peer-to-peer network is used in it, for verifying and timestamping each exchange. It could be managed autonomously without involving a central authority. Due to these improvements, blockchains are considered as the backbone of cryptocurrencies.

2009 : Nakamoto implemented first blockchain as a public ledger, for public transitions made using bitcoin.

2014 : Blockchain evolution is steady and promising and It has became a need in various fields.

Microsoft starts accepting Bitcoin as payments. Many applications are incorporated by the Blockchain technology that can be implemented in various economic sectors. Particularly in the finance sector, significant advancement in the performance of financial transactions.

2.1.2 How does the Blockchain Work?

- A blockchain is a peer-to-peer, distributed database that hosts a constantly growing number of transactions. Each transaction is referred to as a "block," and is secured through timestamped cryptography and is validated by every authorised member of the system using consensus algorithms (i.e., a set of rules). A particular transaction that is not validated by all members of the database is not added to the database. Every transaction is connected to the previous transaction in sequential order, forming a chain of transactions (or blocks). A transaction cannot be edited, or deleted thereby creating an immutable audit trial. A transaction can only be altered by adding another transaction to the chain. Transactions are fixed together in an irreversible chain known as a blockchain.
- Each additional block makes stronger the verification of the previous block and thus the entire blockchain. This makes the blockchain tamper-evident, supplying the key strength of immutability. This removes the possibility of tampering by a bad intentioned actor - and forms a ledger of transactions you and other network members can trust on.

To exemplify this, say that person X wants to send money to person Y for paying an outstanding invoice related to the purchase of software (Fig. 2.1.2). Person X inserts the transaction in the database, thereby creating a block. The transaction (or block) is broadcasted to every authorised member of the network. After validation of the transaction (i.e., approval of the payment) from all the members a block is added to the chain of transactions, which provides a transparent and immutable record of the transaction. The money is then transferred from person X to person Y, and the transaction is finished. The security of the blockchain forbids a hacker from acting as an authorized member of the network.

- All transactions are replicated across the network of users and then accumulated in every member's computer system, enabling a distributed ledger-which can be shared across various organizations, locations, or countries.

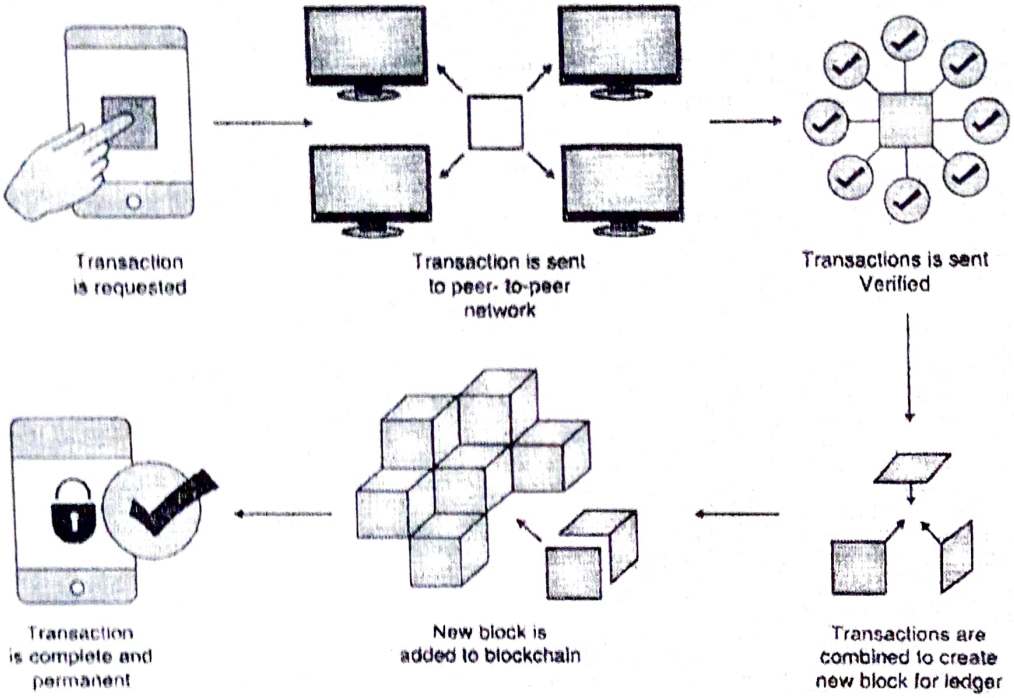

Fig. 2.1.2

2.1.3 Features of Blockchain

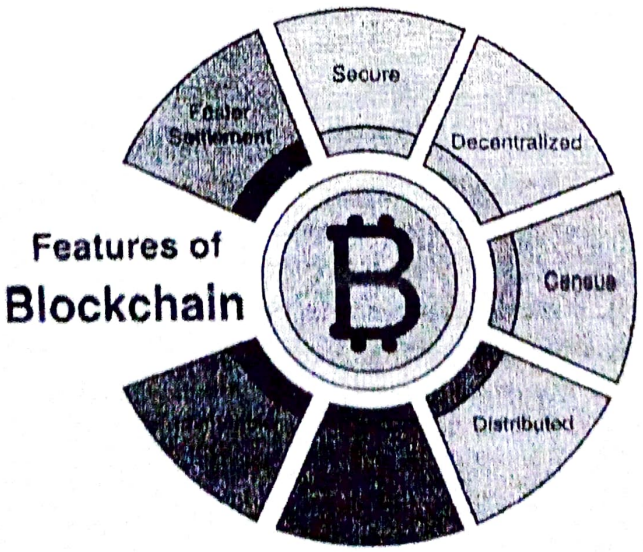

Fig. 2.1.3

1. **Immutable**

- Immutability suggests that the blockchain is an unalterable and permanent network. Creation of immutable ledgers is one of the main values of Blockchain. Blockchain technology operates through a collection of nodes.
- Every node on the system has a replication of the digital ledger. In order to add a transaction validity of every node needs to be checked . It is added to the ledger, only if the majority thinks it is valid. This promotes transparency and corruption proof.
- Once the transaction blocks is added on the ledger, no one can go back and just change it. Thus, no user on the network will be able to edit, update or delete it.

2. **Distributed**

For complete transparency, all network participants have a copy of the ledger. A public ledger will supply complete information about all the participants on the network and transactions. Distributed ledger is one of the major features of blockchains due to many reasons like : In distributed ledger tracking the happenings in the ledger is easy as changes propagate really fast in a distributed ledger. Every node on the blockchain network should maintain the ledger and participate in the validation. The distributed ledger permits anyone with the required access to view the ledger and makes the process reliable and transparent.

3. **Decentralized**

Decentralized technology gives you the power to store your assets in a network without the control and oversight of a single person, entity or organization. Rather a group of nodes makes and maintains the network. In the blockchain network, each and every node has the same copy of the ledger. Decentralization property provides many advantages in the blockchain network : Because of decentralization we have User Control, No single point of failure, No intermediaries, Less Failure, and Zero Scams.

4. **Secure**

- All the records are individually encrypted in the blockchain. Use of encryption adds another layer of security to the complete process on the blockchain network. Since there is no central authority, it means that no one can simply add, update or delete data on the network.
- Every information on the blockchain is hashed cryptographically which signifies that every piece of data on the network has a unique identity. Every blocks contain a unique hash of their own and the hash of the previous block. As a result of this property, the blocks are cryptographically linked with each other. To modify the data means to change all the hash IDs which is nearly impossible.

5. **Consensus**

Every blockchain has a consensus to help the network to perform unbiased and quick decisions. Consensus is a decision-making algorithm for the group of nodes active on the network to reach an agreement quickly and for the system to function smoothly. Nodes might not trust each other although they can trust the algorithm that runs at the core of the network to make decisions. There are lot of consensus algorithms available each with its pros and cons. Every blockchain must have a consensus algorithm or else it will lose its value.

