Project Report on

Distributed ledger system analysis with crypto-currency and other useful applications and use cases

Submitted by

Hammad Hanif Mansoor

In partial fulfillment for the award of the degree

Master of Science in Internetworking

(From University of Alberta)

Under the guidance of

Juned Noonari

September 2018 – March 2019

ACKNOWLEDGMENT

I have taken efforts in this project. However, it would not have been possible without God's mercy and the kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

A special thanks to my supervisor Mr. Juned Noonari for his untiring efforts in guiding and helping me understand and overcome every problem that would have otherwise stopped me mid-way. I thank him for being the motivating force and helping me understand the applications and deliverables of this project.

I am very grateful to Mr. Shahnawaz Mir for guiding me for my project, His co-operation in providing all essential information regarding the project and also for his support in completing the project.

I would also like to express my gratitude to all my friends who provide advice and guidance. Last but not least, to my parents for continuous support and encouragement, whenever I needed. Without their unconditional love and guidance, this couldn't be possible.

My thanks and appreciations also go to university in developing the project and people who have willingly helped me out with their abilities.

ABSTRACT

The main purpose of this research is to explore the potential of Distributed Ledger System analysis (DLT) with respect to cryptocurrencies and other useful application. Distributed ledger technology can play a crucial role in the future of our financial system and other useful sectors. In 2009 cryptocurrency Bitcoin introduced DLT and from past few years it's been targeted by numerous financial and government sectors. With the help of computer science, concepts of encryption such as public-key cryptography and algorithms DLTs made it possible to keep the trusted third party away and to prevent shared databases restricted between the participators. This research on distributed ledger technology (DLT) and blockchain is part of a sequence that will outlines the history that led to the present-day system, mechanisms, and key features of DLT, the difference between private and public DLT, the technology's main advantages and challenges or risk, relevant examples of DLT and discover the substitute concepts and an summary of activities by international organization, government and other participants. Finally, this research proposes the next phases to study and evaluate areas where DLT could possibly be integrated.

Table of Contents

Table of figure

Chapter 1 - Introduction

1.1 Introduction

Ledger is the basic infrastructure of the accounting system. The system records who have or has what, who own what to whom. The idea remains the same; the system uses to record the information. For as long as societies have been engaging in trade or other exchange between them, means of keeping records and accounting tools have differed from time to time concerning the evolving technology. From clay tablets to shells to PAPYRUS, from papers books to digital form such as computers. The objective was always to keep the records safe and effective. Ledgers are the basic foundation.

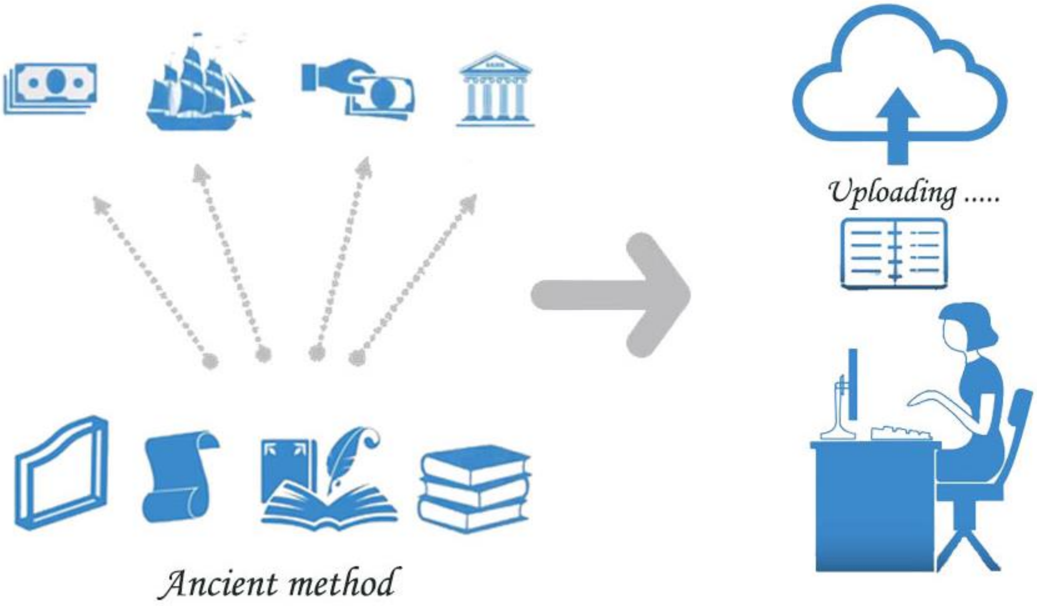


Figure 1- Ancient method Vs. Nowadays

From centuries ledger are maintained by humans, and by the time method and tools have changed, but one thing that never changed in keeping those ledgers was the involvement of the third party. A mediator always had to keep track of the transactions and register and uphold those accounts. The reason behind these third parties was to provide a simple validation platform and also to enable people to trust one another.

The reason behind the massive network of ledger systems was the evolution of the worldwide trade, but the consequences of this system can be terrible and unexpected as the system is defenseless and can be misinterpreted which might take back you to 2008 financial crisis.

Distributed Ledger Technology (DLT) is the major method of the ledger to exclude the involvement of the third party. It allows the ledger to be distributed between all of those who are using it and give them the responsibility to authenticate and preserve it. As a result of the decentralized structure of data registry where business dealing and transaction are transparent, direct, reliable, immediate and incorruptible.

1.2 Background

The central concept behind the blockchain technology was developed in 1991 for electronic ledger for digitally signing chain of information which could prove that nothing changed in the collection of the signed document. In 1992, a system was designed known as Merkle tree or hash tree by Bayer, Stornetta, and Haber which allows numerous document certificate to be organized and collected into one block with better efficiency.

Blockchain was first conceptualized by an individual or gathering of individuals known as Satoshi Nakamoto in 2008. Nakamoto enhanced the structure in a vital way utilizing a Hash cash-like strategy to add information of blocks to the chain without expecting them to be signed or marked by a trusted third party. Nakamoto executed the design next year as a central module of the cryptocurrency, where it functions as a public ledger for all the operation in the network.

The so-called author Satoshi Nakamoto and bitcoin’s first owner remain a mystery. Since then DLT/Blockchain Technology become strongly linked to bitcoin. Most current digital currency scheme follows the Nakamoto’s paper blueprint with a different method. The first blockchain application or use case was bitcoin.

Several electronic cash systems existed before bitcoin but didn’t achieve that level of popularity. Bitcoin achieved convincing capabilities by adopting and implementing blockchain technology. The structure of the blockchain allowed bitcoin to be fulfilled in a distributed manner so that no individual user controlled the currency and no failure existed. Its primary goal was to enable a direct financial electronic transaction between users without the need of the third party. A self-policing mechanism was used by distributed blockchain that confirms valid transactions information were linked to the blockchain. Finally, the distributed ledger of the blockchain builds the architecture with complete time-stamped, transparency and decentralized which gains and promote the trust for the users.

Blockchain and bitcoin connection is similar to the internet and email — a massive system through which you can build several applications. Digital currency is just a part.

1.3 Motivation

Indeed, there are many who trust Bitcoin even today, and blockchain is one and the equivalent, though certainly not. Individuals who started recognizing around 2014 that blockchain could be used for more than cryptocurrency began putting resources in and investigating how blockchain could change a wide range of operation scheme. Blockchain core is an open, decentralized ledger that records permanent method exchanges between two gatherings without requiring confirmation from outsiders. This makes the process productive to a great extent and one individual forecast will drastically reduce the exchange costs.

At the point when business visionaries understood the intensity of blockchain, there was a flood of speculation and revelation to perceive how blockchain could affect supply chains, human services, transportation, insurance, the voting system and that's only the tip of the iceberg. About 15% of money related organizations are as of now utilizing technology of blockchain.

1.4 Technology

Distributed Ledger Technology (DLT) is still at a beginning period of improvement and development. The advancement of blockchain innovation is nevertheless the first, however vital advance towards a disruptive revolution in the ledger technology that could change the conduct of private and public sectors. The innovation can be received so that 'real' changes to ledgers can be made on a fundamental level by anybody (an 'unpermissioned' ledger), or by a predetermined number of people or even a solitary approved individual (in a 'permissioned' ledger). For government applications, 'permissioned' ledgers are probably going to be more engaging than Bitcoin's unpermissioned architecture because they permit the proprietor, or proprietors, of the data to implement runs on who is and isn't permitted to utilize the framework or system. Distributed ledgers have the additional favorable position of moving a great deal of the multifaceted nature of overseeing security out of the spotlight, making system less demanding and less expensive to utilize.

There are numerous unsolved issues to handle before the maximum capacity of this, and related innovations can be acknowledged, including the goals of issues of security, protection, scalability, execution, and flexibility. There is likewise an uncommon array of opportunity to create algorithms that will add modernity to ledgers by supporting smart signature, contract, and different applications. These will improve and broaden the value and scope of services of ledgers. This field is growing quickly, and a large number of these issues are now being researched furthermore, in some case they are resolved. If the government sits tight for perfect arrangements or solution, it will miss the chance to shape and acquire executions of the innovation that will give a most extreme advantage to general society segment.

Chapter 2 - Understanding DLT and Blockchain

2.1 What is distributed ledger technology and blockchain?

Distributed ledger technology (DLT) refers to an imaginative and quick developing technique to recording and sharing information over numerous data stores or ledgers. This innovation takes into consideration exchanges and information to be recorded, shared and synchronized over a distributed system of various network members or participants.

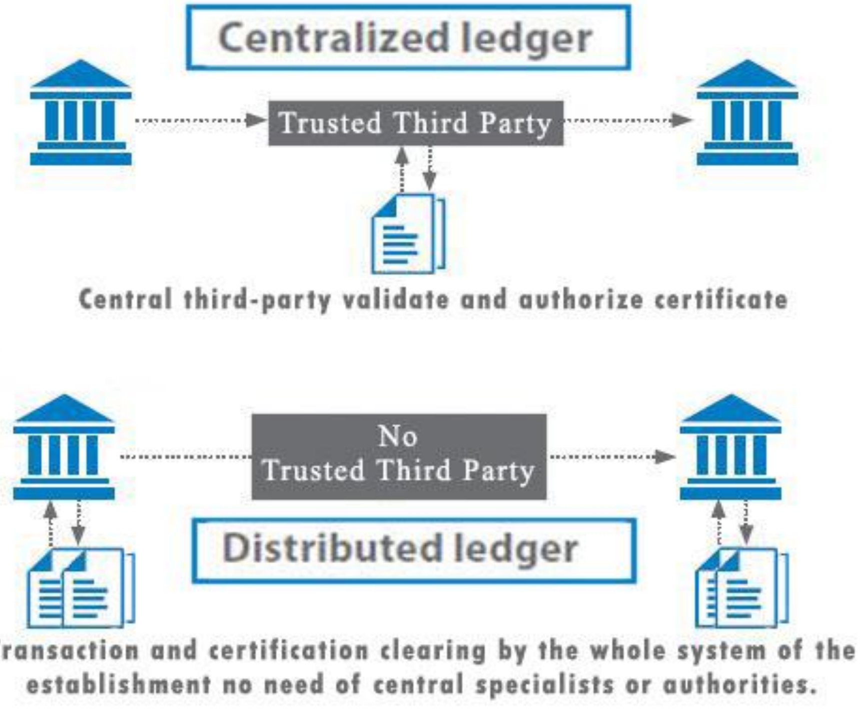

Centralized ledger
Trusted Third Party
Central third-party validate and authorize certificate
No
Trusted Third Party
Distributed ledger
Transaction and certification clearing by the whole system of the establishment no need of central specialists or authorities.

Figure 2 - Centralized Vs. Distributed Ledger

The blockchain is a specific sort of information structure utilized in some distributed ledger which saves and transmits information in bundles called 'blocks' that are associated with one another in a digital 'chain.' Blockchain utilize cryptographic and algorithmic techniques to record and synchronize information over a system in a changeless way.

For instance, a modern digital currency exchange would be recorded and transmitted to a network in a data or information block, which is first approved by system individuals and after that connected to an existing chain of blocks in order, in this way creating a blockchain. As the straight chain develops when new blocks are included the previous block can't reflectively be changed by any system member.

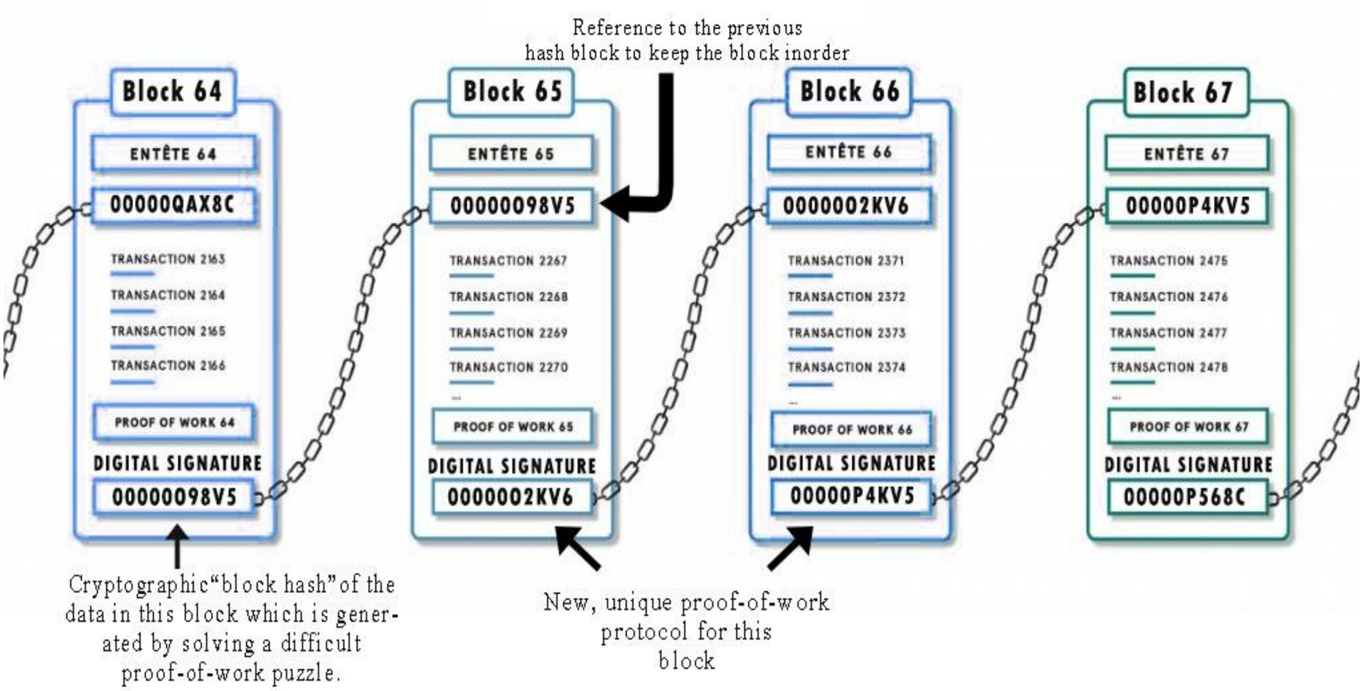


Figure 3- Genesis Blockchain Structure

The blockchain was a certainly quick development by an individual or gathering of individuals known as a pseudonym, Satoshi Nakamoto. The first ***block#0*** made in 2009 is alluded to as Genesis block in Bitcoin Blockchain. It is the basic ancestral parent of all the new block made and if transverse in reverse in time we will achieve genesis block at last.

2.2 Distributed ledger technology and Blockchain Relation to Digital Currencies

DLT has been firmly connected to advance digital currencies since its origin as noted before it was developed as the fundamental innovation of the digital money called Bitcoin. The creator of Bitcoin, composing under the assumed name Nakamoto, depicted the innovation in a 2008 white paper permitting any two agreeable participants to execute straightforwardly with one another without the requirement for a trusted third party. Nakamoto has not been identified until this day, wiped out his entire online existence in 2011.

Blockchain innovation for Bitcoin was intended to tackle for the issue of "double-spending", which restrained a full advancement of cash into the electronic world, comparable to the digital or electronic changes in message, music, records, and documentation. Before Bitcoin, to dodge or avoid double-spending, a trusted central third party was expected to approve the transaction to ensure responsibility and ownership of account and balance. DLT's basic development in the setting of digital currencies standards is that it offers a cryptographic solution for giving security and ensuring framework trustworthiness in a decentralized ledger that is kept up by a system of mysterious members with no need for trust one or more organizations or institution.

The Bitcoin blockchain was planned with the explicit goal of making a digital currency that is free from government control and hides the identities of its network members. Bitcoin was an ideological task from the start, profoundly inserted in the anti-censorship belief system of the online network from which it came, identified as "cypherpunks"[1], who embrace an extreme strand of techno-libertarianism. While Bitcoin was the first utilization of DLT and the first to accomplish scale, the innovation has countless applications a long way past advanced monetary forms.

The secrecy offered for executing quickly on the web pulled in the attention of offenders and Bitcoin has been utilized for financing illegal exercises. Nonetheless, even though the personalities of transacting accomplices can be mysterious, all Bitcoin trade exchanges are logged in a distributed ledger that is observable to the overall public, furthermore, it is conceivable to relate Bitcoin trade with explicit mysterious entities. (This is the reason the term 'pseudonymous' is frequently utilized with regards to Bitcoin.). The namelessness given by Bitcoin can be contrasted with the namelessness given by an email address. All Bitcoin exchanges contain a wallet address of the sender and the recipient, which can be thought of as pseudonymous, to email addresses. While the addresses connected to the trade are known, the proprietors behind the addresses can remain unknown, like making an impression on an email address. Law enforcement authorities were effective in relating real-life identities to the unknown entity in the Bitcoin network on account of the captures identified with Silk Road, an online underground or black market for illegal exercises, including moving of illicit medications or drugs[2].

A couple of features of the Bitcoin blockchain have harmed the cryptocurrency reputation and cause stresses for governments. This consolidates the nonattendance of control of colossal quantities of the bitcoin exchanges and the climb of ransomware PC malware that demands a ransom paid in bitcoin give obscurity. Another issue of concern is bitcoin's data catastrophe issue: if you lose your private key to your wallet, you lose all your money. Conventional, centralized banking is substantially more reliable to this. These altogether include explicit to applications what's more, enterprises encompassing bitcoin, as opposed to highlights of DLT framework. To date, there have not been any genuine honesty issues emerging from the core bitcoin blockchain itself. Regardless of its anti-authority backgrounds, DLT can likewise be utilized to make digital fiat currencies issued by national or government banks.

2.3 How does distributed ledger technology works

DLT goes ahead the impact points of a few shared peers to peer (P2P) advancements empowered by the internet, for example sharing music, email or other media documents, and web communication. In any case, web-based exchanges of asset proprietorship have long been indefinable, as this requires guaranteeing that its actual proprietor just exchanges an asset and ensuring that the asset can't be transferred more than once, i.e., no double spend. The advantage being referred to could be anything of significant worth.

A breakthrough paper was composed by an unidentified individual utilizing the pseudonym Satoshi Nakamoto, (*Bitcoin: A Peer-to-Peer Electronic Cash System*) in 2008, recommended a novel methodology of exchanging "funds" as "Bitcoin" in a peer-to-peer (P2P) way. The hidden innovation for Bitcoin sketched out in Nakamoto's paper was named Blockchain, which alludes to a specific method for sorting out and putting away data and transactions. Hence, different methods for arranging data and transaction for resource moves in a P2P way were conceived – prompting the expression "Distributed Ledger Technology" (DLT) to allude to the more extensive classification of advances.