6. **Unanimous**

The records are validated by all the network participants, before they can be added to the network. If a node wants to add a block to the network then it must get majority voting otherwise the block cannot be added to the network. A node cannot just add, update, or delete information from the network. Every record is updated simultaneously and within the network the updations propagate quickly. So it is impossible to make any change without consent from the majority of nodes in the network.

ster Settlement

Traditional banking systems are inclined to many reasons for fallout like taking days to process a transaction after finalizing all settlements, which can be easily corrupted. On the contrary, blockchain offers a faster settlement compared to traditional banking systems. This blockchain feature makes life easier.

Blockchain technology is improving and increasing day by day and has a really bright future in the forthcoming years. The trust, transparency, and temper proof characteristics gave rise to many applications of it like Ethereum, bitcoin, etc. It is a pillar in making the business and governmental procedures more efficient, secure, and effective.

4 Block Architecture

The architectural components have been generalized and then modified by numerous companies, leading to different blockchain projects like Bitcoin, Hyperledger, Ethereum, etc. Let's consider the bitcoin blockchain architecture here.

Below are the architectural components :

- Transaction
- Block
- P2P Network
- Consensus Algorithm

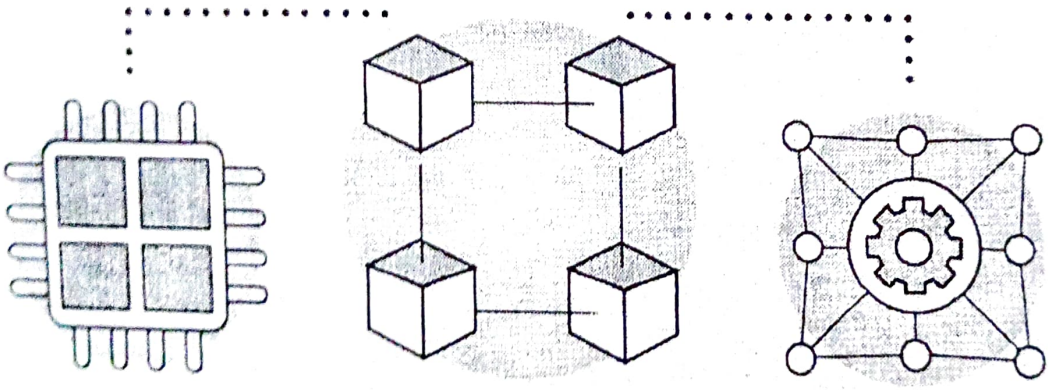

Fig. 2.1.4

Transaction

- The smallest building blocks of a blockchain system are Transactions. They normally comprise of a recipient address, a sender address, and a value. It is identical to a standard credit card statement. The owner transfers the value by digitally signing the hash generated by adding the previous transaction and the receiver's public key.
- The transaction is then publicly declared to the network and all the nodes hold their own copy of the blockchain independently, and the current known "state" is calculated by processing each transaction in the same order as it appears in the blockchain. Transactions are packed together and delivered to each node in the block form. As new transactions are distributed throughout the network, they are independently processed and verified by each node. Every transaction is time-stamped and collected in a block.

ock

Block holds the information as a block header and transactions. Blocks are data structures whose motive is to bundle sets of transactions and are copied to all nodes in the network. Miners create the blocks in the blockchain. Mining is the process of creating a valid block that will be accepted by the rest of the network. Nodes take pending transactions, verify if they are cryptographically precise, and bundle them into blocks to be stored on the blockchain. Block header is the metadata that assists in verifying the validity of a block. The contents of a block metadata is shown in the Fig. 2.1.5.

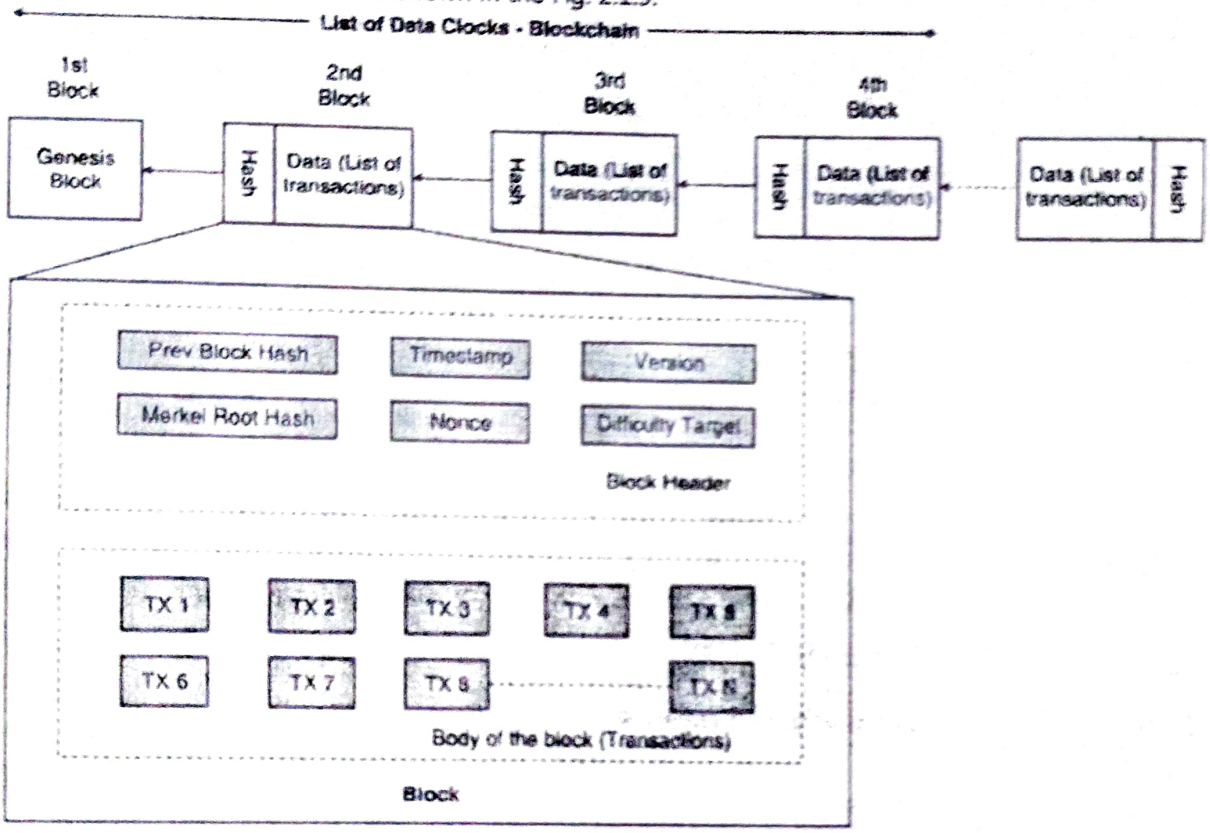

Fig. 2.1.5

A block stores information. Within a block there are many pieces of information, but it doesn't use a large amount of storage space. Blocks generally include these elements, but it might differ between different types

- **Blocksize :** It sets the size limit on the block ensuring that only a specific amount of information can be written in it.
- **Block header :** Includes information about the block.
- **Transactions :** A list of all the transactions within a block.

The transaction element contains the most information hence it is the largest. It is accompanied in storage size by the block header, which includes these sub-elements :

- **Version :** The version of cryptocurrency being used.
- **Previous block hash :** Contains an encrypted number or hash of the previous block's header.
- **Hash Merkle root :** Hash of the transactions in the Merkle tree of the current block.
- **Time :** A timestamp to arrange the block in the blockchain.
- **Bits :** The rating of difficulty of the target hash, denoting the difficulty in solving the nonce.
- **Nonce :** The encrypted number that a miner must solve in order to verify the block and close it.