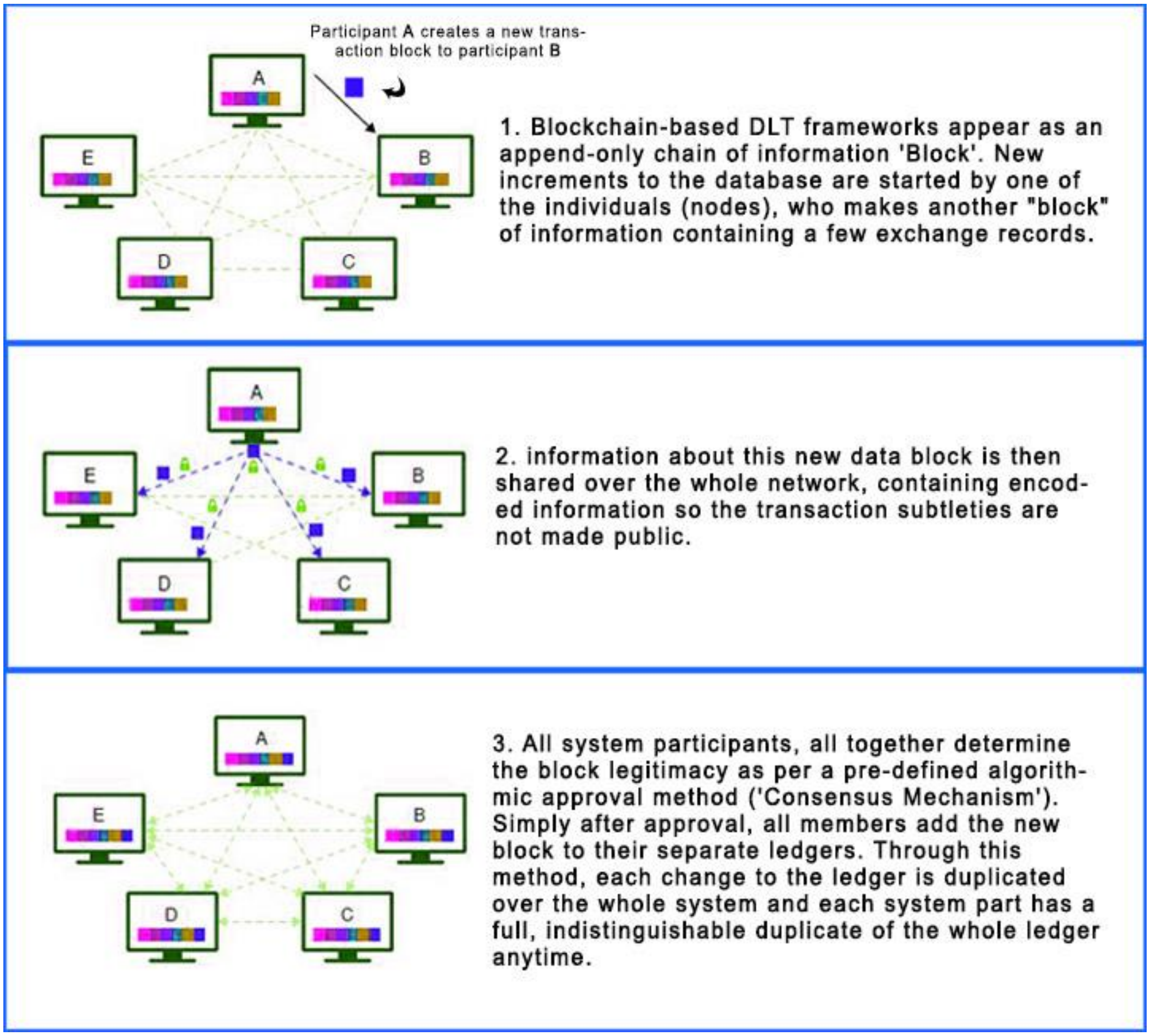
DLT alludes to a novel and a quick developing way to deal with the account and sharing information over various information stores (ledgers), which each has precisely the same information records and are, all things considered, kept up and controlled by a distributed network of computer servers, which are called nodes. One approach to consider DLT is that it is just a dispersed database with certain explicit properties. Blockchain, a specific kind of DLT utilizes cryptographic and algorithmic strategies to make and check a consistently growing, an affix just information structure that appears as a chain of purported 'transaction blocks'– the blockchain – which serves as the ledger.

New augmentations to the database are started by one of the individuals (nodes), who makes another "block" of information, for instance, containing a few exchange or transaction records. Data about this new information block is then shared over the whole system, containing encoded data, so transaction subtleties are not made public, and all system members collectively decide the block legitimacy as indicated by a pre-characterized algorithmic approval strategy *consensus mechanism*. Only after approval, all members add the new block to their separate ledger. Through this component, each change to the record is imitated over the whole system and each system; the part has a full, indistinguishable duplicate of the entire ledger anytime. This methodology can be utilized to record exchanges on any benefit which can be spoken to in an advanced shape. The transaction could be a change in the trait of the interest or exchange of proprietorship. (See figure 4).

Two center qualities of a DLT-based framework are:

(I) Capacity to record, store and trade "information" in digital frame across over various, self-intrigued counterparties without the requirement for a central record-guardian (i.e., peer-to-peer) and without the need for trust among counterparties.

(II) Guarantee there is no 'double spend" (i.e., the equivalent resource or token can't be sent to different parties)


Participant A creates a new transaction block to participant B
A
B
C
D
E
A
B
C
D
E
A
B
C
D
E

1. Blockchain-based DLT frameworks appear as an append-only chain of information 'Block'. New increments to the database are started by one of the individuals (nodes), who makes another "block" of information containing a few exchange records.

2. information about this new data block is then shared over the whole network, containing encoded information so the transaction subtleties are not made public.

3. All system participants, all together determine the block legitimacy as per a pre-defined algorithmic approval method ('Consensus Mechanism'). Simply after approval, all members add the new block to their separate ledgers. Through this method, each change to the ledger is duplicated over the whole system and each system part has a full, indistinguishable duplicate of the whole ledger anytime.

Figure 4- Working of Blockchain based DLT [3]

2.4 Nature of distributed ledgers

Record keeping has dependably been a centralized process that involves reliance on the record attendant. The most critical and essential advancement of DLT is that power over the ledger does not lie with anyone substance but instead is with a few or all system members – depending on the sort of DL. This separates it from other technology progresses, for example, information replication or cloud computing, which are usually utilized in existing shared ledgers. True, this implies in a DL, no single substance in the system can alter past information passages in the records, and no single substance can affirm new increases to the record. Preferably, a pre-defined, decentralized consensus mechanism is utilized to approve new information passages that are added to the blockchain and in this way shape new entries in the ledgers. There exists, anytime, just a single variant of the ledger and each system member possesses a full and up to date duplicate of the whole ledger. Each internal addition to the ledger by a system member is spread to all nodes. After validation is acknowledged, the new transaction is added to all individual ledgers to guarantee information consistency over the whole system or network.

This distributed feature of DLT permits self-fascinated members in the P2P network to mutually record checked information in their particular ledgers, for instance, transaction records, without depending on a confided in central party. The withdrawal of the central party can build speed and possibly dismiss expenses and wasteful aspects related with keeping up the ledger and resulting compromises. Vitally, it can likewise upgrade security because there is never again a single purpose of assault in the whole network. To corrupt or attack the ledger, an attacker needs to pick up control over the larger part of servers in the network; corrupting a single or a few members does not compromise the system integrity.

Nevertheless, security threats in the software application layers based on top the DL can turn into attack surface. Shortcomings in this layer can cause misfortunes to the clients of a DL framework, even when the core technologies stay secure and safe. Striking precedents that caused commercial and reputational harms were the hacks of Bitfinex and Mt. Gox in Japan.

2.5 Types of ledgers

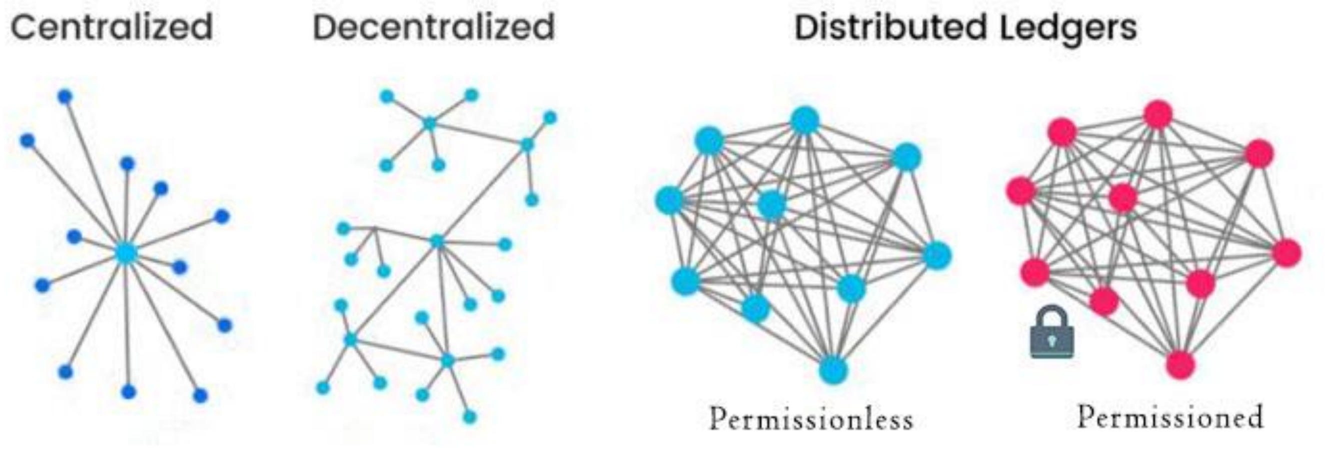


Figure 5- Types of Ledger Systems

Centralized Ledger

All authorities merge their local databases with a centralized electronic ledger that is kept and organized by a trusted central party.

Decentralized Ledger

It is a systematic delegation of the specialist at all ranks of board or management and in the majority of the system network.

Distributed Ledger (Permission-less)

Every node in a P2P network possesses a full and progressive duplicate of the whole ledger. Each proposed neighborhood expansion to the ledger by a arrange participant is communicated over the network to all nodes. Nodes, on the whole, approve the change through an algorithmic accord instrument. After approval is acknowledged, the new option is added to every single separate ledger to guarantee information consistency over the whole network.

Distributed ledger (Permissioned)

In a permissioned framework, nodes require approval from a central section to get to the system and make changes to the ledger. Access controls can contain identity check.

2.6 Open/Permissionless vs. Permissioned Distributed Ledgers

Permissioned and permissionless ledgers

Public or Permissionless ledgers are considered by certain as the 'most flawless' type of Blockchain. A distinctive example of a public or permissionless blockchain is the Bitcoin networks. In this sort of design, the member is 'permissionless' and anybody can participate in the ledger and approve exchanges or transaction with a completely regressed expert. Members are distinguished through pseudonyms or are kept unknown, and exchanges or transaction are validated or approved by 'miners' through a boost framework system. This type of distributed ledger empowers high security yet, in addition, causes high exchange costs because of the asset serious accord mechanism.

Private or Permissioned ledgers have pulled into consideration from organizations. This kind of ledgers limits transparency by illuminating the personality of members in the network, access is limited to a specific number of members or participants, which are known to one another, and is exposed to an endorsement from different individuals from the network. No 'verification of-work' is expected to approve transactions, not at all like on account of the permissionless ledgers, and in this manner, there is no boost framework. Permissioned ledgers can be distributed for closed networks that share

Distributed ledgers frameworks can be open/permissionless or permissioned, and there are key contrasts between them. Ethereum and Bitcoin are the most conspicuous instances of totally permissionless blockchain, where network members can join or exit the system freely, without being pre-endorsed or checked by any element. All that is expected to join the system and add exchanges to the transaction with a machine (PC) with the significant programming or software.

There is no focal proprietor, and indistinguishable duplicates of the ledgers are distributed to all network members.

In permissioned Distributed Ledgers individuals are pre-chosen by somebody – a proprietor or an executive of the ledger, who organize network access and sets the principles of the ledger. This comprehends for various concerns governments and controllers have about permissionless distributed records, for example, identity confirmation of network individuals, whom to permit and manage, and legal responsibility for the ledger. Be that as it may, it additionally diminishes a preferred central standpoint of permissionless blockchain, the capacity to work without the requirement for any single individual playing a controller role, which essentially requires different members to trust in this individual. Be that as it may, even in permissioned DLs, when all is said in done there is no requirement for an administrator for the execution of exchanges or transactions.

Permissioned DLs, which control network access, commonly don't require a computing power-escalated confirmation of-work to check transactions yet depend on various algorithmic guidelines to build up an agreement among individuals. In permissionless DLs, which do not control network access, there is no necessity of any trust between the members and a complicated evidence of-work is thus used to create an agreement about ledger sections. Conversely, on account of a permissioned DL, the administrator stands the responsibility to guarantee that the members in the DL are trustworthy. In permissioned DLs, any node can offer an expansion of a transaction, which is at that point imitated to different nodes, possibly even with no agreement mechanism.

In all actuality, this is definitely not a binary classification however the level of transparency and decentralization of distributed ledger frameworks falls on a range with completely open, permissionless blockchain, for example, Bitcoin toward one side of the range and permissioned blockchain facilitated by private substances on the other, and the exact highlights shift from stage to stage. DLT courses of action can be characterized as far as diverse measurements access to the system (open/shut) versus jobs inside the system (limited/unlimited) – see scientific categorization in Fig 5. Numerous organizations utilize a crossbreed approach where they give the innovation to permissioned networks to be based on open blockchain foundation and subsequently confine roles in a distributed ledger technology with open access.

	Permissionless/Open Blockchain	Permissioned Blockchain
Central/Third party	No need of administrator or central authority	Include some level of external control or administration
Control	Everyone can join	An only preselected individual can join the network
Trust	No-one has to trust each other in the network	High level of trust required among the member in the network
Transparency	Ledger is transparent and open to everyone in the network	Different level of transparency required in the ledger
Security	Security through large distribution in a huge scale network	Security through access control merge with DLT in less scale network
Processing speed	Slow exchange or transaction limits transaction volume	Faster exchange or transaction allows increased transaction volume
Validation	Anonymous user identity or protected by pseudonyms	Identity authentication typically required by owner/administrator
Consensus	Challenging proof-of-work required as consensus mechanism	Variety of consensus mechanisms required (typically less difficult & less costly)
Asset	Native cryptocurrencies but can implement any asset also through tokens	Any
Ownership	No legal entity controls or owns the ledger	Greater legal clarity over ownership, the administration is usually a legal entity
Example	Ethereum and bitcoin	Hyper ledger Fabric and R3 Corda

Table 1: Permissionless Vs. Permissioned Blockchain [4]

Few industries makes a refinement between private/public (as far as access) and permissioned/ permissionless (as far as roles) distributed ledger. Ripple, for instance, has a permissioned ledger yet the information is approved by all members. Accordingly their framework can be viewed as public, permissioned ledger. A permissioned DLT where the information is approved just by many members would be considered a private, permissioned ledger.[5]

Both open and permissioned DLs will have valuable applications. The innovation is still at a beginning period of advancement and there are diverse future situations: some trust the business will, in the long run, combine to one overall open blockchain (much the same as one overall web). A wide range of private blockchain (similar to numerous diverse private intranets), while others trust that a few open will keep on existing side-by-side. Initially, the web was a web of data, which had the impact of democratizing access to data. A likely future situation of the blockchain could be a web of significant worth, democratizing access and capacity of digital resources. Since Bitcoin's begin in 2009, more than 600 extraordinary public and private distributed ledger systems or network have developed. However, just a bunch have accomplished and developed phase of advancement. Many blockchain applications are based on public blockchain – transcendently Ethereum and Bitcoin.

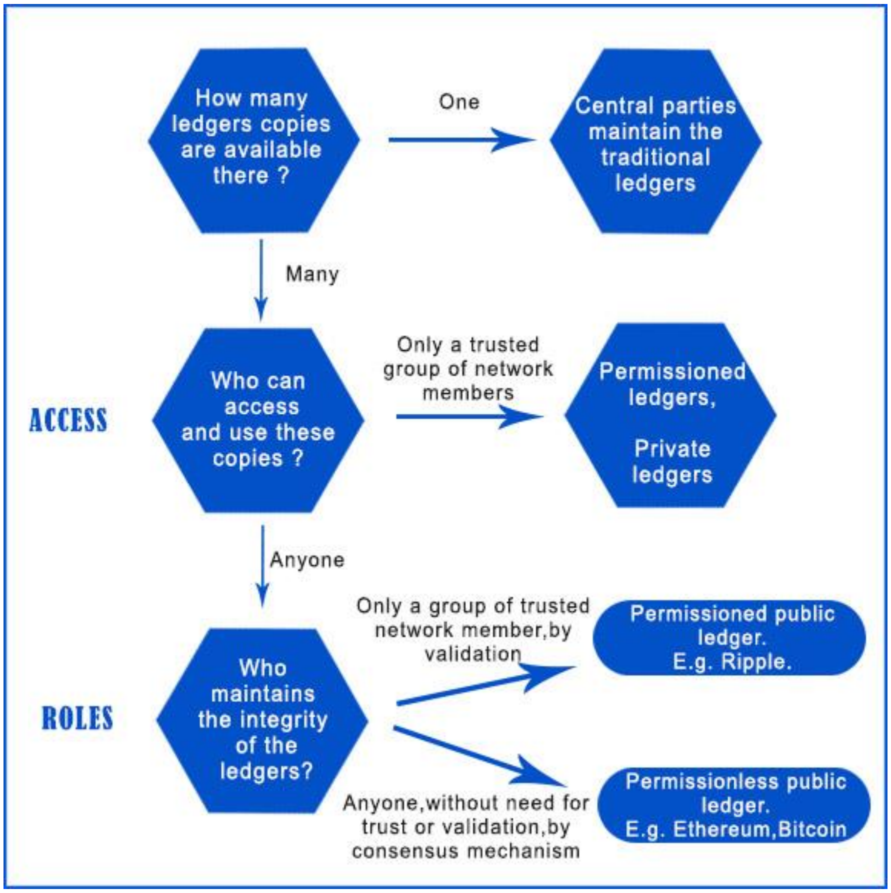

How many ledgers copies are available there ?
One
Central parties maintain the traditional ledgers
Many
ACCESS
Who can access and use these copies ?
Only a trusted group of network members
Permissioned ledgers,
Private ledgers
Anyone
ROLES
Who maintains the integrity of the ledgers?
Only a group of trusted network member,by validation
Permissioned public ledger. E.g. Ripple.
Anyone,without need for trust or validation,by consensus mechanism
Permissionless public ledger. E.g. Ethereum,Bitcoin

Figure 6- Distributed Ledger Taxonomy[6]

2.7 DLT framework

This segment starts to look at the vital and adequate components which include a DLT framework. The point is to give adaptability in the examination and order of DLT frameworks. As appeared in below diagram, a DLT framework can be separated into three reliant center layers:

1. **Protocol layer:** The arrangement of programming characterized rules that decide how the framework works.
2. **Network layer:** Interconnected performing artists and forms that execute the convention.
3. **Data layer:** Information moving through the framework that conveys particular importance in connection to the structure and capacities the framework is planned to play for clients.

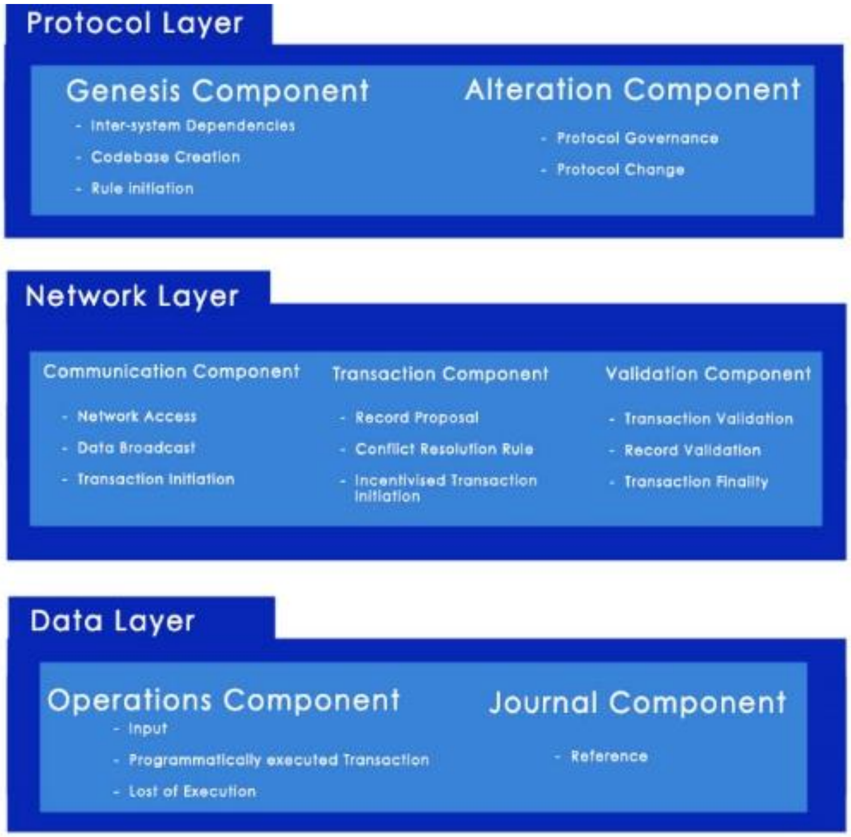
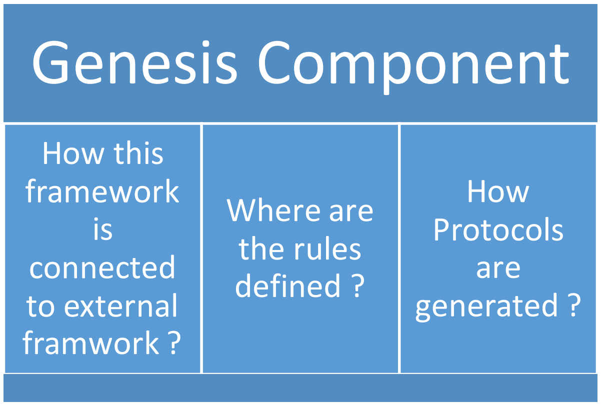


Figure 7-DLT system framework

Each **layer** is made out of at least one more component involved in the creation or process of a DLT framework. A component is an intelligent arrangement of related procedures essential for the working of the framework. A process is a progression of activities completed by performing actors to accomplish a particular goal or arrangement of destinations engaged with the productive activity of a component.

2.7.1 Protocol Layer

The Protocol layer is the establishment of the whole DLT framework. It characterizes the arrangement of formal rules that administer the system and classifies its building structure. The protocol can be viewed as a set of 'constitutional' procedure of action settled upon by all framework members. The protocol consists two parts:

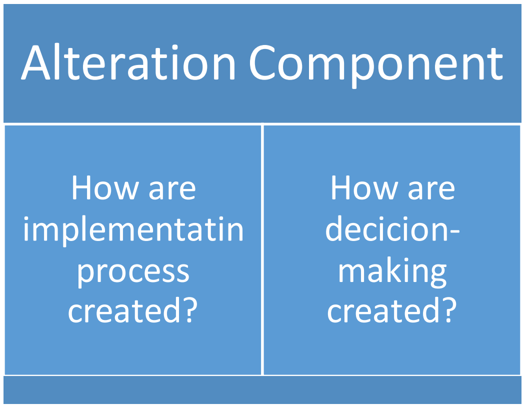


Genesis Component

How this framework is connected to external framwork ?

Where are the rules defined ?

How Protocols are generated ?



Alteration Component

How are implementatin process created?

How are decicion-making created?

Genesis Component:

Symbolizes the procedures of the DLT framework at the time of system launch. It comprises of the introductory codebase and engineering determining the tenets of engagement inside the framework, including the main ('Genesis') record.

Alteration Component:

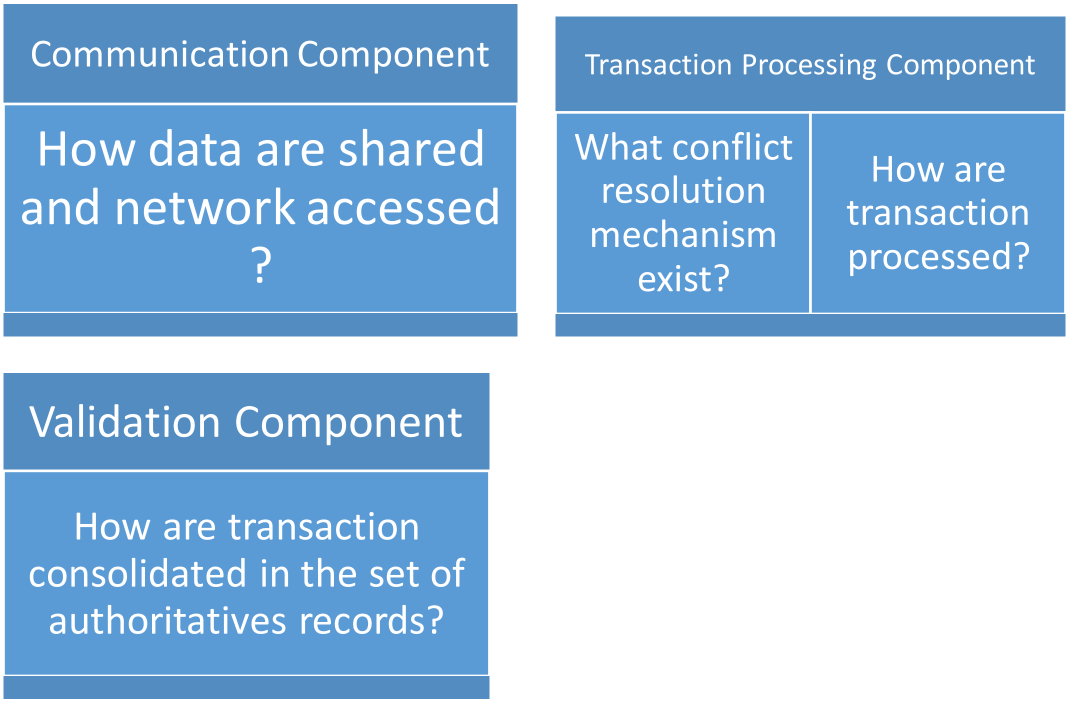
Sets out how the protocol spreads after some time. It incorporates a governance approach (for example how aggregate choices are made) just as an execution thought (for example how the aftereffect of those choices are consolidated). The alteration component require not to be an unequivocal piece of the protocol; without a doubt, generally, DLT frameworks move the administration and related issues 'off-chain'.[7]

Off-chain versus On-chain

The term 'off-chain' alludes to whatever occurs outside of the boundaries limits of a DLT framework. This is the opposite of 'on-chain' which alludes to whatever occurs inside the limits of the DLT framework.

2.7.2 Network Layer

The network layer contains interconnected actors that cooperatively store, process and share data or information. The network layer is the practical execution of the convention rules, portraying how members get to the framework, how information is shared inside the system, how the record is updated and how members confirm the legitimacy of records and transactions. It contains three core parts:



Communication Component

How data are shared and network accessed ?

Transaction Processing Component

What conflict resolution mechanism exist?

How are transaction processed?

Validation Component

How are transaction consolidated in the set of authoritatives records?

Communications Component:

Determines which actor can progress toward becoming members and access the system (open versus closed), how information is shared (private versus public) also, who has the approval to start transaction (restricted versus unrestricted).

Transaction Processing Component:

A set of procedures that determines the mechanism of refreshing and updating the common shared set of authoritative records:

(I) Which members have the authority to update the shared set of legitimate records (permissionless versus permissioned)

(II) How members reach agreement over executing these updates.

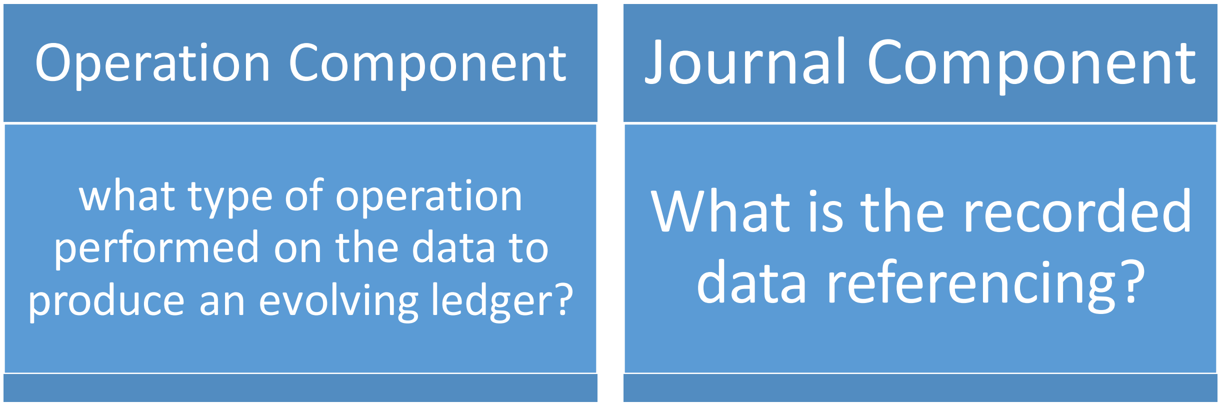
Approval Component:

Sets out the activities attempted by each auditor to check whether transaction and records adjust to protocol rules, for example, are non-conflicting and valid. This is an essential part of a DLT framework that furnishes nodes with the capacity to confirm autonomously what happens inside the framework.

There is a prevalent view that records stored on a DLT framework is *immutable* and can never be turned around. In any case, that isn't really the case: DLT frameworks give unique degrees of transaction finality depending on the framework structure. This implies an affirmed (and executed) transaction may be liable to reversal.

2.7.3 Data layer

The data layer alludes to the information processed and DLT system stores that information in the type of records. A DLT framework exists for the express reason of making a shared data structure, with lot of crucial features, the most vital of which are more often persistence, productiveness, straightforwardness, standardization, what's more, restriction resistance. Inside a set of information states, operation, property rights, and relations well-defined by a DLT framework system, this record gives a legitimate version of records at a minute in time that is both shared among the clients of the framework what's more, refreshed after some time as clients connect to each other through the framework. The information layer comprises of two segments:



Operation Component

what type of operation performed on the data to produce an evolving ledger?

Journal Component

What is the recorded data referencing?

Operation Component:

The processes which run how and which information is utilized in the making of new records, alteration of existing records, and the execution of code. This may likewise incorporate 'smart contracts'.

Journal Component:

Concerns the substance of the kept records i.e. what information inside records are being referenced, or on the other hand 'what is in the blocks?

Censorship Resistance

Censorship Resistance is a term ordinarily utilized with regards to DLT which for the most part refers to the failure of a single party or cartel to singularly perform any of these action:

1. Change rules of the framework
2. Censor or block the transactions
3. Freeze balance or Seize accounts

Reference/Value Linking

The nature of the records, and the value to which they point, are vital parts of the journal component. Inner object may have referenced by records (for example a local token such as bitcoin or ether) or something outer to the framework (for example a physical item followed over a supply chain network).

DLT frameworks can just implement records that reference endogenous (internal) objects

The distinction between exogenous (outside) and endogenous (inner) objects is essential in delineating the limits of a DLT framework, it can only naturally and freely uphold transactions that point to interior assets endogenous to the framework. When the records reference exogenous items, requirement moves toward becoming reliant on external agent.

In such cases, enforcement depends on existing legitimate and financial structures or different arrangement outside of the DLT framework. A few structures (for example Bitcoin) are unequipped for fitting in with the choices of external operators, (for example, courts) without the participation of the members who have control over the particular subset of asset at issue - an idea alluded to as 'power'.

2.8 Merging layers

The proposed calculated system breaks a DLT framework down into three fundamental layers:

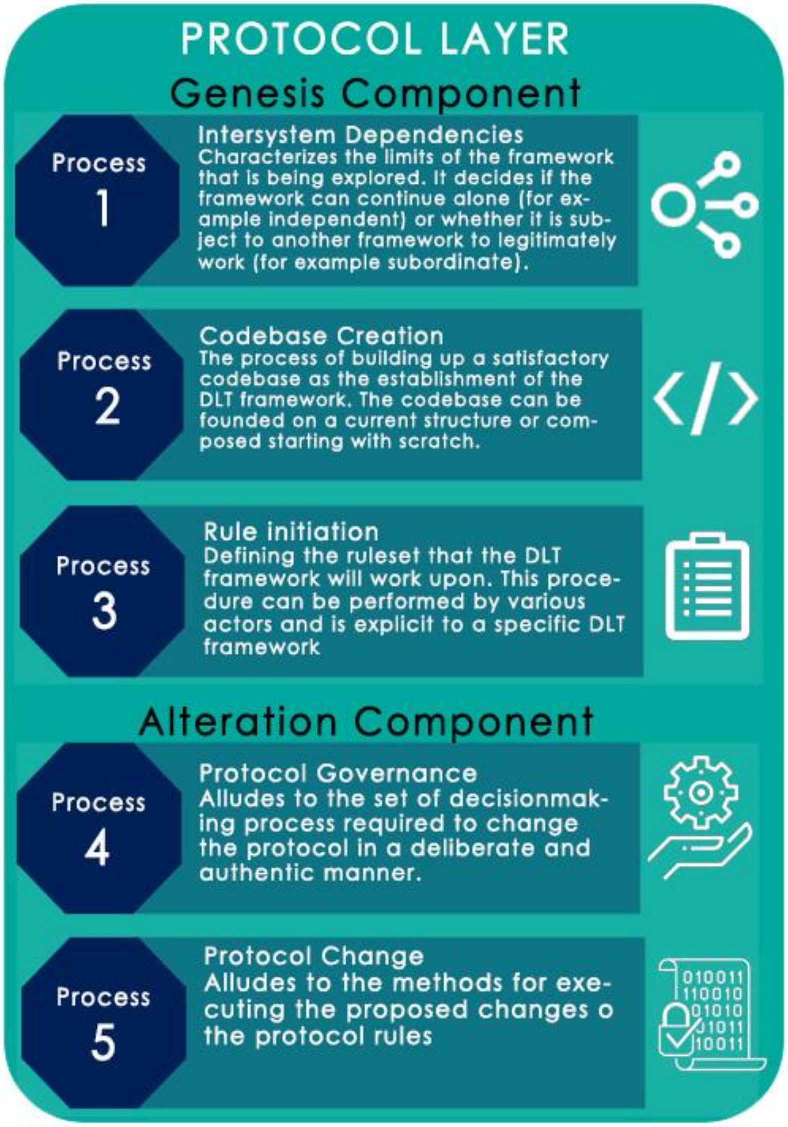
- The protocol layer characterizes, update, and manage the worldwide rule set that administers the system.

- The network layer actualizes the rule set and execute the steps required to achieve framework wide consensus.

- The data layer determines the nature and significance of the data over which agreement is reached

The above figure will summaries the component and its processes relating to each layer of the DLT system.

DLT System Framework Overview



PROTOCOL LAYER

Genesis Component

Process 1

Intersystem Dependencies
Characterizes the limits of the framework that is being explored. It decides if the framework can continue alone (for example independent) or whether it is subject to another framework to legitimately work (for example subordinate).

Process 2

Codebase Creation
The process of building up a satisfactory codebase as the establishment of the DLT framework. The codebase can be founded on a current structure or composed starting with scratch.

Process 3

Rule initiation
Defining the ruleset that the DLT framework will work upon. This procedure can be performed by various actors and is explicit to a specific DLT framework

Alteration Component

Process 4

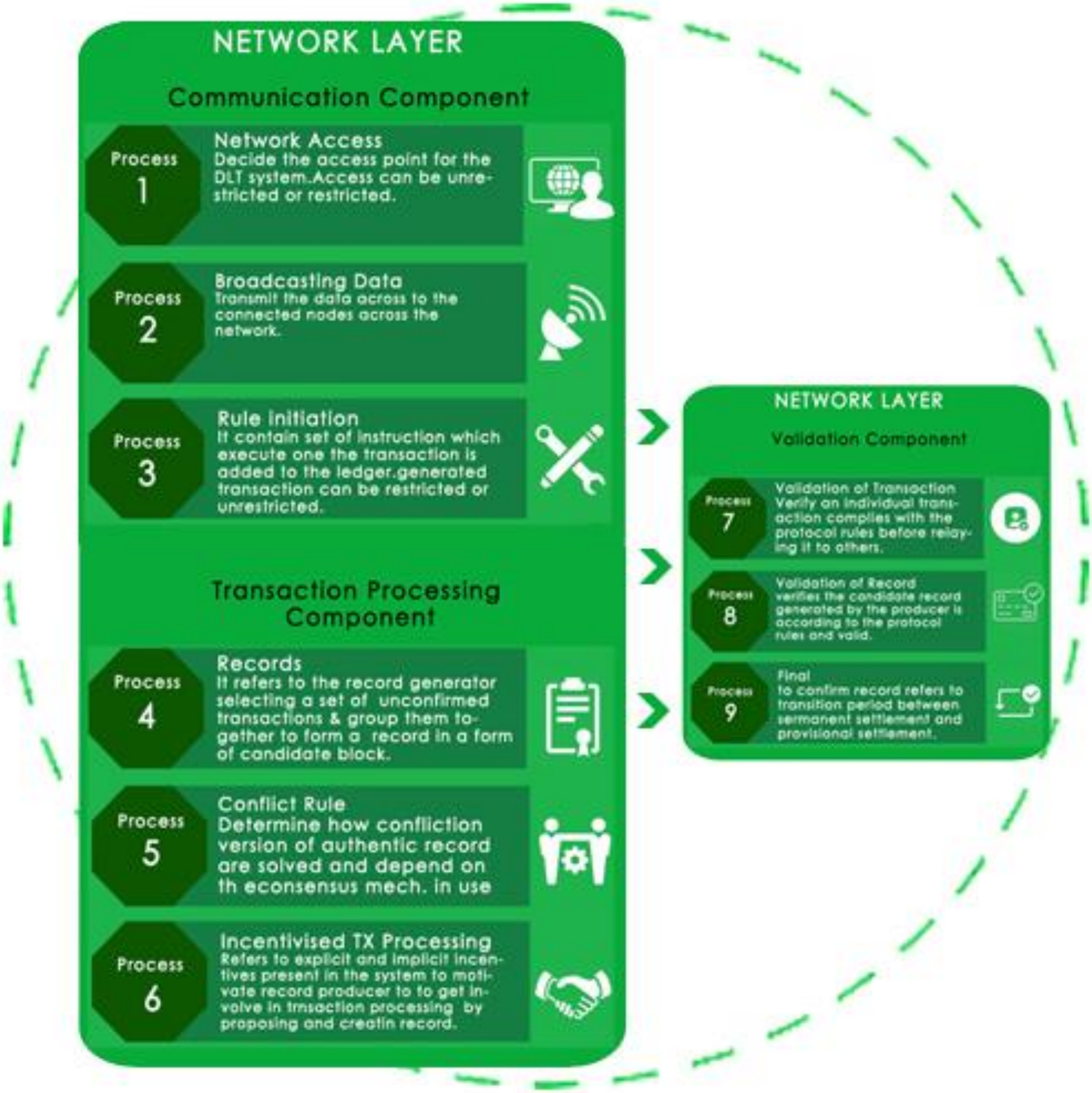
Protocol Governance
Alludes to the set of decisionmaking process required to change the protocol in a deliberate and authentic manner.