- The rest of a block contains transactions. Depending on the choice of a miner, it can be any number of transactions bundled in a block.
- The first block of the chain is called the Genesis block.

3. P2P Network

- Blockchain is a Peer to Peer (P2P) network working on the protocol known as IP protocol. A P2P network is a flat topology with no centralized node. All nodes equally provide and can utilise services while collaborating using a consensus algorithm. Peers contribute to the storage and computing power that is required for the maintenance of the network. P2P networks are usually more secure because they do not have a single point of failure or attack as in the case of a centralized network.
- A blockchain network can be a permission-based network and a permissionless network also. A permissionless network is also called a public blockchain because anyone can join the network, while a permission-based blockchain is called a consortium blockchain. Pre-verification of the participants within the network is required in a permission-based blockchain or private blockchain and the parties are usually known to each other. In a typical blockchain architecture, every individual node in a network preserves a local copy of blockchain. The decentralization of blockchain architecture is the individual credit of the P2P network that it is built on.

4. Consensus Algorithm

All these copies of a single ledger are synchronized due to a consensus algorithm. The consensus mechanism makes sure that the local copy every individual party has, they are consistent with each other and is the most updated one. The copy that every individual node has are identical to each other. It could be arguably framed that the consensus algorithm forms the core of every blockchain architecture.

2.2 Centralized Vs. Decentralized Systems

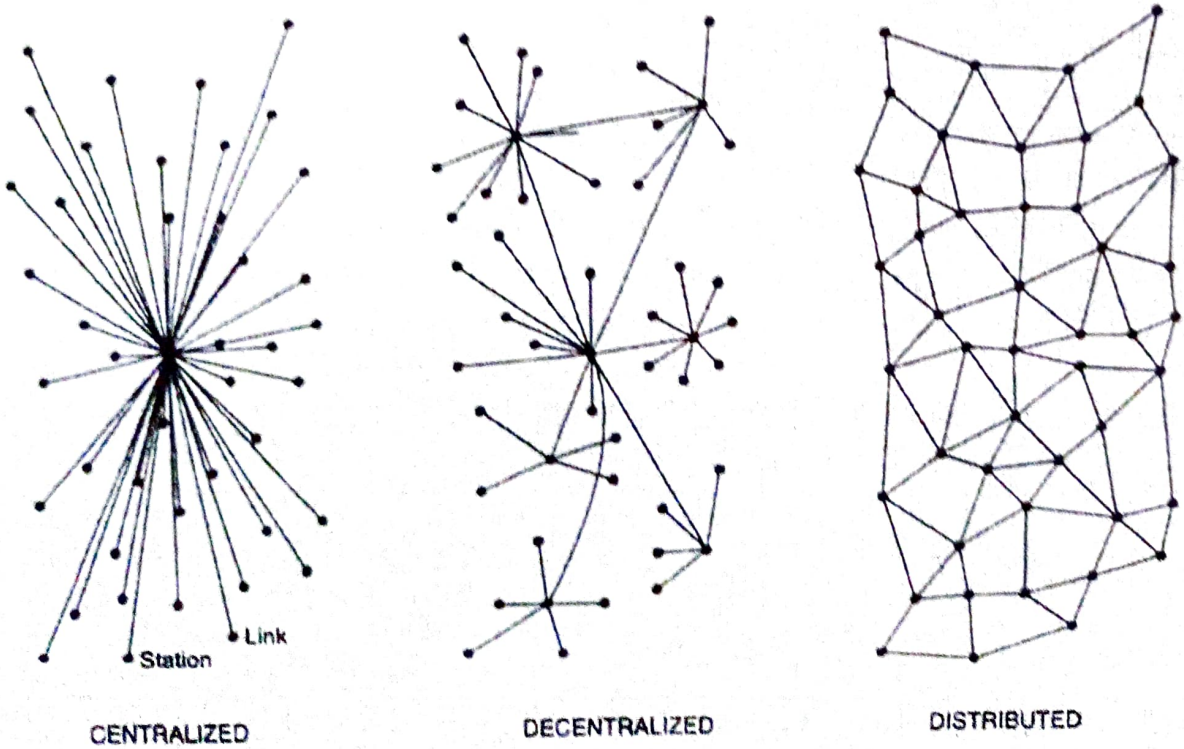

Fig. 2.2.1 : Different types of Networks/Systems

2.2.1 What does Centralization Mean?

- Centralized systems are traditional (client-server) IT systems in which a central authority is in control of the functions and data for the platform which is in use, and who is entirely in charge of all operations on the system. All users of a centralized system rely on a single source of service. A centralized authority controls the majority of online service providers like Google, Amazon, eBay, Apple's App Store, Facebook, YouTube, Twitter, and others like your bank account.
- For example, if you are using the Facebook platform, then Facebook has complete control over the various aspects of their features including the decision of who and who cannot join the platform. This means to verify a data transaction, a third-party intermediary must do this on your behalf. This means that using the Facebook platform, if you are sending a message to your friend, then the data will be verified and then transferred by the said platform.
- Another example is sending an email using any mail domain. The instant you send an email to another person, the email service provider is aware of what you sent and when you sent it. This information is accumulated privately without any identifier, but a copy of that information is with the email service provider, in any case. This means that you must trust that your data is kept private by the email provider.
- Centralized systems are traditional (client-server) IT systems in which there is a single authority that controls the system, and who is entirely in charge of all operations on the system. All users of a centralized system rely on a single source of service. The majority of online service providers like Google, Amazon, eBay, Apple's App Store, and others use this conventional model for delivering their services.

Advantages of Centralization

Some advantages of centralization are as follows :

- **Reduced Costs :** As centralized networks or organizations are pre-planned and setup is ready so the costs associated with it do not exceed budgets. The cost will be increased only when expansion of the network is required. So the biggest advantage of centralization is the cost associated with it.
- **Command Chain :** In the centralized organization, the chain of command is known by them. That is every person in the organization is aware of their role and whom do they need to report to. They are aware of whom they are reporting and their subordinates and actions too. This also means that delegation is effortless in the chain. Senior executives can finish the work in the best possible way by easily delegating work to their subordinates and finalizing. If work is successfully completed, it creates a trust among the workers and chain, improving the confidence necessary to make it work. One central node or a collection of nodes are responsible for transactional verification in a centralization network.

Quick Decision Implementation : In centralization organizations or networks the decision implementation are quick. As centralized networks have less nodes or people, it requires fewer communication among the different levels of authorization.

Also, if a centralized network decides to implement a change, it can be finished in a matter of minutes. For example, a centralized network can put more stress on the KYC procedure and determine to add more requirements for it. As the network is centralized, they can push the new guidelines or modify the KYC procedure which can go live almost immediately after proper testing.

Disadvantages of Centralization

Some disadvantages of centralization are as follows :

- **Security** : For any system security is a nightmare. In the centralized system, it is a major issue since all the data is stored in one location and all operations are executed through a central server which makes it an easy victim for various types of attacks especially to Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The enormous increase of IoT devices in our environment accelerates the chances to exploit security vulnerabilities within IoT devices which are poorly secured. Therefore, both IoT devices (source of data) and centralized cloud server (data storage location) are an easy prey for various security attacks.

- **Trust** : The centralized organizations are not completely trustable or secure. The trust is an agreement that is set by the user and the service provider. Nevertheless, that's an agreement and it can break easily.