Process 5

Protocol Change
Alludes to the methods for executing the proposed changes o the protocol rules

Figure 8 - Protocol Layer

The network comes as an immediate outcome from the usage of the protocol rules. The network comprises of an interconnected gathering of actors and process that hold fast to an innovation standard (protocol) and effectively take an interest in the exchange of information and data.



NETWORK LAYER

Communication Component

Process 1
Network Access
Decide the access point for the DLT system.Access can be unrestricted or restricted.

Process 2
Broadcasting Data
Transmit the data across to the connected nodes across the network.

Process 3
Rule initiation
It contain set of instruction which execute one the transaction is added to the ledger.generated transaction can be restricted or unrestricted.

Transaction Processing Component

Process 4
Records
It refers to the record generator selecting a set of unconfirmed transactions & group them together to form a record in a form of candidate block.

Process 5
Conflict Rule
Determine how confliction version of authentic record are solved and depend on th econsensus mech. in use

Process 6
Incentivised TX Processing
Refers to explicit and implicit incentives present in the system to motivate record producer to to get involve in trnsaction processing by proposing and creatin record.

NETWORK LAYER

Validation Component

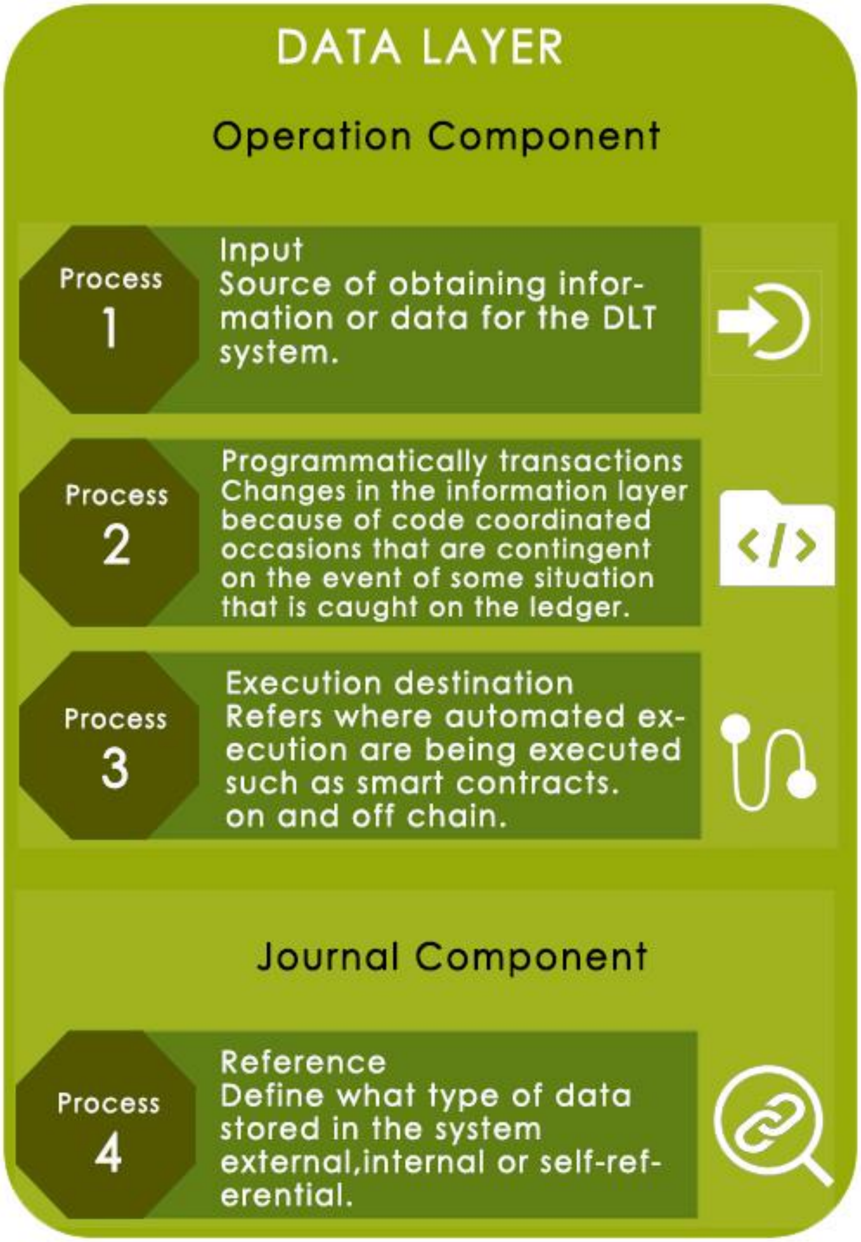
Process 7
Validation of Transaction
Verify an individual transaction complies with the protocol rules before relaying it to others.

Process 8
Validation of Record
verifies the candidate record generated by the producer is according to the protocol rules and valid.

Process 9
Final
to confirm record refers to transition period between sermanent settlement and provisional settlement.

Figure 9 - Network layer

Together, the protocol layer and network layer endorse the development of the data layer which is collected after over time as the transaction is composed into the ledger by the exercises of members utilizing the DLT framework.



DATA LAYER

Operation Component

Process 1

Input
Source of obtaining information or data for the DLT system.

Process 2

Programmatically transactions
Changes in the information layer because of code coordinated occasions that are contingent on the event of some situation that is caught on the ledger.

Process 3

Execution destination
Refers where automated execution are being executed such as smart contracts. on and off chain.

Journal Component

Process 4

Reference
Define what type of data stored in the system external,internal or self-referential.

Figure 10- Data layer

Chapter 3 - DLT Framework Interaction

3.1 Within the Structure Boundaries

3.1.1 Interdependencies of layer

DLT frameworks comprise of three layers that are reliant in the sense that the *lower layers* of the framework make the *higher levels* conceivable. The requesting isn't spatial but instead reflects ideal and functional conditions.

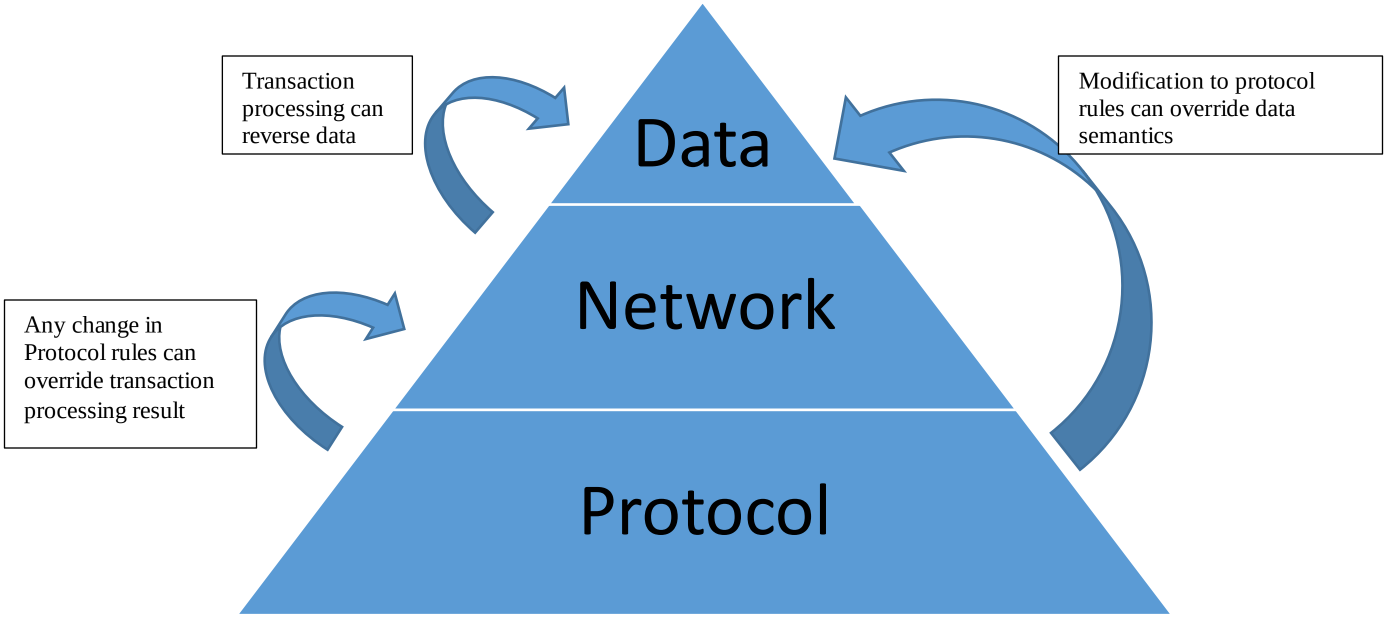
The protocol layer outlines the ruleset overseeing the operation of the system network of interconnected members. The protocol represented network layer, like this, host the data layer that records the time-requested entries and adjustments to the ledger.

A protocol is only a bit of programming which without by itself is passive. A protocol is 'enlivened' when a system executes it or implemented it. A network is an arrangement of free servers and capacity that takes an interest in protocol categorized tasks. In contrast to numerous conventional IT models, where the storage and servers are entirely claimed, operated, and kept up by single corporate or government element a DLT network includes a gathering of heterogeneous members who do not know or confide in each other ex-ante yet who contribute assets to the system in return for the value gained from the contribution in DLT system.

The network and protocol layers, thus, empower the development and maintenance of the data layer. A shared database made by multi-party consensus and having unique properties, for example, alter resistance.

3.1.2 Hierarchy of layers

Evaluating the relationship between the layers is important, and see how these effect one another, while considering the flexibility, robustness and alter obstruction of a DLT framework


Transaction processing can reverse data
Modification to protocol rules can override data semantics
Any change in Protocol rules can override transaction processing result
Data
Network
Protocol

The system can affect the data layer as it forms transaction, plotting record makers can choose to control self-assertive information by overlooking and declining to transfer comparing transaction (for example not adding them to records). This implies regardless of the data layer being apparently permissionless (enabling anybody to build applications over the DLT framework), it risks being edited or controlled by conspiring record makers. The protocol layer can affect both the network and the data layer. Since the protocol indicates the principles under which the framework works, an alteration in rules can abrogate choices taken by record makers at the data layer amid the transaction preparing process. Also, altering protocol tenets can change the semantics of handled information and abrogate past arrangements at the data layer.[8]

Decisions and actions at the data and network layers can always be overruled by the protocol layer

It pursues that whoever has power over the convention layer can impact specifically both the system and the information layer. Choices that are taken at the system layer commonly just effect the information layer, yet in specific cases, either layer can be utilized to organize convention changes over a system (for example Bitcoin's BIP flagging procedure; Decred's on-chain administration casting a ballot model). This implies a framework really versatile to external obstruction needs adequate decentralization at both the protocol layer and the network layer so as to maintain a strategic distance from single-party oversight and control. For instance, specific blocks or transactions can be 'blacklisted' at the protocol level. A decentralized network layer over a central protocol layer is constantly helpless to subjective rule changes that abrogate consensus choices taken by record makers.

3.1.3 Designing structure

'One Size Fits All' - not in this case.

Diverse objective requires distinctive structure decisions. Plan arrangements at one layer of a DLT framework can affect different layers or parts and lead to various framework qualities, forcing a trade-off of costs and benefits. Each framework makes these trade-offs as per their goals and their security, trust, and risk models. A framework may support a particular property, yet that decision will unavoidably come to the detriment of another. For example, the existence of trust in the framework (e.g., distinguished, managed elements in a closed DLT framework) takes into account a more adaptable design approach than a DLT framework worked to limit the trust prerequisite between members (Bitcoin).

In the beginning, DLT frameworks put specific accentuation on keeping all parts of their framework 'decentralized,' in order to enhance the networks censorship opposition which came at a huge cost: wasteful repetition, moderate affirmation speed, natural scaling constraints, low throughput, high vitality expenses, and poor client encounter. Consequent DLT frameworks have looked to address a portion of these issues, yet these structure decisions come to the detriment of other framework properties or an increase in the framework's centralization.

Each plan choice includes a complex arrangement of trade-offs

With modern innovation, trade-offs most frequently rotate around a similar arrangement of properties (for example decentralization, speed validation, security, multifaceted nature, throughput, trust needs, size of the network). The decentralization trade-off has been the most deliberated: for the most part, the more incorporated the DLT framework, the quicker, less expensive, and all the more effectively it runs.

Use case prerequisites should manage structure decisions, and satisfactory exchange offs.

It is exceptional for a design decision to rule another entirely. Commonly, one cannot get every one of the advantages without any drawbacks. Consequently, one must know about the tradeoffs included while investigating explicit structure choices, and cautiously assess whether the subsequent tradeoffs are worthy. At last, a DLT framework is intended to fill a particular need: that reason should direct design decisions and satisfactory tradeoffs.

3.2 Beyond the structure boundaries

3.2.1 Structure Perception

DLT frameworks are only from time to time independent. Rather, they are frequently in steady communication with different frameworks. The figure portrays the distinctive kinds of frameworks arrangements seen in DLT organizations.

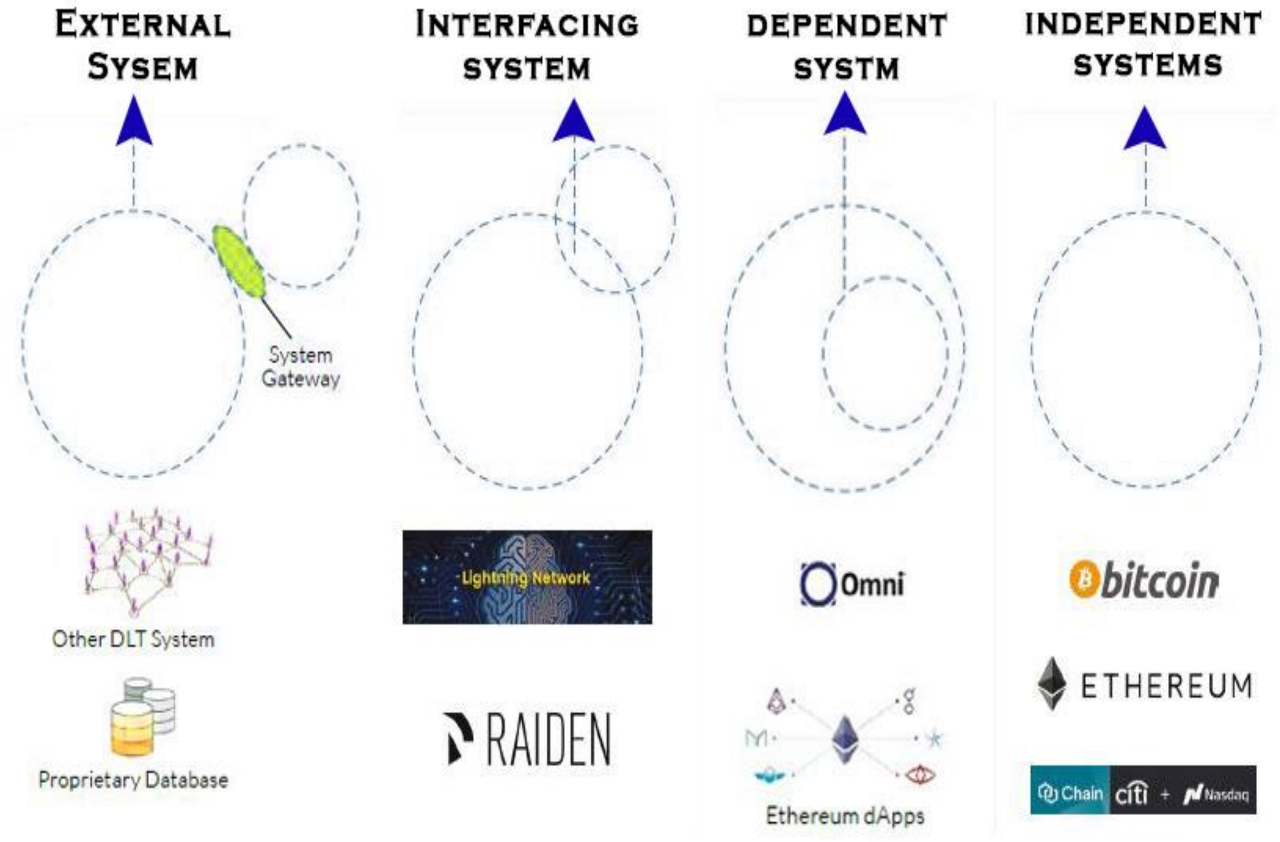


Figure 11-Structure Perspective

External Systems

An external system is hooked or combined with, a 'central' DLT framework. The external framework is structurally disconnected to or distinct from, the central DLT framework. An external system can be associated with the framework being referred to by means of a gateways (either through an indirect or direct interface). This could be other DLT frameworks just as restrictive databases or services (for example exchange, wallets or applications). A case of a direct framework gateway would be an atomic swap, though an indirect framework gateway would include a trusted intermediary to exchange tokens from an exclusive database to the framework, or between two dissimilar DLT frameworks.

Interfacing Systems

An interfacing DLT system is a system that 'deftly' utilizes core functionality delivered by another DLT system yet which could simply be reconfigured to utilize another 'base-layer' DLT system if necessary. This implies that if one framework stopped to exist, the interfacing framework would probably get by for probably some time without anyone else and might almost certainly keep working by misusing the elements of an option 'base layer' DLT. The long haul survival of an interfacing framework relies upon the proceeded with presence of something like one 'base-layer' DLT framework, and a breakdown of a base-layer framework may make huge disturbance the interfacing framework. Precedents incorporate 'layer-2' arrangements, for example, the Lightning Network reliant on Bitcoin and the Raiden Network dependent on Ethereum. These frameworks are generally intended to enhance the adaptability and usefulness of the base layer, without compromising off network decentralization or security.

These network are based on the individual stage and depend on a particular idea called 'state channels'. Basically, parties 'route' payments by trading signed transactions among one another off-chain and just communicated back to the DLT framework to open or close channels. This takes into consideration cost-efficient and real-time transactions by changing over the base layer from the 'cash' layer to a 'settlement' layer.

Subordinate Systems

A dependent DLT system must interact with another DLT system so as to work legitimately. All alone, such a framework isn't independent. Instances of dependent frameworks are Omni and Counterparty which function over Bitcoin just as the dApps ('decentralized applications') running on Ethereum. Omni, for example, is completely reliant on Bitcoin, as it is a protocol which tracks resources that exist as subjective information inside certain Bitcoin transactions. Omni gets its conclusiveness and security properties from Bitcoin while accumulating semantic substance to transaction, it doesn't exist outside of Bitcoin.

Omni use the OP_RETURN opcode to accomplish this impact. The bit strings inserted in OP_RETURN outputs are on the whole comprehended by clients of the Omni protocol as portrayals of benefits, however to other, uninterested Bitcoin clients not interfacing through the Omni protocol, they look like normal transaction (yet with some installed metadata), and are treated in that capacity. The outputs of these exchanges are generally called 'colored coins'

Self-Sufficient/Independent Systems

An independent DLT system has the majority of the component needed for its continued task linked into its essential engineering, and the framework itself is adequate to empower the core functionality. Such frameworks don't rely upon different frameworks for their activity, aside from the more extensive Internet foundation (for example dependence on TCP/IP or comparable protocol and the fundamental system foundation). Open system, for example, the Bitcoin and Ethereum primary nets just as permissioned frameworks, for example, the NASDAQ Linq blockchain.

Contingent upon the idea of the records (for example exogenous/external), a framework may require input from outer sources. This prerequisite alone is deficient to block characterizing a DLT framework as independent. For instance, a DLT framework signifying asset transfer in a supply-chain network ought to almost certainly hold on and work regardless of whether external information isn't received, despite the fact that it will rely upon gateways or interfaces to supply data relating to the creation or physical transfer of assets.

3.2.2 Exogenous and Endogenous References

Records may point toward endogenous data as well as exogenous data. Endogenous data is information that comes solely from inside the core framework. Exogenous information alludes to information that tracks data about a similar substance or a relationship that is external to the DLT framework. Exogenous entries might be demonstrations of assets (nonmonetary or monetary), or other information. A case of endogenous information would be a record of bitcoin units inside the Bitcoin framework, while a case of exogenous information could be a record following extravagance handbags on a worldwide supply-chain network.

Mostly, if the information is in only journal then it only alludes to actualities about user activities on the platform, or detail about the previous history of the DLT framework itself, at that point the reference type is endogenous. Whereas if the information alludes to some state in the world external to the DLT framework or the user collaboration with the DLT framework, at that point the reference type is exogenous.

Figure 12 gives a portrayal of the pathways for information cooperation between the DLT framework and outer stages.

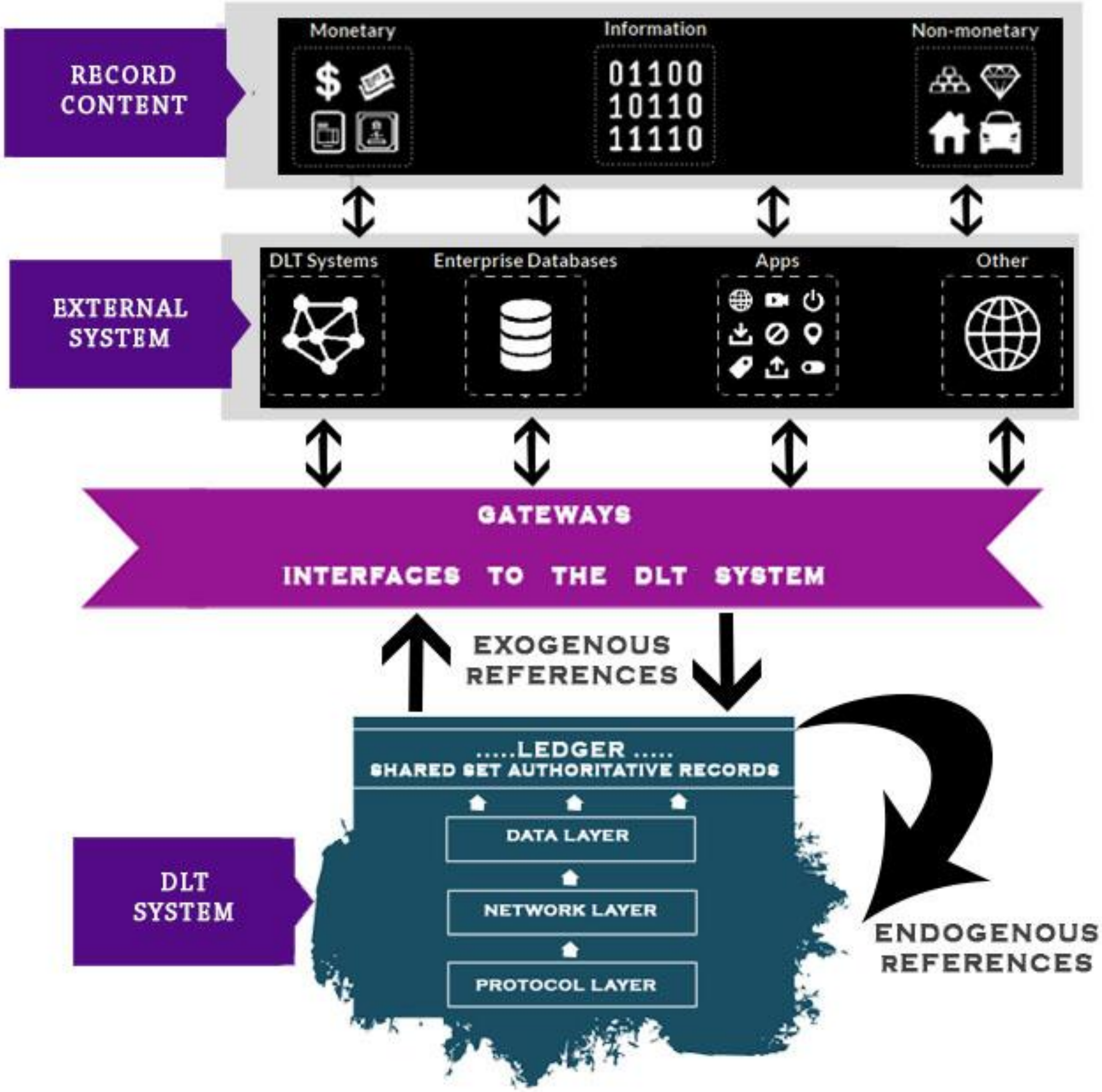


Figure 12-Structure Perspective

The distinction among exogenous and endogenous information may appear to be thin, however it isn't. Bitcoins, for instance, just exist as information records inside the Bitcoin DLT framework. The best way to change its state is to change the information record inside the Bitcoin framework stock costs, handbags or climate readings are the example of things that exist self-sufficient of the DLT framework, and whose state can change without modifying records inside the DLT framework that tracks them.

Connecting a DLT with an external framework involves gateways to go about as an interface, oracle overcome gap between the DLT framework and external frameworks by filling in as a wellspring of data. On account of a supply-chain network, this could be RFID tags appended to the extravagance goods and checked by machines at each irregular station. Other outside frameworks (for example other DLT frameworks, applications, exclusive endeavor databases, and so on.) may discuss their very own recorded data with the first DLT framework, giving information that turn out to be a part of it.

Interfacing with an outer wellspring of information requires a passage; this undermines the capacity of a DLT framework to consequently and autonomously authorize choices

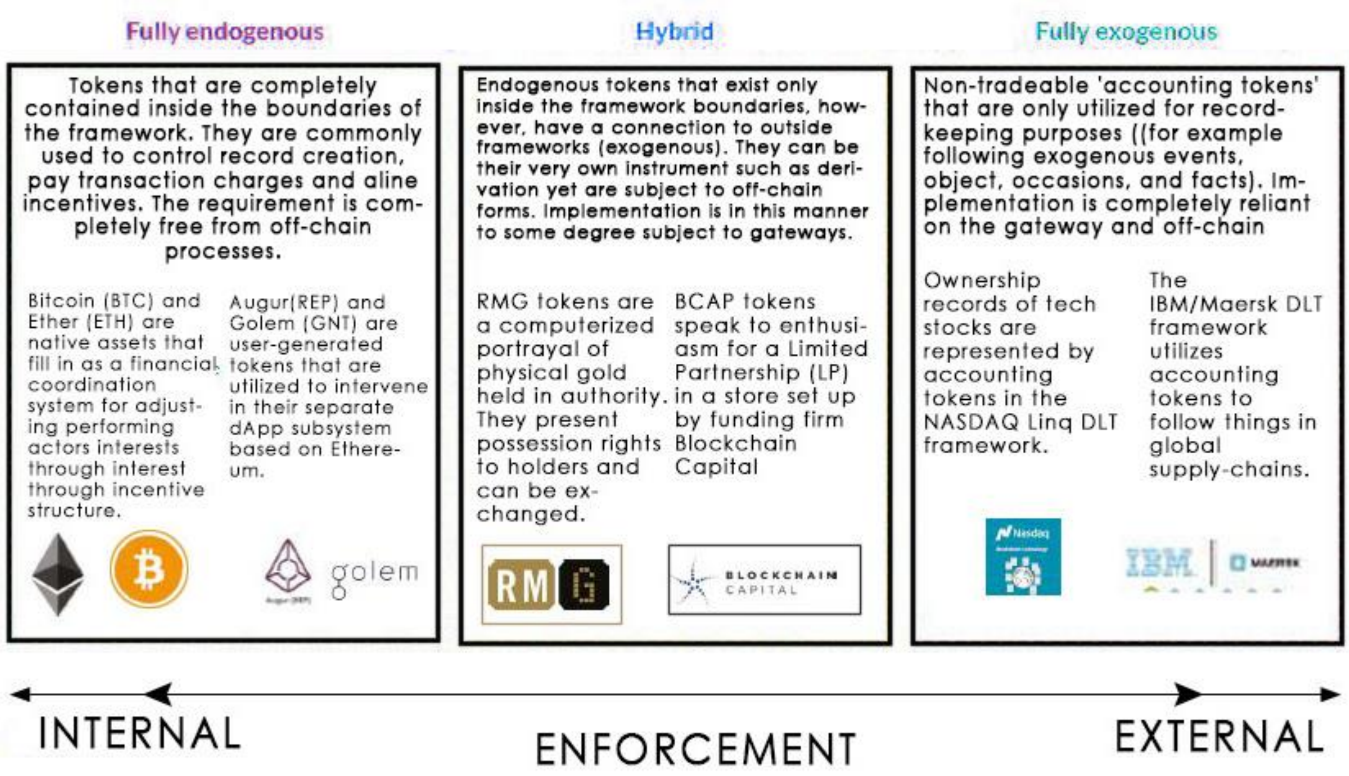
Utilizing the case of a supply-chain network, a DLT framework may appropriately record the development of RFID labels, however those gadgets may not really be connected to (or implanted in) the articles they are taken to speak to one could envision a delivery crate loaded up with only RFID labels that could trick the DLT framework into tolerating a fake transaction representing a sizeable transaction of physical resources. Likewise, some number of RFID labels might be poor, and exchanges would not really be recorded.

A DLT framework just has successful implementation abilities (for example the capacity to naturally execute decisions) concerning endogenous information (for example internal references that solely exist inside the limits of the framework). Records referencing exogenous assets, realities, or occasions are given by external agents who must be depended or trusted through non-system, intends to report sincerely as well as uphold decisions. In the earlier supply-chain network model, parties with a mutual enthusiasm for legitimately recording the exchange would need to create frameworks to avoid or improve any breakdowns, for example, coupling the RFID interface with a physical investigation.

A DLT framework can just freely and independently upholds decisions that include endogenous record references

What can be carved to the journal are eventually controlled by the protocol. This doesn't mean, in any case, that the protocol fundamentally expressly spreads out the majority of the information types that can be recorded by the DLT framework. For instance, a DLT framework capable for supporting a Turing-complete smart contract delivers its users with the adaptability to characterize novel data-types for the smart contracts that they make. At last, records may likewise reference information that convey parts of both endogenous and exogenous nature, in which case they are alluded to as ***hybrid***. A precedent would be a security directly issued on a DLT framework (endogenous in light of the fact that it solely exists inside the framework boundaries) that is subject to off-chain cash flows (exogenous, as it requires an association with an external framework). On account of hybrid references, it is increasingly hard to decide the precise implementation abilities of the DLT framework in because the connection between the two viewpoints may change starting with one record then onto the next.

Hybrid references are a fast-developing subfield as companies are progressively endeavoring to change over existing resources on to a DLT framework. As such, it might require further gradation later on. Figure 13 condenses the three sorts of references that records in a DLT framework can point to. Native resources are completely endogenous as they are totally contained inside the boundaries of the framework and don't require a formal relationship with the external world. Conversely, completely exogenous records are solely referencing external data, which requires the presence of a gateway to receive information and to enforce decisions outside the DLT framework. Exogenous information is useless inside the framework without an appended connection covering to the material world. Interestingly, hybrid records reference information which shares both exogenous and endogenous attributes. Accordingly, implementation is to some degree subject to gateway.



Tokens that are completely contained inside the boundaries of the framework. They are commonly used to control record creation, pay transaction charges and aline incentives. The requirement is completely free from off-chain processes.

Bitcoin (BTC) and Ether (ETH) are native assets that fill in as a financial coordination system for adjusting performing actors interests through interest through incentive structure.

Augur(REP) and Golem (GNT) are user-generated tokens that are utilized to intervene in their separate dApp subsystem based on Ethereum.

Endogenous tokens that exist only inside the framework boundaries, however, have a connection to outside frameworks (exogenous). They can be their very own instrument such as derivation yet are subject to off-chain forms. Implementation is in this manner to some degree subject to gateways.

RMG tokens are a computerized portrayal of physical gold held in authority. They present possession rights to holders and can be exchanged.

BCAP tokens speak to enthusiasm for a Limited Partnership (LP) in a store set up by funding firm Blockchain Capital

Non-tradeable 'accounting tokens' that are only utilized for record-keeping purposes ((for example following exogenous events, object, occasions, and facts). Implementation is completely reliant on the gateway and off-chain

Ownership records of tech stocks are represented by accounting tokens in the NASDAQ Linq DLT framework.

The IBM/Maersk DLT framework utilizes accounting tokens to follow things in global supply-chains.

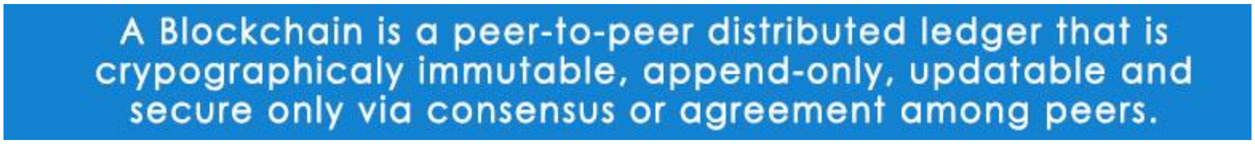
Figure 13- Types of References

Native Assets

A DLT framework's local resources are the essential computerized asset(s), assuming any, predefined in the convention. They are by definition endogenous to the framework. These advantages are commonly utilized by the convention to direct record generation, pay exchange charges on the system, lead 'financial approach', or adjust motivations. For instance, Ethereum's ETH token is its local resource, in spite of the fact that the Ethereum blockchain additionally has a wide scope of other client characterized tokens (utilizing the ERC20 standard, for instance). Local resources by and large assume a framework basic job in the working of the framework as they are a fundamental segment of the complex financial motivation structure.

Chapter 4 - Understanding Blockchain and its requirements

4.1 What is blockchain?



A Blockchain is a peer-to-peer distributed ledger that is crypographicaly immutable, append-only, updatable and secure only via consensus or agreement among peers.

Blockchain is stable digital ledger system it is the database that upholds a constantly growing list of records which are known as a block. The blockchain itself is made out of blocks, with each block demonstrating a set of transactions. As a data structure, a blockchain has a few interesting properties. First of all, blocks are provably unchangeable and secured from tampering once published. This is conceivable because each block holds a hash or numeric process of its content that can be utilized to confirm the reliability of the transaction. Then, the hash of a block is reliant on the previous hash of the block. This successfully makes the whole blockchain history permanent, as altering the hash of any square $\boldsymbol{m} - \boldsymbol{i}$ would likewise change the hash of block $\boldsymbol{m}$.

The blockchain itself does not rely upon a central, trusted party. Fairly, it is adopted by all nodes partaking in the network. Since no centralized professional may confirm the legitimacy of the blockchain, a module for achieving system consensus must be utilized. In Bitcoin, a Proof of work is utilized to guarantee organize consensus[9]. This methodology necessitates that any node wishing to add a block to the blockchain must finish a computationally costly puzzle first (however effortlessly verifiable). At an abnormal state, this guarantees a consensus of the system because there is an opening cost (the calculation time) to constructing a block. There are a few different strategies utilized, such as Proof of Activity [10]and Proof of Stake[11] and. However all are intended to drive the network to consensus on blockchain legitimacy.

Miners are nodes that collect and assembles the block and merge them into the blockchain. It is through the miners that the consensus methodology is authorized. In Bitcoin, for instance, miners are boosted by gathering transaction charges and furthermore by a reward for adding the block to the blockchain. In any case, there should exist a motivation for them to just expand over substantial blocks, which thusly drives the whole network to consensus12.

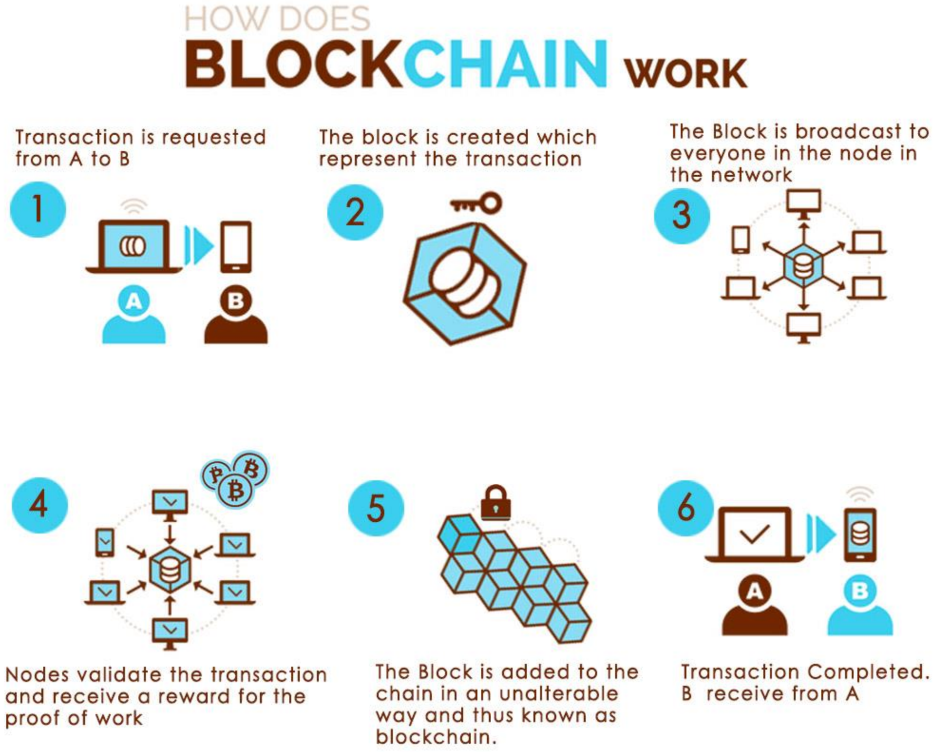

HOW DOES
BLOCKCHAIN WORK
Transaction is requested from A to B
1
A
B
The block is created which represent the transaction
2
The Block is broadcast to everyone in the node in the network
3
4
Nodes validate the transaction and receive a reward for the proof of work
5
The Block is added to the chain in an unalterable way and thus known as blockchain.
6
A
B
Transaction Completed. B receive from A

Figure 14- How does blockchain works?

4.2 Fundamental of Blockchain and some primary advantages

Business Transaction is recorded in immutable blocks to the ledgers. All the affirmed and approved transaction blocks are connected from the genesis block which is the first one to the most current block with each block connected to its past block utilizing the cryptographic hash of the past block.

A blockchain is a registered record of the considerable number of transaction that has occurred in the network since the start of the blockchain. The blockchain fills in as a solitary wellspring of accuracy for the network.

The blockchain grasps the possibility of disrupting any type of transaction that expects data to be trusted. This implies all mediators of trust, as they exist today, are presented to disrupting in some shape with the approach of blockchain innovation. The figure underneath shows how blockchain innovation is settling issues with the way data related transaction happen today. The crates on the left represent vital issues, while those on the right outline how blockchain innovation helps address them.