  When in the system there is a lapse of security Big corporations experience trust issues from their users, once in a while. It happens when people tend to ignore the service for some time before the service provider fix the trust by offering solutions and payment to those affected. Because all the data is stored in a centralized database. Above things are happening in a centralized system.

- **Privacy** : The centralized system is vulnerable to data manipulation. Collecting real-time data of different devices and storing it at one place with the authority of the centralized server can violate the data privacy. The collected data may contain sensitive information about nodes such as their passwords, financial accounts, etc. As these data are stored in one place, it can be easily violated. On the contrary, there are various examples of privacy violation by service providers. For example, information about their customers is sold by some service providers to marketing companies that can use this information to examine nodes' behavior. Also, if a public utility company found that their smart meter data analysis might be the evidence that will result in lawsuits or high costs. They will alter or even delete these data. Hence, privacy is another issue in the centralized IoT system that needs to be tackled.

- **Single point of failure** : Due to single point where the whole network is dependent on the Centralized organization and even if Single point of failure occurs creates a problem for it. Organizations know about the disadvantage and hence have arranged measures to contain it. For mission-critical services, the fact that there is a chance for failure is a big disadvantage.

- **Scalability Limitation** : In most of the cases, a single server is used, which leads to scalability limitations. For the centralized system, it is a major issue since it based on controlling and managing all processes using a central authority. This system can scale well but only for small networks. It will be impractical, deploying a centralized system for large business organizations with many branches in different locations. Based on the management hierarchy it will be hard to transport decisions to different locations. In the IoT context, since there is a massive increase in the number of IoT devices, there are many uncertainty about the capability of the centralized architecture of the IoT to scale and function efficiently with the increasing demands.

- **Inflexibility** : There are huge workloads coming from different nodes in the network, since the centralized server carries out all processing operations and controls all nodes connected to the network. Although the centralized server schedules the workload to prevent peak-load concerns when people across an organization need to use it simultaneously, the pack schedule and delay associated with this process, restrict the flexibility of the user while doing their own work.

- **Access and Diversity :** The network can accessed by the nodes for different needs. However, centralized systems need their nodes to access the information on the network consistently using identical processes. This kind of network may not give the flexibility needed by several nodes with different needs. In addition, a single operating system is used by a centralized system for the whole network. While this can benefit some nodes, it restricts diversity within the network and can forbid some nodes from accessing the network. Since the IoT system is heterogeneous and dynamic in nature with different devices and objects, ensuring access to various heterogeneous devices should be a basic priority for the centralized IoT architecture.

2.2.2 What is Decentralization?

- A decentralized system is a type of network in which the nodes are not dependent on a single master node; instead, control is distributed among many nodes. This is similar to a model in an organization where each department is in charge of its own database server, thus eliminating the power of the central server and dispersing it to the sub departments who manage their own databases. The fundamental idea of decentralization is to distribute authority and control to the peripheries of an organization. This configuration provides various benefits for organizations, such as enhanced efficiency, accelerated decision making, better motivation, and a reduced load on top management.

Decentralization was initially made possible using blockchain technology. A significant innovation in the decentralized paradigm that has induced this new era of decentralization of applications is decentralized consensus. Bitcoin client was the first-ever blockchain, which was created in 2009. Using a consensus algorithm Bitcoin enables a user to agree on something, without the necessity for an intermediary, central, trusted third party or a service provider. A consensus mechanism governs this competition, and the most commonly used method is known as Proof of Work (PoW).

When a person sends Bitcoin to someone else, transactions are not verified by a centralized authority. Rather, anybody can hook up their computer to the Bitcoin system to help verify a movement of funds. Every device connected to the system is known as a "node", and in total there are thousands of independent nodes all of which are helping to operate the network.

From a semi-decentralized model to a fully decentralized one, Decentralization is applied in varying degrees depending on the circumstances and requirements.

From a blockchain perspective Decentralization can be viewed as a mechanism that provides a way to remodel existing paradigms and applications, or to create new applications, for the sake of giving full control to the users.

dvantages of Decentralisation

Provides a trustless environment : No one has to trust or know anyone else in a decentralized blockchain network. Within the network every member has a copy of the exact same data in the form of a distributed ledger. If any member's ledger is corrupted or altered in any way, majority of the members in the network will reject it.

Improves data reconciliation : Many companies often exchange data with their partners. This data, is typically transformed and stored in each party's data storage place, only to resurface when it needs to be passed downstream. Each time the data is transformed, it opens up possibility for data loss or incorrect data to enter the work stream. Every entity has access to a real-time, shared view of the data by having a decentralized data store.

Reduces points of weakness : In systems Decentralization can reduce points of weakness where there may be too much dependence on specific actors. These weak points could lead to integral failures, including failure in providing promised services or inefficient service due to the depletion of resources, periodic interruptions, bottlenecks, lack of sufficient incentives for good service, or corruption.

Optimizes resource distribution : Decentralization can also help optimize the distribution of resources so that the promised services are provided with consistency and better performance, as well as a reduced probability of catastrophic failure.

Decentralization usually has some disadvantages such as lower transaction output, but preferably, the trade offs are worth the improved stability as well as the service levels they produce.

Decentralization should be put in application where it makes sense. It doesn't mean that just because it's a blockchain application, it needs to be 100% decentralized.

Information and Communication Technology (ICT) has conventionally been based on a centralized paradigm whereby a central authority, such as a system administrator controls the database or application servers. With Bitcoin and the emergence of blockchain technology, this model has transformed and now the technology exists, which allows anyone to start a decentralized system and operate it with no single trusted authority or no single point of failure. It can either be run independently or by requiring some human intervention, depending on the model and type of governance used in the decentralized application administrating on blockchain.

2.2.3 What is a Distributed System?

- A distributed system, computation and data are spread across multiple nodes in the network. At times, this term is confused with parallel computing. While there is some intersection in the definition, the main difference between these systems is that in a parallel computing system, computation is executed by all nodes simultaneously for the sake of achieving the result; for example, parallel computing platforms are used in simulation and financial modeling, weather research and forecasting.
- On the contrary, in a distributed system, computation may not happen in parallel and data is duplicated across multiple nodes that users view as a single, coherent system. Variations of both of these models are used to achieve speed and fault tolerance. In the parallel system model, there still is a central authority which governs processing and has control over all nodes.

Difference between Centralized, Distributed and Decentralized

	Centralized	Distributed	Decentralized
Network/hardware resources	Maintained & controlled by single entity in a centralized location.	Spread across multiple data centers & geographies; owned by network provider.	Resources are owned & shared by network members; difficult to maintain since no one owns it.
Solution components	Maintained & controlled by central entity	Maintained & controlled by solution provider.	Each member has exact same copy of distributed ledger.
Data	Maintained & controlled by central entity	Typically owned & managed by customer.	Only added through group consensus.

	Centralized	Distributed	Decentralized
Control	Controlled by central entity	Typically, a shared responsibility between network provider, solution provider & customer.	No one owns the data & everyone owns the data.
Single Point of Failure	Yes	No	No
Fault tolerance	Low	High	Extremely high
Security	Maintained & controlled by central entity	Typically, a shared responsibility between network provider, solution provider & customer.	Increases as # of network members increase.
Performance	Maintained & controlled by central entity	Increases as network/hardware resources scale up and out.	Decreases as # of network members increase.
Example	ERP system	Cloud computing	Blockchain

- This implies that the system is still centralized in nature. The critical difference between a decentralized system and distributed system is that in a decentralized system, no central authority exists, whereas in a distributed system, there still exists a central authority that governs the entire system.

2.2.4 Distributed Ledger

- The distributed ledger created using blockchain technology is unlike a traditional network, as it does not have a central authority common in a traditional network structure (see the Fig. 2.2.2). Decision-making is generally done by the central authority, who decides in all aspects. The traditional database structure therefore is controlled by the central authority.

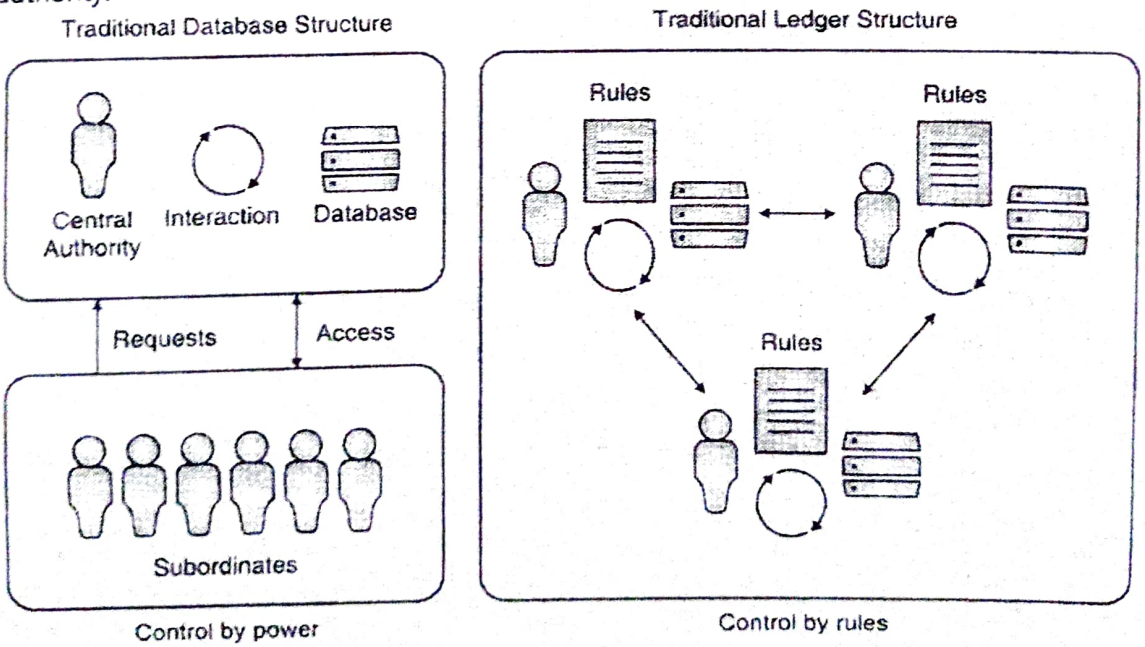

Fig. 2.2.2 : Traditional Database vs. Distributed Ledger

- Having layers of abstraction not only helps to understand the stack better but also helps to build products that meet the stack to accomplish an open system. The layers are abstract from each other, making the system easy to maintain and more robust. Changes made to one of the layers does not affect the other layers. Similarly we use layered architecture in the blockchain.
- The blockchain technology is based on a layered approach Fig. 2.3.1. The block- chain technology is decomposed into different layers that will in turn help in better understanding of the design of the blockchain and security. There are some of the layers for the blockchain technology discussed in the following sections.

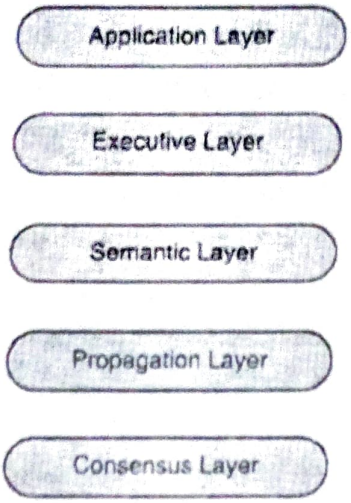

Fig. 2.3.1

2.3.1 Application Layer

- Multiple Applications can be built on a blockchain technology, as we know it is a decentralized, tamper proof, and shared ledger technology. Application layer contains the applications that are used by end users to interact with the blockchain network. Some applications built in the application layer can interface with the other layers; thus, the application layer is on the top of this layer.
- The application layer is the one wherein a user can code the desired functionality and can build the application. The application needs to be installed on each node, since the blockchain technology is a decentralized technology and there is no server involved. It neither has a client-server model, nor does it have a shared server to access the server-client, and this is exactly how Bitcoin works. Even though there are some instances where blockchain is in the back-end and the applications need to be hosted on a web server and requires server-side programming, it would be better if there were no server involved in the blockchain network as it would defeat the purpose and benefit of blockchain technology.
- Therefore, your target needs to create an application for these final outcomes, which often involves traditional technologies for software development such as scripting, programming, embedded development, APIs and so on. It may require the development of server software, online pages, and APIs, among other benefits.

2.3.2 Execution Layer

- This layer operates the executions of all the instructions that were performed at the application layer for every node present on the blockchain network. This layer has the actual code and rules that are executed. The set of instructions could vary from simple ones to multiple instructions. The Execution layer consists of underlying rules, smart contracts and chain code.

- For example, when some funds need to be transferred from one person to another, a smart contract which is a small code needs to be executed. Now if one application is existing on all the nodes of the blockchain network the code needs to be executed independently on all the nodes. To prevent the inconsistencies in the output, the execution of code on a set of input should always produce the same output for all the nodes that are present on the blockchain. In case of Bitcoins, these are simple scripts which are not complete by Turing and only allow some of the instructions.
- On the contrary, complex executions are allowed in Hyperledger and Ethereum. Ethereum code or written smart contracts are compiled into machine code or Bytecode, which is implemented on its own Ethereum virtual machine. A much simpler approach is used by Hyperledger to its smart chain code contracts. It supports multiple high-level words such as Java and Go and supports the execution of compiled machine codes in docking images.
- Applications send instructions to the execution layer (chaincode; for Hyperledger fabric), which does the execution of transactions and ensures the deterministic nature of the blockchain (such as permissioned blockchain like hyperledger fabric).

2.3.3 Semantic Layer

- The semantic layer is a logical layer because there is order in blocks and transactions. This layer deals in validation of the blocks being generated in the network and also validating the transactions performed in the blockchain network. When a transaction comes up from a node, the set of instructions are executed on the execution layer and are validated on the semantic layer. Semantic layer is also in charge for the linking of the blocks created in the network. As we have heard that each block in the blockchain contains the hash of the previous block except the Genesis block. This linking of blocks needs to be defined on this layer.
- In the case of bitcoin, in this layer, it is checked whether the execution of this transaction is authorized or a double-spending attack or is it carrying out a legal transaction, etc. Bitcoin is existing as a transaction that represents the state of the system. In order to be able to spend bitcoin, you must spend one or more previous transactions, and there is no concept of an account. It means that when someone executes a transaction, they utilize one of the previous transactions in which they have received at least the amount they are now spending.
- Every node must verify this process by going through previous transactions to see if it is a valid business. Ethereum, on the other hand, has an account system, which means that the recipient's account and the seller's account have been upgraded. The rules of the system can be defined, in this layer, such as data components and structures. There may be somewhat more complex situations compared to simple exchanges. Often, complex instruction sets are encoded in smart contracts. The system state is updated when the smart contract is requested upon the transaction receipt. A smart contract is a special account with private states and executable code. A block usually has several smart contracts and several bundles of transactions.
- Merkle tree data structures are defined in this layer, with the Merkle root in the block header to maintain the connection between the block header and the set of transactions in the block (usually key-value storage on disk). Storage methods, Computer models, memory / disk-based processing, etc., can be defined in this logical layer. Furthermore, the semantic layer defines how blocks are connected. Every block in the blockchain includes a hash from the previous block, all the way to the genesis block. Even though the ultimate blockchain status is achieved through contributions from all levels, the blockchain connection must be explained through this layer. You may want to cite another application in this list, depending on the application case.