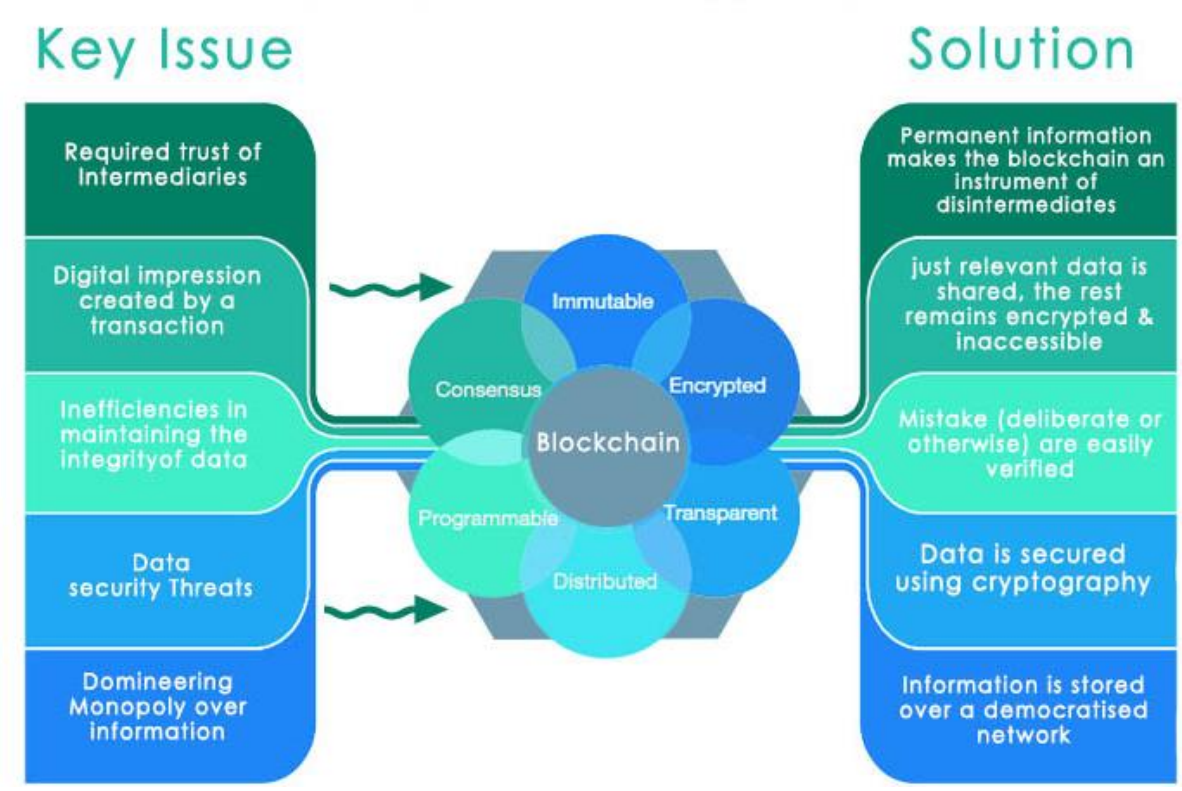

Key Issue
Solution
Required trust of Intermediaries
Digital impression created by a transaction
Inefficiencies in maintaining the integrityof data
Data security Threats
Domineering Monopoly over information
Immutable
Encrypted
Consensus
Blockchain
Programmable
Transparent
Distributed
Permanent information makes the blockchain an instrument of disintermediates
just relevant data is shared, the rest remains encrypted & inaccessible
Mistake (deliberate or otherwise) are easily verified
Data is secured using cryptography
Information is stored over a democratised network

Figure 15- Blockchain advantage

4.3 High-level blockchain network

Figure 16 demonstrates the essential segments that involve a blockchain and its nature. There is numerous minor departure from this basic conceptual structure that includes different features, yet the diagram is a helpful method to present the way that blockchain's work.

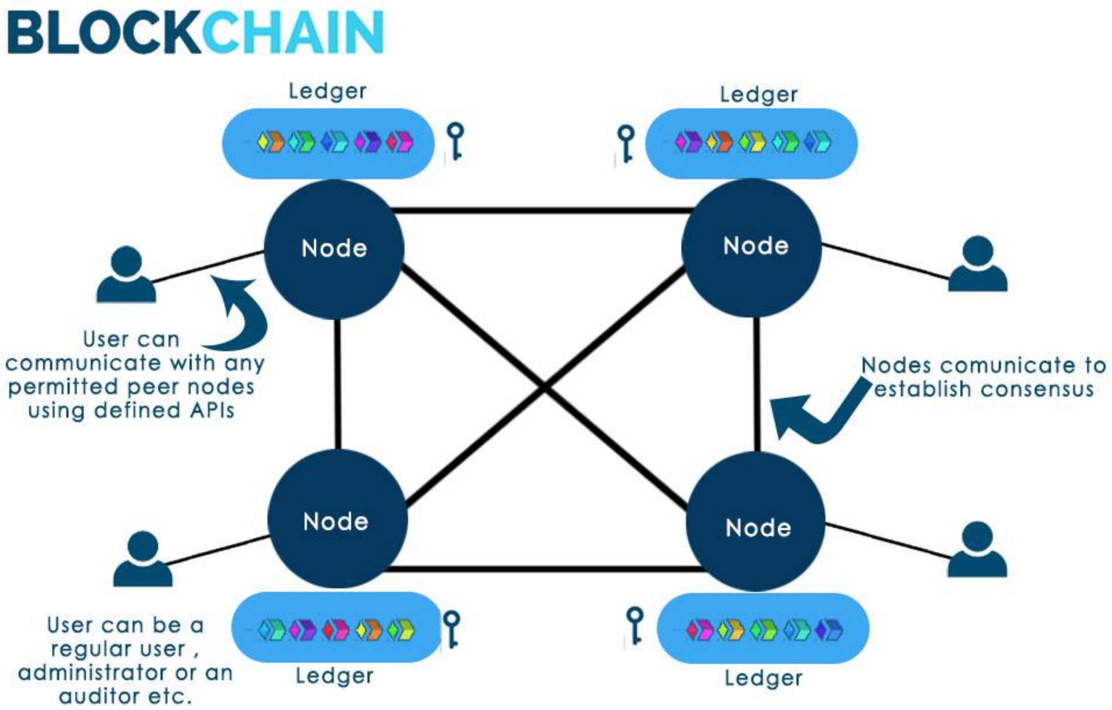

BLOCKCHAIN
Ledger
Ledger
Node
Node
User can communicate with any permitted peer nodes using defined APIs
Nodes comunicate to establish consensus
Node
Node
User can be a regular user , administrator or an auditor etc.
Ledger
Ledger

Figure 16- Generalized Blockchain component.[13]

A blockchain framework comprises of various nodes, every one of which has a neighborhood duplicate of a ledger. In many frameworks, the nodes have a place with various organization. The nodes speak with one another to pick up concurrence on the substance of the ledger and don't require a central authority to organize and approve transactions.

The method of gaining the agreement is called **consensus**, and various distinctive algorithms have been produced for this reason. Clients send **transaction** solicitations to the blockchain in order to perform the activities the chain is intended to provide. When a transaction is finished, a record of the transaction is added to at least one of the ledger and can never be adjusted or expelled. This nature of the blockchain is called **immutability**

Cryptography is utilized to secure the blockchain itself and the communications among the components of the blockchain system. It guarantees that the ledger cannot be tampered, except by the expansion of a new transaction. Cryptography gives trustworthiness on messages from users or among nodes and guarantees tasks are just performed by approved entities.

The specialist to perform a transaction on a blockchain can utilize one of two models, *permissioned* or *permissionless*. In a permissioned blockchain, users must be enlisted or enrolled in the blockchain before they are permitted to perform transactions. The enrollment procedure gives the user authorizations that are utilized to recognize the user when the individual performs exchanges. In a permissionless blockchain, any individual can perform a transaction, yet they are generally confined from performing tasks on any information however their own.

Many business-oriented blockchain incorporate the capacity to utilize **smart contracts**, now and then called **chain code**. A smart contract is an executable programming module that is produced by the blockchain proprietors, introduced into the blockchain itself and implemented when pre-characterized rules are met. At the point when an individual sends a transaction to the blockchain, it can summon a smart contract module which performs the function defined by the maker of that module. Smart contracts, as a rule, can write or read to a local information store which is discrete from the blockchain itself and can be refreshed when the transaction happen. The business foundation contained in a smart contract makes or works on business information that is contained in this information store.

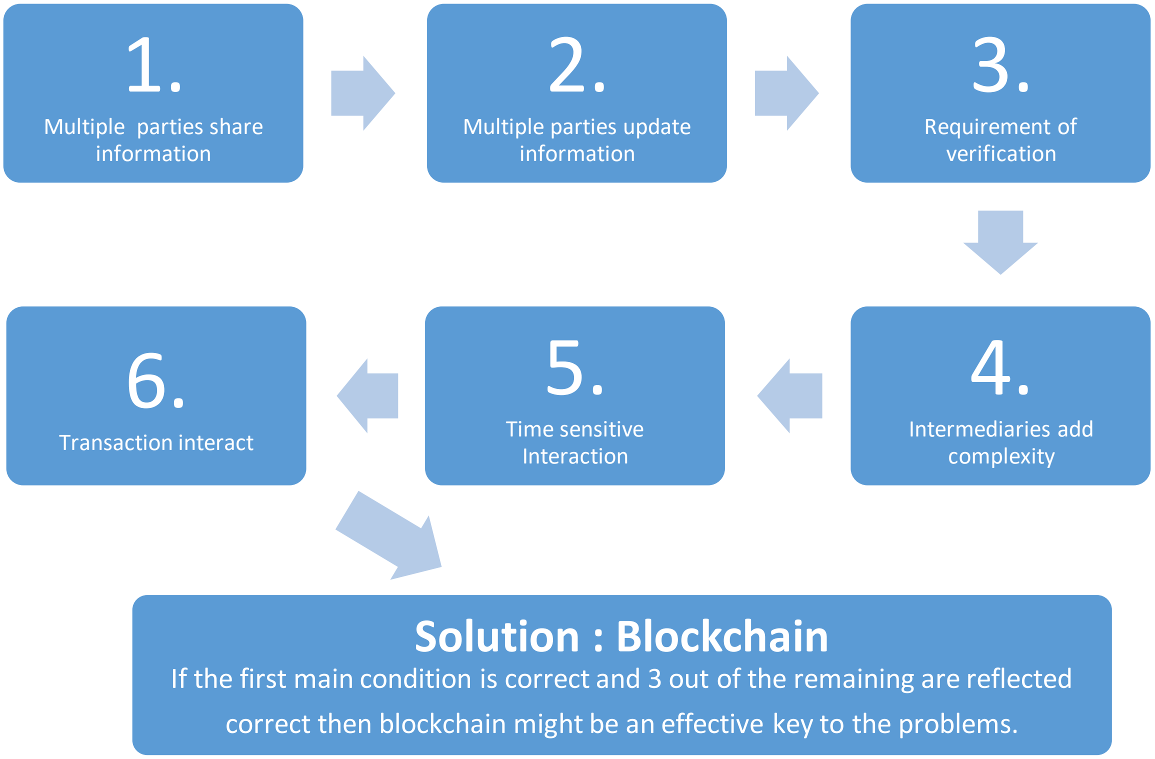
In a simple blockchain, each node is indistinguishable, and each duplicate of the ledger is indistinguishable. Nonetheless, increasingly complex blockchain permit contrasts in the nodes and the records. Some blockchain encourage the idea of **subchains**, sometimes also known as **channels**.

Subchains are consistently independent chains that involve the corresponding physical blockchain. An alternate entity might claim each subchain and may be open to another set of users. Nodes might be set up with the goal that a few nodes take part in certain subchains and not in different subchains.

The consequence of this architect is that the ledger on a few nodes will contain transactions for that subchain while the ledger on different nodes will not. Another minor departure from the simple blockchain is one in which nodes are assigned explicit purposes as opposed to being indistinguishable in their function. This configuration might be utilized to improve performance and execution since the system can be quicker if each node does not need to perform each task required for a transaction on the chain.

4.4 Fundamental requirements of blockchain arrangement

Blockchain innovation is not a one-stop solution for all issues that emerge out of transacting with information and resources. It cannot be actualized in all utilization cases for digitization. One must comprehend blockchain, its characteristics and recognize utilizes situations where this arrangement will be practical and valuable. There are a few simple requirements that can actually determine whether blockchain technology is feasible. The chart below shows the elements that can be utilized to decide the capability of blockchain as an answer.


1.
Multiple parties share information
2.
Multiple parties update information
3.
Requirement of verification
4.
Intermediaries add complexity
5.
Time sensitive Interaction
6.
Transaction interact
Solution : Blockchain
If the first main condition is correct and 3 out of the remaining are reflected correct then blockchain might be an effective key to the problems.

Blockchain invention is really just when various gatherings share information and need a perspective of regular data. However, different gatherings sharing information is not the main passing criteria for blockchain to be a reasonable arrangement. To recognize the adequacy of a blockchain arrangement, we observationally characterize a cut-off of three out of the accompanying five achievement criteria:

Multiple parties update information

At the point when activities attempted by different gatherings should be recorded and the information originating from various gatherings need to be updated.

Requirement of verification

When it is essential to build trust among gatherings and influence them to comprehend that their activities that are being recorded are substantial.

Intermediaries add complexity

At the point when an exchange is reliant on various middle people, and their essence expands the expense and multifaceted nature of the transaction.

Time-sensitive Interaction

When it is profitable for the business to minimize delay and speed up a transaction.

Transaction Interact

At the point when transaction made by different members collaborate and rely upon one another.

4.5 Basic features of a blockchain network

There are a few qualities that apply to Blockchain frameworks that influence their design and execution:

- **Cryptography:** Blockchain Transactions accomplish legitimacy, trust, and completeness dependent on cryptographic verifications and hidden scientific calculations between different exchanging and trading accomplices.

- **Immutability:** This term specifies that blockchain exchanges cannot be erased or adjusted.

- **Provenance:** In a blockchain ledger, provenance is an approach to follow the cause of each transaction with the end goal that there is no debate about the starting point and succession of the transactions in the record.

- **Decentralized registering structure:** These computing infrastructures include processing nodes that are fit for settling on independent processing and computational choices regardless of what other companion processing nodes may choose.

- **Distributed transaction stage:** This platform handles a series of transaction, including trading worth, resources, or different substances.

- **Decentralized database:** Each participant accomplice approaches a conveyed database entirely consistently. No single participant controls the database, which each participant can recover or confirm whenever required without having a central party.

- **Shared and distributed secretarial ledger:** These ledgers can be private, public or semi-private/public. Ledgers can be shared among members with security and privacy. In permissioned blockchain, members can see the transaction completely with authorization and still look after obscurity. These transactions are conclusive and irreversible since every transaction are connected to each previous transaction in the record. The ledger sections are time ordered and computationally and cryptographically designed to guarantee changelessness, and the ledger itself is broadly imitated.

- **Software development platform:** This platform makes utilization of APIs, shared network (P2P), and private or public, or hybrid systems. Transactions are programmable since the original record is computerized in nature, which prompts smart and programmable contracts and contracts requirements.

- **Cloud computing:** Blockchain frameworks often include the utilization of distributed computing platform. Distributed computing offers the possibility to utilize many assets in connection to information data storage and furthermore the capacity to convey adaptable and versatile handling assets to the examination of data.

- **Peer-to-peer networks:** In these network active nodes communicate with one another specifically and without a central node or substance.

- **Wallet:** An anchored information store of access accreditations of a client and related information, which incorporates client IDs, passwords, authentications, and encryption keys.

Blockchain network operation attempt to adaptability and simultaneousness, guarantee no single purpose of failure, and incorporate pluggable segments like databases and different consensus mechanism. Effective usage supports staggered secrecy and protection which is accomplished through multichannel or sub-chain correspondence, different sub-records, and numerous stakeholder for transaction permeability dependent on a need-to-know-premise.

Chapter 5 – Cryptography and its construction

5.1 Introduction

Blockchain frameworks can appear to be mind-boggling, in any case, they can be adequately comprehended by looking at every part innovation exclusively. At a high-level state, blockchain use recognized software engineering systems (connected records, appropriated systems administration) along with cryptographic primitives (digital signature, hashing, and private/public keys) blended with monetary concepts, for example, ledgers.

5.2 Peer to peer Architecture

P2P Stands for peer-to-peer. The peers are computer systems in a p2p network that are connected via the internet to each other. Files can be shared between the networks directly to each peer without a centralized system. This p2p attribute is the foundation upon which the blockchain's architecture resides. The establishment of the P2P network and its central role in blockchain technology could be as inviting an innovative communication system. Blockchain no longer requires confidence in the all - powerful third parties as users can interact directly with each other across a secure distributed and decentralized network. Even though each peer's participation in the network is open to examining, through extremely sophisticated, state-of-the-art cryptography, all information, and identities of participants are entirely concealed on a blockchain.

In a P2P network, the user simultaneously uses and provides the foundation of the network. However, it is entirely voluntary to provide resources. Each peer is treated as equivalent and generally known as nodes. A peer makes a portion of computer resources, such as processing power, disk storage or network bandwidth, accessible directly to other participants without any need for a stable host’s or central server coordination. Although all nodes are equal, they can assume different roles within the network of the blockchain, like a miner's or a "full node." The entire blockchain is copied to a single device in the case of a complete node even though the device is associated with the network. Which means that the information stored on a blockchain architecture can not be destroyed, lost or altered because doing so would mean terminating each full node from the network. Therefore, as long as there is a single node with a blockchain copy, all records will remain intact, providing that network can be rebuilt.

A Blockchain protocol functions over the internet, on a P2P network that are running the protocol and holding an identical duplicate of the transaction ledger, enabling transactions of P2P value without an intermediary by machine consensus. Blockchain are files that is either a shared and public transaction ledger that tracks every transaction from the block of genesis to the present day. The blockchain is a trusted, shared, public transaction ledger that can be inspected by everyone but controlled by no single user. It is a distributed database that keeps the list of transaction data records which are continually growing, secured cryptographically from manipulation and revision.

The ledger is manufactured utilizing a chain of blocks or linked list, where every block contains a specific number of exchanges that were approved by the system in a given duration. The crypto-financial standard arrangements of the blockchain protocol manage the rule sets and incentive mechanism of all stakeholders in the system.

Peer-to-peer networks differ entirely from the old-fashioned client-server models common today. No central storage point required in P2P, such as a server. As an alternative, information is continuously exchanged and recorded among all network members. This is also radically different from a centralized architecture that drops when more users go along with it, while on the other hand, the P2P network can radically increase its power by adding more nodes or devices. This data transmission technique is a massive development because information or data is not kept in a centralized system which makes it much less susceptible to misplace, hacked or exploitation[14].

5.3 Nodes in Blockchain network

Each member in a coin's network are known as node. There are diverse kinds, however every one of them shares one explicit characteristics – you'll require explicit equipment so as to have or just interface with one.

The nature of the Blockchain innovation is decentralized, one of the key properties that made it so pleasing to the public. It depends on the standards of a P2P (Peer to Peer) network. In many systems, there are no devoted servers, not one expert, however a consensus among clients. Just like all vital to the security and uprightness of the system, turning into an individual from a certain crypto coin network isn't just energizing yet in addition a duty.

Take Bitcoin for instance, you have two kinds of nodes.

- ***Full nodes*** which store a duplicate blockchain and in this way ensure the security and rightness of the information on the blockchain by approving information.

- The second kind is a ***lightweight node*** – every client partaking, who needs to interface with a full node so as to synchronize to the present condition of the system and have the capacity to take part

5.4 Types of Blockchain Nodes

More or less, there are two fundamental kinds of nodes – full nodes and light nodes. Another term to depict nodes is customers which supply wallet function. Full ones contain a duplicate of the blockchain's history, including all blocks made. Light nodes or Simple Payment Verification (SPV) nodes are for the most part wallets that download just the headers of blocks and spare hard drive space for clients. Following are the distinctive kinds of nodes in detail.

5.4.1 Full Nodes

Full nodes perform as a server in a decentralized system. Their fundamental undertakings incorporate keeping up the consensus between different nodes and confirmation of exchanges. They likewise store a duplicate of the blockchain, consequently being increasingly secure and empower custom capacities, for example, private and instant send transactions.

Full nodes are the ones recommended for the best future of the network. On the off chance that over 51% of them don't concur with the suggestion, it gets skipped. At times, this can prompt a hard fork in which the network can't concur on a specific change and along these lines go their different ways, making two chains. The most notable case of that incident is the *Bitcoin Cash Fork*.

1. Pruned Full Node

One kind is the pruned full node. The particular characteristics here is that it starts downloading blocks from the earliest starting point and once it achieves as far as possible, erases the most old ones, holding just their headers and chain situation. For instance, on the off chance that you set a size cutoff of 550MB, you will store all the most recent blocks that can fit in that hard drive space, however so as to get to that state, you would initially need to experience the whole blockchain to approve each one of those past blocks.

Pruned nodes are viewed as full nodes and consensusingly can likewise confirm exchanges and be engaged with the consensus.

2. Archival Full Node

Archival full nodes are what the vast majority refer to when discussing full nodes. They imagine a server which has the full blockchain in its database. As their fundamental undertaking is to keep up consensus and approve blocks. The contrast among pruned and Archival node is one – the portion of hard drive space they take on your PC or server.

Archival nodes can be separated into two or three subtypes – those that can add blocks to the blockchain and ones that can't.

Nodes Which Can Add Blocks

Give us a chance to start by covering the fundamental members in the blockchain – nodes which can add blocks to it. They rely upon the consensus rules being implemented and require something like one full authentic node to work.

i. Miners -Mining Nodes

An idea you may as of now be comfortable with, miners are really nodes either full or light ones who plan to demonstrate that they have finished the expected work to make a block. Thus the consensus name Proof of Work. To finish the assignment, as mineworkers need to either be an authentic full node themselves or get information from other full nodes on the system to know the present status of the blockchain and the required parameters for the following block in line.

Members in the process utilize equipment parts such as CPUs, ASICs or GPUs to tackle a cryptographic issue. The first individual to finish the task broadcast his outcomes to the system so it very well may be checked by full nodes and once consensus is accomplished – he is allowed the privilege to add a block to the current blockchain. For their work, miners are compensated with a pre-characterized measure of coins in addition to any exchange charges for the block. This set reward sum is called a coinbase transaction. Thinking of it as the primary transaction in the block, it is complimentary, as the miner himself made the block and included it.

ii. **Stakers - Staking Nodes**

Staking can be contrasted with having a conventional fiat cash deposit. You purchase coins and hold them, while consequently you get an interest back as a prize. While there are diverse assumes the Proof of Stake consensus instrument, the principle characteristics is that winning cash can be contrasted with taking an part in a lottery. Staking is a round of possibility, which while with a lower hindrance to entry, offers less assurance contrasted with mining and can be confounding on occasion.

The ultimate objective is to decide, in view of a pre-characterized set of standards and luck, who’s next to generate a block and get compensated. Components incorporate coin age, what number of you have and their proportion to accessible ones in the system. In staking, you needn't bother with any costly apparatus, you just keep your crypto wallet online all day, every day, which should be possible with a gadget like the Raspberry Pi.

iii. **Authority Nodes**

Network that make utilization of such algorithms need to characterize a fixed number of authority nodes. Who and how many they will be voted on by the network or characterized by the development group. The task of these nodes is, similarly as with full nodes, is to make and approve blocks, while in the meantime distributing data to clients on the network. All members, not picked to be an authority node, will run lightweight nodes “light nodes” which rely upon the broadcasted information to probably work on the blockchain.

iv. Master Nodes

Contrasted with full nodes, master nodes themselves can't add blocks to the blockchain. Their solitary reason for existing is to track exchanges and approve them. Regardless of whether it will be staker or miners, they're the ones composition blocks on the blockchain. An additional advantage, in any case, is that by running a master node, you secure the system as well as can gain an offer of the prizes for your administrations.

To set up a master node, you should bolt away a specific entirety of assets as security. You are required to be online every minute of every day and facilitating on a Virtual Private Server is viewed as great practice.

5.4.2 Lightweight (SPV) Nodes

Another sort of blockchain nodes, utilized in everyday crypto-operations, is the Simple Payment Verification (SPV) node or lightweight node.

These sorts of nodes communicate with the blockchain while depending on full nodes to furnish them with the essential data. As they don't store a duplicate of the chain, they just request the present status for which block is last, and processing broadcast transaction.

Note: it's obvious to see that running SPV node doesn't require numerous resources, however it sacrifices security for convenience.

5.4.4 Lightning Nodes

Lightning nodes are a fascinating idea. They fit into the imperatives of neither full nodes nor lightweight nodes.

The idea is to set up a relationship between clients outside the blockchain. Along these lines, the burden on the system is decreased, exchange times are abbreviated essentially and there's expanded ease of use of crypto coins. Transaction expenses are actually low in the lightning network, the likeness approximately 10-20 satoshi.

The manner in which it works is by opening a different payment channel between substances. Take for instance a food restaurant and Hammad. Hammad and the restaurant make something like a box of safe deposit such as multi-signature address to which they both have separate keys. Hammad stores his assets and utilizes them to pay for his food. Every exchange is settled upon by the two sides and happens practically quickly. When he's had enough food or essentially comes up short on cash, he or the restaurant can close the connection, take the most recent accounting report and broadcast it onto the system.

Thus waiting for every transaction will be affirmed and filling the system with space-wasting information, gatherings can interface between one another and bring down the heap on the blockchain. Besides, in the event that another person needs to manage a similar group, the lightning system will look for a way with minimal number of go-betweens and least transaction charges, in this way decreasing waiting time.

5.5 The end result for a Node after a Fork?

Here we will discuss how these nodes integrates with the consensus network and forks.
Forks plays an important role when large part of the community are willing to accept the alteration! A designer chooses to make new client, utilizing the source code of the coin and executes the proposed change. Clients willing to go on toward that path, download the new form and choose to help the now forked chain[15].

5.5.1 Hard Fork

A hard fork is a change to the system consensus algorithm. Each alteration that isn't perfect with the past adaptation of the customer utilized, is viewed as a hard fork. Parameters of the consensus that can incite a hard fork when changed, may incorporate another block reward, block time, progress from PoW to PoS, execution of master-nodes and others.

When a hard fork is propelled, each node on the system that hasn't refreshed to the new form of the product is dismissed by the consensus with respect to its working on invalid principles. That is one reason why community and developer try to avoid changes, as it implies that the transition phase may reduce the security of a system or few people may be left out.

5.5.2 Soft Fork

Another method for acquainting changes with a system is by means of a soft fork. In spite of hard forks, with this sort of modification, there's no compulsory principle for clients to refresh their nodes.

One such model is the expansion of the Segregated Witness highlight to BTC. Currently, transaction can be made on bitcoin’s blockchain with or without utilizing this component. Once 95% of the network clients are updated to the SegWit version, the consensus will naturally change and decline any old transaction. Thus, we have a smoother progress that does not compel clients to quickly update.

5.6 Crypto-currencies and cryptography

Monetary standards need some approach to control supply and authorize different security properties to avoid cheating. In fiat monetary forms, associations like central banks control the cash supply and include anti-counterfeiting highlights to physical cash. These security highlights increase present expectations for an assailant, in any case, they do not profit difficult to fake.

Eventually, the law requirement is essential for preventing individuals from defying the norms of the framework.

Cryptographic forms of money too should have safety efforts that keep individuals from altering the state of the framework, and from quibbling that is, putting forth commonly conflicting expressions to various individuals. For example, Rish persuades Hammad that she paid him an advanced coin, she will not be able to persuade Saqib that she paid him that equivalent coin. Yet, in contrast to fiat monetary forms, the security guidelines of cryptographic forms of money should be upheld simply mechanically and without depending on a focal expert.

As the word proposes, digital currencies make substantial utilization of cryptography. Cryptography gives an instrument for safely encrypting the rules of a cryptographic money framework in the framework itself. We can use it to counteract altering and evasion, just as to encode the standards for production of new units of the money into a scientific convention. Before we can appropriately comprehend digital forms of money at that point, we'll have to dig into the cryptographic establishments that they depend upon.

Cryptography is a profound scholarly research field using many progressed scientific methods that are famously unobtrusive and muddled to get it. Luckily, Bitcoin just depends on a bunch of generally basic and well-known cryptographic developments. In this section, we will explicitly think about cryptographic hashes and digital signatures, two primitives that end up being very valuable for building digital forms of money. Future parts will present progressively entangled cryptographic plans, for example, zero-knowledge verifications, that are utilized in proposed expansions and changes to Bitcoin.

When we have taken in the important cryptographic primitives, we will talk about a portion of the manners by which those are utilized to assemble cryptographic forms of money. We'll finish this part with a few instances of straightforward cryptographic forms of money that show a portion of the structural difficulties that we have to manage.

5.7 Hash function

The main cryptographic basic that we'll have to comprehend is a cryptographic hash function. A hash function is a scientific capacity with the accompanying three properties:

- Its (input) information can be any string of any size.
- It delivers a fixed size output. To make the dialog in this part solid, we will accept a 256-bit output size. Be that as it may, our exchange remains constant for any output estimate as long as it is adequately huge.
- It is computable. Of course, this means you can figure out what the output of the hash function is in a sensible measure of time for a given input string. All the more actually, processing the hash of an *n* -bit string ought to have a running time that is O(*n*).

Those properties characterize a general hash function, one that could be utilized to fabricate an information structure such as a hash table. We are going to concentrate only on a cryptographic hash function. In order to be cryptographically secure, a hash function must have three additional properties:

i. Collision-resistance.
ii. Hiding.
iii. Puzzle-friendliness.

We will look all the more carefully at every one of these properties to pick up comprehension of why it is helpful to have a capacity that carries on that way. The reader who has contemplated cryptography should know that the treatment of hash functions in this book is somewhat not quite the same as a standard cryptography course reading. The puzzle-friendliness property, specifically, is certifiably not a general prerequisite for cryptographic hash functions, however, one that will be valuable for cryptographic forms of money explicitly.

5.7.1 Collision-resistance

The main property for a cryptographic hash function is to make sure that its collision-resistant. A collision happens when two separate input sources produce a similar output. A hash function H (.) is collision-resistant if no one can discover a collision. Formally:

Collision-resistance: A hash function is considerd safe if that it is infeasible to determine two values x and y , such that $x \neq y$, yet $H(x) = H(y)$.

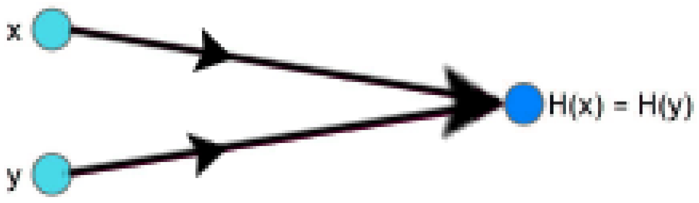
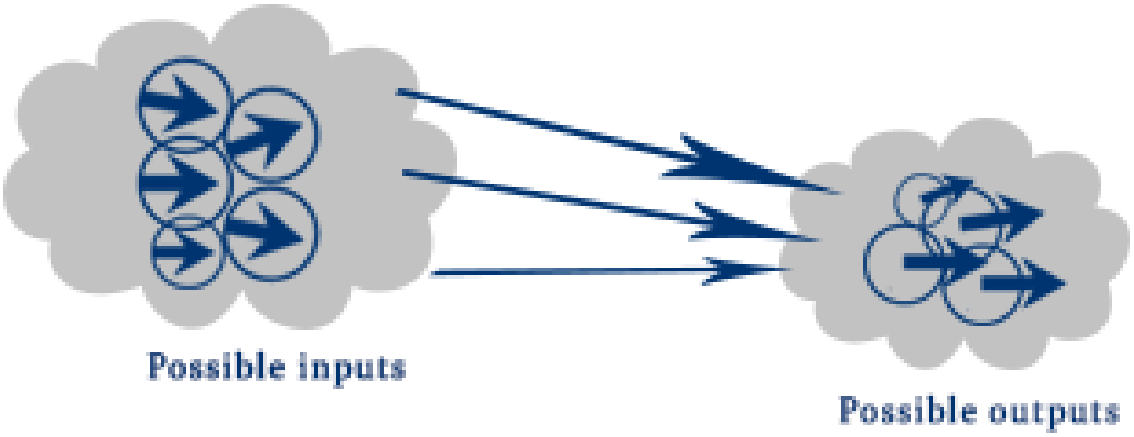


Figure 17- Hash Collision

x** and **y are different input values, yet when input into hash function ***H***, they produce the same output.

Notice that we said no one could discover an impact. However, we didn't state that no collision occurs. All things considered, we know the fact that collision does happen, and we can demonstrate this by a basic tallying contention. The hash function input space contains all strings of different lengths. yet the output space contains as it were strings of a particular settled length. Since the input space is bigger than the output space, in fact, the input space is interminable, while the output space is limited. Input strings must be used to guide the same output string. Truth be told, by the Pigeonhole Principle, there will fundamentally be an exceptionally substantial number of conceivable sources of input that guide to a specific output.



Since the quantity of sources of input surpasses the quantity of outputs, we are ensured that there must be no less than one output to which the hash function maps more than one input.

Presently, to exacerbate things even, we said that it must be difficult to discover a collision. However, there are techniques that are ensured to discover an impact. Consider the accompanying straightforward strategy for finding an impact for a hash function with a 256-bit output estimate: pick 2^{256} + 1 unmistakable qualities, Record each hash and check whether there are two outputs equivalent. Since we have selected more contributions than conceivable outputs, some pairs must have an impact when the hash function is applied[16].

The strategy above is ensured to discover an impact. In any case, if we pick random info sources and register the hash esteems, we will discover a collision with high likelihood well before inspecting 2^{256} + 1inputs. Truth be told, if we haphazardly pick only 2^{130} + 1 inputs, it turns out there is a 99.8% possibility that no less than two of them will impact. The way that we can discover an impact by just analyzing generally the square base of the quantity of conceivable outputs results from a marvel in likelihood known as the birthday oddity. In the homework inquiries toward the finish of this part, we will look at this in more detail.

This collision-detection calculation works for each hash function. Be that as it may, obviously, the issue with it is that this takes, quite a while to do in the most pessimistic scenario, you would have to figure the hash function 2^{256} + 1 times for a hash function with a 256-bit output, and about 2^{128}

times all things taken into account. That is obviously a cosmically substantial number — if a PC ascertains 10,000 hashes for each second, it will take more than one octillion (10^{27}) years to ascertain 2^{128} hashes. For another state of mind about this, we can say that, if each PC at any point made by mankind was registering since the start of the whole universe, up to now, the chances that they would have discovered a collision is still imperceptibly little. So little that it's path not exactly the chances that the Earth will be decimated by a mammoth meteor in the following two seconds.

We have subsequently observed a general yet unreasonable calculation to discover an impact for any hash function. A progressively troublesome inquiry is: is there some other technique that could be utilized on a specific hash function so as to discover an impact? At the end of the day, despite the fact that the conventional collision recognition calculation isn't doable to use, there still might be some other calculation that can proficiently discover an impact for an explicit hash function. Consider, for instance, the accompanying hash function[17].

$$H(x) = x \bmod 2^{256}$$

This function meets our necessities of a hash function as it acknowledges contributions of any length, restores a uniformly sized output (256 bits), and is productively computable. Be that as it may, this capacity likewise has a productive technique for finding a collision. Notice that this capacity just returns the last 256 bits of the input. One collision that would be the qualities 3 and $3 + 2^{256}$. This basic precedent delineates that even though our nonexclusive impact recognition technique isn't usable practically speaking, there are probably some hash functions for which an effective impact location technique exists.

However for other hash functions, we don't have the foggiest idea if such strategies exist. We presume that they are collision safe. Nonetheless, there are no hash functions ended up being collision-resistant. The cryptographic hash functions that we depend on practically speaking are simply functions for which individuals have attempted ridiculously elusive collision and haven't yet succeeded. Now and again, for example, the old MD5 hash function, collision were in the long run found following quite a while of work, driving the capacity to be deplored and eliminated of practical use. Thus we trust that those are collision safe.

Application: Message digests

Now that we recognize what collision-resistance is, now the question is:

What is collision-resistance valuable for?

Here's one application: If we realize that two sources of input **x** and **y** to a collision-resistant hash functions *H* are extraordinary, at that point it's protected to accept that their hashes H(x) and H(y) are unique —— in the event that somebody knew an **x** and **y** that were extraordinary however had a similar hash, that would interrupt our presumption that H is collision safe.

This contention enables us to utilize hash output as a ***message digest***. Consider Secure-Box, a verified online record stockpiling framework that enables clients to transfer documents and guarantee their respectability when they download them. Assume that Hammad transfers an extremely huge document, and needs to most likely check later that the document he downloads is equivalent to the one he transfers. One approach to do that would be to spare the entire huge document locally and straightforwardly contrast it with the record he downloads. While this works, it to a great extent invalidates the point of transferring it in any case; if Hammad needs access to a nearby duplicate of the document to guarantee its respectability, he can simply utilize the local copy specifically.

Collision-free hashes give a rich and productive answer for this issue. Hammad simply needs to keep in mind the hash of the first document. When he later downloads the record from Secure-Box, he registers the hash of the downloaded record and looks at it to the one he put away. If that the hashes are the equivalent, at that point he can presume that the record is, in fact, the one he transferred, yet if that they are unique, at that point Hammad can presume that the document has been altered. Recollecting the hash hence permits him to recognize incidental defilement of the record amid transmission or on Secure-Box's servers yet in addition a purposeful change of the document by the server. Such guarantee conceivably pernicious behavior by different entities is at the core of what cryptography contributes with us.

The hash fills in as a fixed length digest, or unambiguous outline, of a message. This gives us a very proficient approach to recall things we have seen previously and remember them once more. Though the whole document may have been gigabytes long, the hash function is of fixed length, 256-bits for the hash functions in our precedent. This significantly decreases our capacity prerequisite. Later in this section and all through the book, we will understand applications for which it is helpful to utilize a hash as a message digest.

5.7.2 Hiding property

The next property that we need from our hash function is that it is hiding property. The hiding property attests that in case we are given the output of the hash function y = H(x), there is no plausible approach to make sense of what the info, x, was. The issue is that this property cannot be valid in the expressed structure. Think about the accompanying straightforward model: we will complete an examination where we flip a coin. If that the consequence of the coin flip was going, we will report the hash of the string "heads." If the result was tails, we will declare the hash of the string "tails."

We at that point ask somebody, a challenger, who didn't see the coin flip, yet just observed this hash output, to make sense of what the string was that was hashed (we will before long observe why we should need to play diversions like this). They would therefore only figure the hash of the "heads" string and the hash of the "tails" string, and they could see which one they were offered. They can therefore only make sense of what the input was in several ways.

The challenger could think about what the string was because there were just two conceivable estimations of x, and it was simple for the foe to attempt them two simply. To almost certainly accomplish the stowing away property, it should be the situation that there's no estimation of x which is especially likely. That is, x needs to be looked over a set that's, in some sense, very spread out. If that x is browsed such a set, this strategy of attempting a couple of estimations of x that are particularly likely won't work.