2.3.4 Propagation Layer

- The back layer is more of a special phenomenon having no more interaction with other nodes in the system. A propagation layer handles the peer-to-peer communications between the nodes that allow them to discover each other and get synced with some other node in a network. When a transaction is executed, it gets broadcasted to all other nodes in the network. Also, when a node proposes a block, it will immediately get broadcast so that other nodes in the network can use this newly created block and work upon it.
- Therefore, this layer determines the spread of transactions/blocks in the network, which ensures the entire network's stability. Most blockchains are designed in such a way that they forward a transaction/block immediately to all nodes they are directly connected to in the network when they learn about a new transaction/block. Hence, this layer defines the propagation of the block or a transaction in the network and ensures the stability of the complete network.
- However, depending upon the network bandwidth or network capacity sometimes the propagation could occur immediately, sometimes it may take a longer. On the asynchronous Internet network, Transaction delays or deadlocks are common. Several propagations take a few seconds, and others take longer, depending on network bandwidth, node capacity, and several other factors.

2.3.5 Consensus Layer

- This layer is the foundation layer for most of the blockchain systems. The main aim of this layer is to make sure that all the nodes must agree on a common state of the shared ledger. The layer also deals with the security and safety of the blockchain. There are many consensus algorithms which can be applied for generation of cryptocurrencies like Ethereum and Bitcoin, they use proof-of-work (PoW) mechanisms to choose a node randomly out of various nodes present on the network that can propose a new block.
- Once the block is proposed and grown by all the nodes, it will be checked whether the block is a valid block with all legal transactions and the PoW problem has been solved correctly. Then, they will increase this block to their copy of the blockchain and build depending on this basis. There are many different types of the consensus protocol, such as Proof of Stake (PoS), Delegated PoS (dPoS), Practical Byzantine Fault Tolerance (PBFT), etc.

We can re-divides the above five layers into four Layers, with the bottom to top being **Layer 0, Layer 1, and Layer 2, Layer 3.**

1. **Layer 0 :** The components required to make blockchain as hardware, protocols, connections, and other components that form the foundation of a blockchain ecosystem.Interchain operability is also enabled by Layer 0, which allows blockchains to communicate with one another.Layer 0 provides the underlying infrastructure for blockchain. Layer 0 often employs a native token to enable participation and development. For example: Polkadot, Avalanche, and Cosmos are Layer 0 blockchains.
2. **Layer 1 :** Layer 1 blockchain is an advancement in layer 0.The Layer 1 blockchain is a collection of solutions that improves the Layer 0 blockchain. It has its own Consensus mechanisms. Under this layer, the blockchain network is maintained functionally. However, scaling is a limitation in the layer one blockchain. Any changes and issues arising in the new protocol in layer 0 will also affect layer 1. It is also called an implementation layer. Layer 1 symbolizes the actual blockchain. The large number of jobs that this tier must manage frequently causes scalability problems. As more individuals enter a blockchain, the amount of computational power required to solve and add blocks to the chain grows, resulting in higher fees and longer processing times.

Examples of layer one blockchain are Bitcoin, Ethereum, Cardano, Ripple, etc.

3. **Layer 2** : Layer 0 has many interactions that have been removed by layer 2. To enhance the blockchain's productivity, extra processing power is required.For specific blockchains layer, 2 is the scaling solution. As a result, layer 1 cannot be enlarged without relocating all processing to a second layer created on top of the first that is layer 2. This is made feasible by allowing third-party solutions to be integrated with layer 1. It works with third-party integration and removes the limitations of layer 1. It is by far the most popular approach for solving scaling issues attached to POW networks. At present, various industries have begun implementing layer two technologies.

   For example, consider the Lightning Network as an example of a layer 2 blockchain deployed on the Bitcoin blockchainor Polygon is a Layer-2 Scaling solution that runs on top of Ethereum and it solves the scalability issue in Ethereum

4. **Layer 3** : The last layer of the blockchain ecosystem and the one visible to the human eye. Layer 3 blockchain is also referred to as the "application layer". The main task of this layer is to host the DAapps and many other protocols that enable other apps. Here, the blockchain protocol is split into two significant sub-layers, that being, application and execution. Layer 3 not only provides UI, but also utility in the form of intra- and inter-chain operability, such as decentralized exchanges, liquidity provisioning, and staking applications.

2.4 Why Blockchain is Important

- Business runs on information. The faster it's received and the more accurate it is, the better. Blockchain is ideal for delivering that information because it provides immediate, shared and completely transparent information stored on an immutable ledger that can be accessed only by permissioned network members.
- Blockchain technology provides one of the most secure and safe online transactions so the field of Blockchain in the IT sector is growing very fast. A blockchain network can track orders, payments, accounts, production and much more. And because members share a single view of the truth, you can see all details of a transaction end to end, giving you greater confidence, as well as new efficiencies and opportunities.

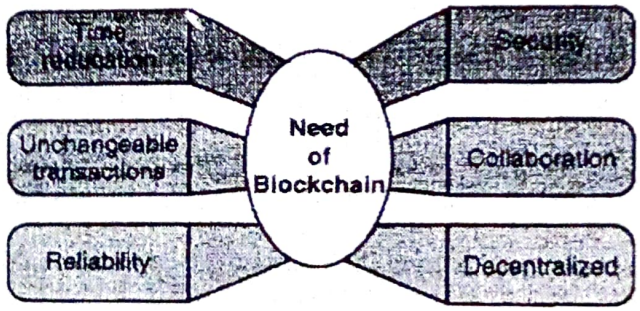

Fig. 2.4.1

2.4.1 Important Properties of Blockchain

Blockchain Applications are adopted because of following properties :

1. **Decentralisation** : Within a distributed network, data presented in the form of a distributed ledger. If the information is somehow corrupted, most participants can reject it, that's why it minimises possibilities of fraud.
2. **Immutability** : Once performed, a transaction is recorded with no further alterations or removal. Time and date stamps ease data tracking in the long run. Therefore, blockchain ensures reliable data audits.

3. **Cybersecurity :** Due to powerful encryption and immediate recording, the probability of attacks performed by malicious intruders falls to its lowest level. Anyways, it's a lot more complicated to hack such a network compared to systems stored on dedicated servers. Access permissions and data anonymization also guarantee high privacy protection.

4. **Cost reduction :** Organisations, an decrease in operational expenses due to eliminating facilitators, an ability to process transactions efficiently, Data can be aggregated automatically as well as reporting and auditing processes is easy.

5. **Traceability :** Tracing the origin of products and managing inventories more efficiently is of high importance for retailers. Thus, enabling a high quality of goods in real time.

By bringing disruption and business transformation, blockchain technology helps organisations minimise security-related risks and decrease operational costs.

2.4.2 Blockchain for Industries

- Industry leaders are using IBM Blockchain to remove friction, build trust and unlock new value. Several industries like Unilever, Walmart, Visa, etc. use blockchain technology and have gained benefits in transparency, security, and traceability.
  - Supply chain
  - Healthcare
  - Government
  - Retail
  - Media and advertising
  - Oil and gas
  - Telecommunications
  - Manufacturing
  - Insurance
  - Financial services
  - Travel and transportation
- As a database, a blockchain preserves information electronically in digital format. Blockchains are renowned for their crucial role in cryptocurrency systems, such as Bitcoin, for maintaining a decentralised and secure record of transactions. The innovation with a blockchain is that it guarantees the security and fidelity of a record of data and builds trust without the need for a trusted third party. Virtually anything of value can be tracked and traded on a blockchain network, cutting costs and reducing risk involved. It is estimated that Blockchain technology has been adopted by more than one-third of the companies in the world and demand for blockchain developers is ever-increasing.

2.4.3 Blockchain Limitations

- When deciding whether to implement it or not, it's important to know what challenges lie behind its introduction and the technology itself.

1. **Low Scalability :** The problem is that transaction speed depends largely on network congestion, which means that the more people or nodes are involved, the slower the pace is. This problem is related to scalability issues with blockchain networks. In simple words, the more people or nodes join the network, the chances of slowing down is more!

   Here's an example : Centralized payment systems can process tens of thousands of transactions per second, while Bitcoin can only manage seven.

2. **Lack of Awareness :** People do not know the true value of blockchain and how they could implement it in different situations.

3. **Limited availability of technical talent :** According to estimates, each year, the need for high-skilled blockchain developers is more. But in blockchain technology, there are not so many developers available who have specialised expertise in blockchain technology. Hence, the lack of developers a problem of developing anything on the blockchain

4. **Implementation Challenge :** It's all about initial financial investments. For some businesses, implementation costs may turn out to be overwhelming.

5. **Immutable :** In immutable, we cannot make any modifications to any of the records. It is very helpful if you want to keep the integrity of a record and make sure that nobody ever tampers with it. But immutability also has a drawback.

6. **Private Key Issues :** In the decentralised environment, private keys owned by individuals may become a weak spot. When you are dealing with a private key, then you are also running the risk that somebody may lose access to your private key. It happens a lot in the early days when bitcoin wasn't worth that much. If stolen, it puts both sensitive data and finances in problem. If lost, then wallet access is gone forever.

7. **Problematic Integration With Legacy Systems :** If the blockchain solution is to be integrated with outdated systems already in use, possible data loss or corruption risks arise,

8. **High Energy Consumption :** Most blockchain-based solutions, like Bitcoin, use a proof-of-work consensus algorithm for validating transactions, which utilizes excessive computing power comparable to the yearly electricity consumption of a country like Denmark. With the resources needed to cool down the equipment, prices are only rising. So, if proof-of-work is your only option, you'll have to pay for it with energy costs.

2.5 Blockchain Adoption

"Web 3.0" technology that has gained popularity. It relies on decentralised technology. Blockchains are ecosystems that demand broad adoption to work effectively.

- **Blockchain So far :** Initially, it was about Bitcoin; but afterwards many other cryptocurrencies also came into the market following the trend. While some of them found their fate, some other cryptocurrencies lagged behind. However, soon the blockchain technology found its real potential and extended to many other unpredictable domains. financial domains like Banking, Insurance, Healthcare Industry, Enterprise software development, and so on; today the blockchain is excessively changing existing technology frameworks of almost all domains. The blockchain market is expected to grow $20 billion by 2024, according to prominent statistics websites.

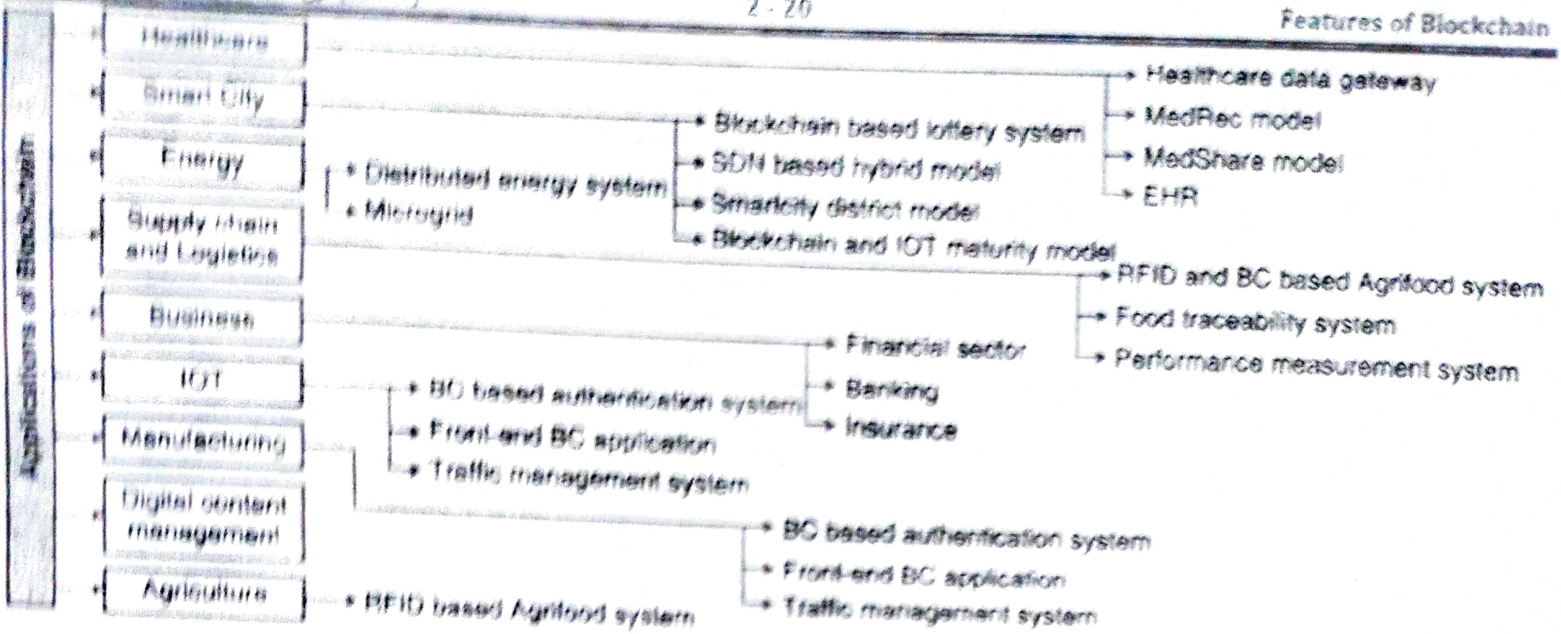

Fig. 2.5.1

- **Banking and payments** : All the payment and banking systems are now moving towards blockchain. Bitcoin-like cryptocurrencies can manage the payment systems without any geopolitical restrictions. An example of bitcoin-based remittance is ABRA. JPMorgan Chase has entered the blockchain space with the JPM Coin, which is designed to facilitate real-time cross-border payments between its business clients. Other banks like Goldman Sachs and Citigroup have also experimented with blockchain. The incumbents performed an equity swap built on Axoni's Axcore blockchain in February 2020.

- **Stock market** : For years, companies have worked to ease the buying, selling, and trading of stocks, and now new blockchain focused startups are looking to automate and secure the process more efficiently than any past solution.

- **Crowdfunding** : It is a popular method of fundraising, for new projects and startups. In block-chain based crowdfunding platforms trust is built through online reputation systems and smart contracts, which eradicates the need for a central party who charges high fees for this service. Tokens of their own can be released by new projects that later can be exchanged for services, products or cash.For example, the movie BRAID became the first major feature film to be financed through a token "crowdsale" on the Ethereum blockchain through its $1.4M campaign on Wefund.

- **Crypto exchanges** : One way blockchain reduces conventional cybersecurity risk is by simply removing the need for human intermediaries thus lessening the threat of hacking, corruption, or human error.

- **Insurance** : The global insurance market is centered on trust management. Blockchain is the new way of handling trust. Blockchain guarantees trust by mutual distrust between participants. Most blockchain applications in the insurance industry today are focused on improving operational efficiency. An example of a blockchain based insurance management system is 'Aeternity'. Insurwave, a joint project between consulting firm EY and blockchain company Guardtime, delivers a blockchain platform aimed at marine insurance.

- **Healthcare** : The use of blockchain technology could allow hospitals, payers, and other parties in the healthcare value chain to share access to their networks without compromising data security and integrity. Health Verity is one of the platforms, combining a health data exchange with a blockchain product to manage permissions and access rights.

Cyber Security : In blockchain, data is secured and verified using cryptography. This will restrict all unauthorized hacks and modifications in the system. It eliminates the middlemen from the system so that no one can make any unauthorized changes.

Supply chain : The blockchain can revolutionize the supply chain by providing better accountability, transparency, and feedback mechanism along the supply chain. Using the blockchain supply chain management, any product can be tracked completely and easily. Each and every movement, along with the condition of a product, can be recorded in the blockchain with IoT sensors. Provenance and Block verify is a blockchain based supply chain management system.

Online Data Storage : Data on the centralized server like Google Drive, One drive etc. are vulnerable to the single point of failure. Blockchain permits distributed data storage in a more robust and secure way. Storj is one such encrypted cloud storage facility.

Networking and IoT : The blockchain technology can be applied in Networking and IoT to build a decentralized network of IoT devices. This eradicates the requirement for a central location to manage the IoT devices.

Government : Applying blockchain technology in government systems will reduce red-tapism, bureaucratic hurdles, and increase transparency and efficiency of government operations. The government of Dubai has already started to implement this technology. Some governments are taking it upon themselves to realize the benefits of blockchain. For example, the Swedish Land Registry has explored the use of blockchain for land registration, potentially eliminating fraud and reducing ownership disputes. It has also tested the use of smart contracts to execute property sales.

Vaccine distribution & monitoring : Governments in Malaysia and Singapore are already employing blockchain to authenticate vaccine certificates, using systems that can trace the exact vaccine batch of the vial used for an individual.

Multimedia and entertainment : Now blockchain has stepped into the entertainment field where the third party interference is to a higher extent. Implementation of blockchain technology in this fields will remove the middleman from the scenario. One of the entertainment areas where blockchain has already started their implementation is Online music E.g : Ujo and Mycelia music.

- **Real estate :** It is another important area where blockchain implementation will create a drastic change. The current real estate system is facing a lot of transfer and ownership issues. Implementation of blockchain in this field can control the entire real estate systems with shared ledgers.
- **E-commerce :** Blockchain technology has the potential to transform e-commerce by lowering transaction costs and tightening transaction security. E-commerce giants such as Walmart, Amazon, and Alibaba have begun exploring blockchain technology.

Adaptation of Blockchain in Indian Government :

- The Indian government is looking forward to establishing a national blockchain framework that will help in transforming the future of as many as forty-four sectors including education, pharma, farming, energy, e-governance, and the likes. The possibility of incorporating and adopting the blockchain technology in India is vast and the government is all set to ease the way for faster and smoother adoption of this new technology.

- For details follow https://blockchain.gov.in/

1. Central Board of Secondary Education Markscard.
2. Karnataka Secondary Education Examination Board Markscard.
3. Karnataka Board Pre University Exam Markscard.
4. Karnataka Birth and Death Certificate.
5. EAASTHI - Urban Property.
6. Supply Chain Management System for medicines (Aushada) of Karnataka.

2.5.1 Generation of Blockchain

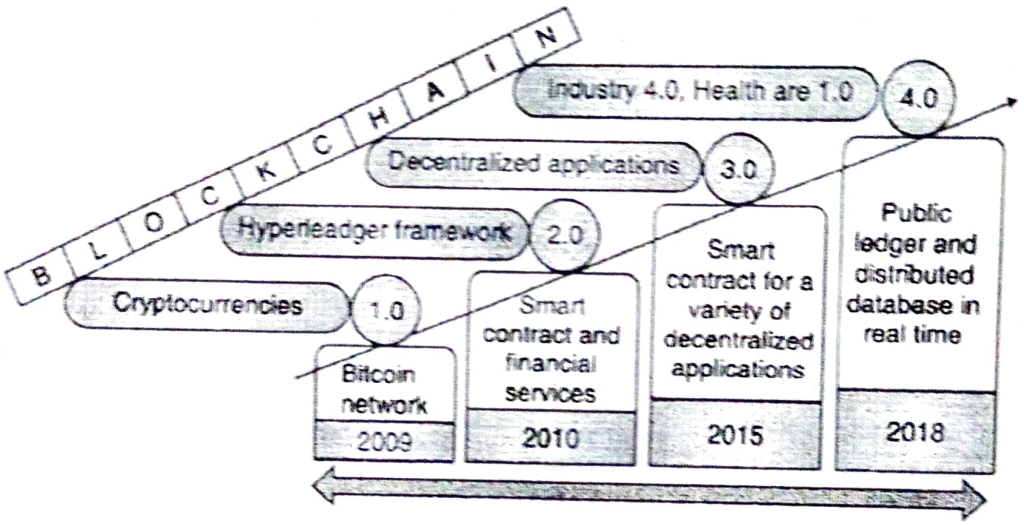

Fig. 2.5.2

1) **BLOCKCHAIN 1.0 :** The first generation of the technology began with the bitcoin network in 2009, which is known as blockchain 1.0. The creation of the first cryptocurrencies was introduced in this generation. The plan was all about payment and its functionalities to generate cryptocurrency.

2) **BLOCKCHAIN 2.0 :** Smart contract and nancial services for various applications were introduced in the second level of the blockchain technology, in 2010. In this generation, the development of blockchain with Etheruem and hyperledger frameworks was proposed.

3) **BLOCKCHAIN 3.0 :** The convergence towards the decentralized applications was introduced in this generation of blockchains. Various research areas such as governance, health, IoT, business, supply-chain, and smart city were considered for building decentralized applications. In this level, hyperledger, etheruem, and other platforms were used, having the ability to code smart contracts for various decentralized applications.

4) **BLOCKCHAIN 4.0 :** This generation mainly concentrated on services such as public ledger and distributed databases in real-time. This level has ideal integration of Industry 4.0-based applications. It uses the smart contract which removes the need for paper-based contracts and regulates within the network by its consensus.

Review Questions

Q.1 Define Blockchain. What are the important features of it ?

Q.2 Explore the anthology of rai tones with distributed ledger.

Q.3 Explain the birth of Blockchain along with the history of block chains.

Q.4 Explain the evolution of Blockchain with timeline.

Q.5 What are the different phases of Blockchain evolution ?

Q.6 Discuss about major Blockchain platforms evolved during phase 3 of Blockchain.

Q.7 Compare centralised and decentralised system.

Q.8 Explain the concept of centralised system with its pros and cons.

Q.9 Explain the concept of decentralised system with its pros and cons.

Q.10 Explain the concept of distributed system with its pros and cons.

Q.11 Compare between centralised versus decentralised versus distributed system.

Q.12 Describe the Blockchain mechanism in brief.

Q.13 Draw and explain the structure of block.

Q.14 State and explain the constitute of a block header.

Q.15 Enlist the methods of identifying the block uniquely ?

Q.16 Write down the importance of genesis block ?

Q.17 List the key features of blockchain ?

Q.18 What do you mean by blocks in the blockchain technology ?

Q.19 List elements are in Every block of the Blockchain ?

Q.20 Elaborate the changing process of blocks.

Q.21 If someone tries to hack block number 5 in a chain of 15 blocks ,what will happen and why ?

Q.22 Explain the layers of Blockchain.

Q.23 State and explain different terms related to Blockchain.

Q.24 Explain Semantic Layer of blockchain technology.

Q.25 Explain Propagation Layer of blockchain technology.

Q.26 Explain Consensus Layer Layer of blockchain technology.

Q.27 Elaborate the importance of Blockchain with respective to safety.

Q.28 How decentralisation is achieved through Blockchain technology ?

Q.29 How can digital freedom be achieved through Blockchain ?