The inevitable problem is: Would we be able to achieve the hidden property if the qualities we need are not analyzed by a spread - out set like in our "heads" and "tails?"

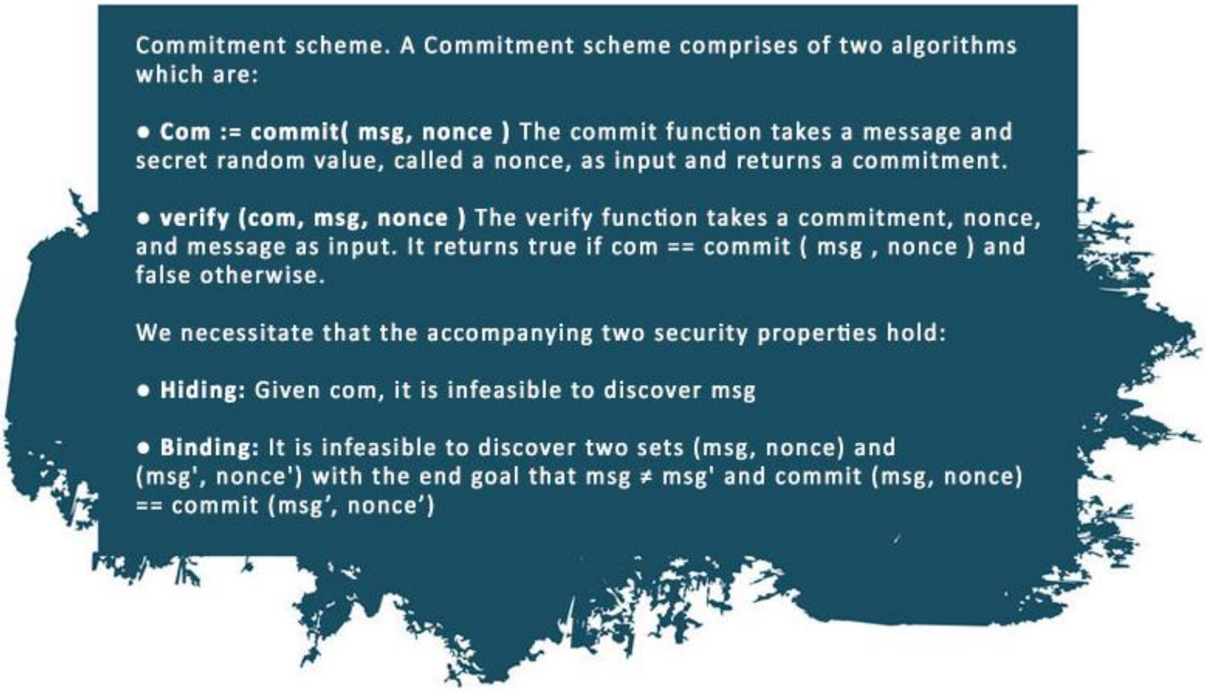
Luckily, the appropriate response is yes! So maybe we can stow away even an input that is not spread out by linking it with another information that is distributed outside. We would now be able to be marginally progressively exact about what we mean by concealing (the dual vertical bar || indicates concatenation).

Hiding.
A hash function .H is hiding, when a value ***r**, which is secret is* chosen from a prospect distribution that has *high min-entropy*, then given .*H(r || x)* it is infeasible to find *x.*

In information-theory, min-entropy is a proportion of how unsurprising a result is, and high min-entropy catches the natural thought that the circulation (i.e., arbitrary variable) is very spread out. What that implies explicitly is that when we test from the distribution, there's no specific esteem that is probably going to happen. In this way, for a solid model, if r is picked consistently from among the majority of the strings that are 256 bits in length, at that point a specific string was picked with likelihood $1/2^{256}$, which is an imperceptibly minor value.

Application: ***Commitments.***

Presently we should take a peek at a use of the hiding property. Specifically, what we need to do is something many refer to as a responsibility. Commitments is a simple and advanced way of taking a value, packing it and putting it outside for everybody to see. You subscribed to what's inside the package when you do that. However, you have not opened it, so despite the fact that you've focused on a value, which remains a mystery value from everybody. Afterward, you can open the package and uncover the value that you committed on before.



Commitment scheme. A Commitment scheme comprises of two algorithms which are:

- **Com := commit(msg, nonce)** The commit function takes a message and secret random value, called a nonce, as input and returns a commitment.

- **verify (com, msg, nonce)** The verify function takes a commitment, nonce, and message as input. It returns true if com == commit (msg , nonce) and false otherwise.

We necessitate that the accompanying two security properties hold:

- **Hiding:** Given com, it is infeasible to discover msg

- **Binding:** It is infeasible to discover two sets (msg, nonce) and (msg', nonce') with the end goal that msg ≠ msg' and commit (msg, nonce) == commit (msg', nonce')

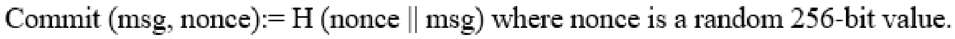
To utilize a commitment scheme, we first need to create a random nonce. We at that point apply the commit function to this nonce together with msg, value being committed on, and we distribute the commitment *com*. This stage is practically equivalent to putting the fixed envelope on the table. At a later point, if that we need to uncover the value that they focused on before, we distribute the random nonce that we used to make this commitment, and the message, msg. Now, anyone can confirm that msg was to be sure the message focused on before. This stage is undifferentiated from opening up the envelope.

Each time you focus on an esteem, it is vital that you pick another arbitrary esteem nonce. In cryptography, the term nonce is utilized to allude to an esteem that must be utilized once.

The two security properties manage that the algorithms really carry on like sealing and unsealing an envelope. Initially, given com, the responsibility, somebody taking a look at the envelope can't make sense of what the message is. Binding is the second property. This guarantees when you commit on what's in the envelope, you can't alter your opinion later. That is, it's infeasible to discover two unique messages, with the end goal that you can commit on one message, and afterward later case that you commit on another.

Now how would we realize that these two properties hold?

Before we can answer this, we have to talk about how we're going to really actualize a commitment scheme. We can do as such utilizing a cryptographic hash function. Consider this commitment plot:



Commit (msg, nonce):= H (nonce || msg) where nonce is a random 256-bit value.

Commiting to a message, we produce an irregular 256-bit nonce. At that point we connect the nonce and the message and return the hash of this value as the commitment. To check, somebody will register this equivalent hash of the nonce they were given linked with the message. What is more, they will check whether that is equivalent to the commitment that they saw.

Look again at the two properties that we expect of our commitment plans. If we replace the commit instantiation and check com as H (nonce/msg), these properties progress to become:

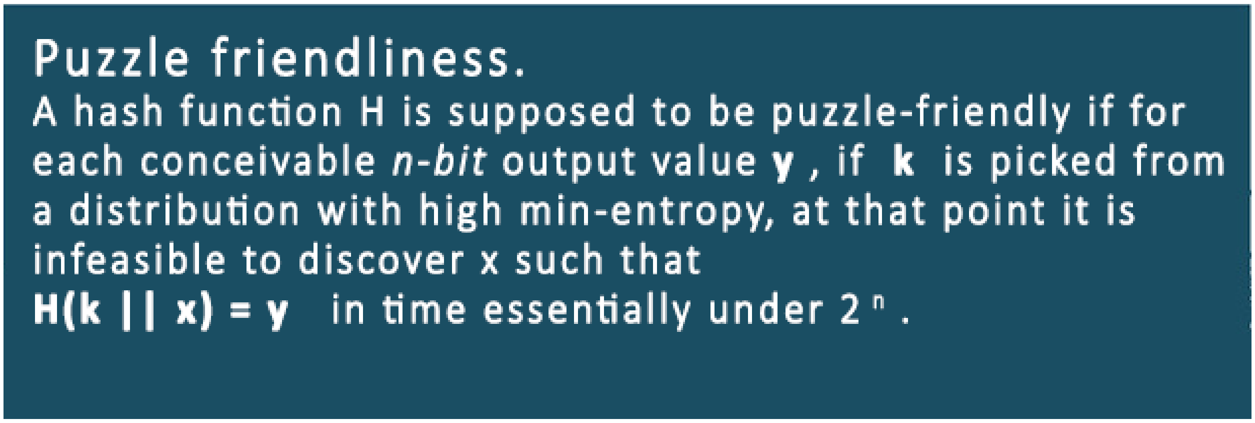
- Hiding: Given H (nonce/msg), it is impossible to detect msg.
- **Binding:** It is infeasible to discover two sets (msg, nonce) and (msg', nonce') with the end goal that msg ≠ msg' & H(nonce || msg) = = H(nonce'|| msg')

The hiding property of commitment is actually the concealing property that we required for our hash capacities. If that key was picked as an arbitrary 256-bit value, at that point, the concealing property says that if that we hash the link of the key and the message, at that point, it is infeasible to recoup the message from the hash output. What's more, incidentally, the coupling property is inferred by the collision-resistant property of the basic hash function. If the function of hash is collision-resistant, at that point, it will be infeasible to discover particular qualities msg and msg' such that H(nonce ‖ msg) = H(nonce' ‖ msg') since such value would without a doubt be a collision.

Accordingly, if H is a hash function that is collision-resistant and hiding, this commitment plan will work, as in it will have the essential security properties.

5.7.3 Puzzle-friendly

The third security property we require from hash capacities is that they are puzzle-friendly. This property is somewhat difficult.

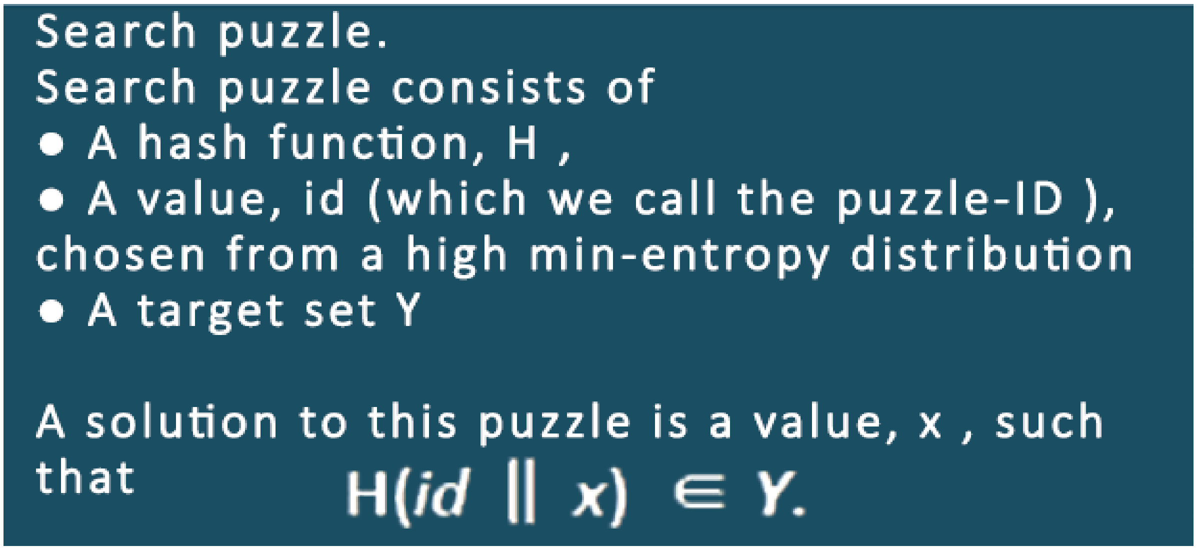


Puzzle friendliness.
A hash function H is supposed to be puzzle-friendly if for each conceivable *n-bit* output value **y** , if **k** is picked from a distribution with high min-entropy, at that point it is infeasible to discover x such that
H(k || x) = y in time essentially under 2^n .

Instinctively, this means if that somebody needs to target on the hash function to turn out to a few specific output value y, that if there is a portion of the input that is picked in a reasonably randomized manner, it is tough to discover another value that hits precisely that target.

Application: Search puzzle.

How about we consider an application that delineates the convenience of this property. In this application, we're going to construct a search puzzle, a mathematical issue which requires looking through an extremely huge space so as to discover the solution. Specifically, search puzzle has no alternate ways. That is, there's no real way to locate a valid solution other than looking through that vast space



Search puzzle.

Search puzzle consists of

- A hash function, H ,
- A value, id (which we call the puzzle-ID), chosen from a high min-entropy distribution
- A target set Y

A solution to this puzzle is a value, x , such that $H(id \| x) \in Y$.

The instinct is this: if H has an n-bit output, at that point it can take any of 2 n values. In order to solve the puzzle it involves an input to be found so that the output falls within the set Y, which is smaller than all output. The size of Y decides how hard the puzzle is. If that Y is the arrangement of all n-bit strings the puzzle is trifling, while if Y has just 1 component the puzzle is maximally hard. The way that the puzzle-id has high min-entropy guarantees that there are no easy routes. Despite what might be expected, if a specific estimation of the ID were likely, at that point somebody could cheat, say by pre-computing an answer for the puzzle with that ID18.

If that an inquiry perplex is puzzle-friendly, this suggests there's no explaining methodology for this puzzle which is much superior to anything simply trying random value of x. Thus, in the event that we need to represent a puzzle that is hard to illuminate, we can do it along these lines as long as we can produce puzzle-IDs in a reasonably random. We're going to utilize this idea later when we talk about Bitcoin mining, which is a kind of computational puzzle.

5.8 SHA-256 Hash function

We've talked about three properties of hash capacities, and one use of each of those. Now how about we talk about a specific hash function. There are lots of hash function in presence, yet this is the one *Bitcoin* utilizes principally, and it's a truly decent one to utilize. It's called **SHA-256**.

Hash function works on input of subjective length. Fortunately, as long as we can manufacture a hash function that takes a shot at fixed-length contributions, there's a nonexclusive strategy to change over it into a hash function that acts on arbitrary-length inputs. It's known as the ***Merkle-Damgard transform***.

SHA-256 is one of various usually utilized hash functions that make utilization of this strategy. In normal phrasing, the hidden fixed-length collision-resistant hash function is known as the ***compression function***. It has been demonstrated that if the basic compression function is collision resistant, at that point the general hash function collision resistant also.

The Merkle-Damgard change is very simple. State the compression function takes input of length m and creates an output of a length n. The input to the hash function, which can be of any size, is partitioned into *blocks* of length *m-n*. The building process fills in as pursues: pass each block together with the output of the past block into the compression function. Notice that the length of the input will at that point be *(m-n) n=m*, which is the information length to the compression function. For the very first block, to which there's no previous block output, we rather utilize an ***Initialization Vector (IV).*** This number is reused for each call to the hash function, and practically speaking, you can simply find it in a measures archive. The last block's output is the outcome that you return.

SHA-256 utilizes a compression function that takes input from 768 bits and generates output from 256 bits. The size of the block is 512 bits. For a graphical representation of how SHA-256 works, see Figure 18.

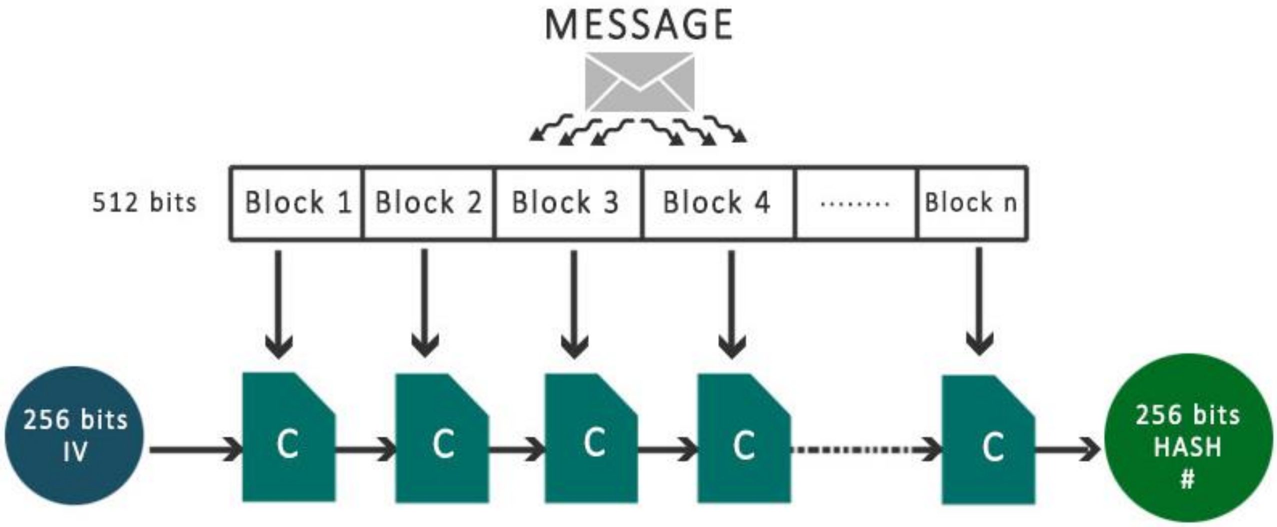


Figure 18 - Hash Function SHA-256

Using the Merkle-Damgard transformation, SHA-256 transforms a fixed-length collision-resistant compression function into a hash function that accepts entirely arbitrary inputs. The input is "padded" to a multiple of 512 bits in length.

We have discussed about ***hash functions***, ***cryptographic hash functions*** with special characteristics, ***applications*** of these properties and a clear and specific hash function that we use in ***Bitcoin***.

Sidebar: Hash functions modeling.

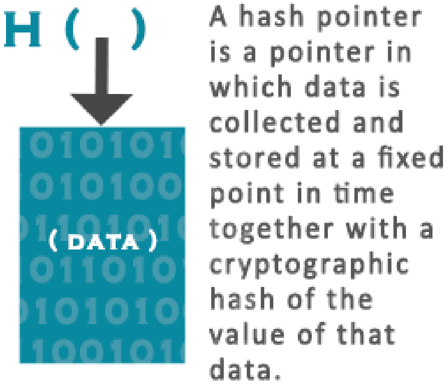
Hash functions are the cryptographic knife of the Swiss Army: In a spectacular variety of applications, they find a place. The flip side of this versatility is that different applications require hash functions to provide security with slightly different properties. It is extremely difficult to determine a list of hash function properties that would lead to proven security across the board. In this text, we have selected three properties that are crucial for Bitcoin and other cryptocurrencies to use hash functions.

Not all of these properties are necessary for any use of hash functions even in this space. For example, in bitcoin mining, puzzle-friendliness is only essential.

Secure system designers often throw the towel and model hash functions that produce an independent random value for each possible input. In cryptography, the use of this "random oracle model" to demonstrate security remains controversial. Regardless of one's position in this debate, it is a valuable intellectual exercise to build secure systems to think about how to reduce the security properties that we want in our applications to the fundamental properties of the underlying primitives.

5.9 Hash pointers and Data structure

Hash Pointer is a useful data structure used by many systems. A hash pointer is clearly a pointer where some information is collected and stored in conjunction with a cryptographic hash. Normal pointer helps to collect the information, but a hash pointer also allows you to validate that the data has not changed.



A hash pointer is a pointer in which data is collected and stored at a fixed point in time together with a cryptographic hash of the value of that data.

Figure 19- Hash pointer

We can logically take a common data structure that utilizes pointers such as a linked list or a binary search tree and implements them with hash pointers instead of as usual.

Blockchain We have created a linked list in Figure 20 using hash pointers. This data structure will be called a block chain. Each block has a pointer as well as data to the prior block in the list, as in a normal linked list with a series of blocks, the prior block pointer is replaced by a hash pointer in a block chain. Each block not only define the value of the previous but it also validate that the value is not altered. We store the list head, which only a regular hash-pointer is pointing to the latest data block.

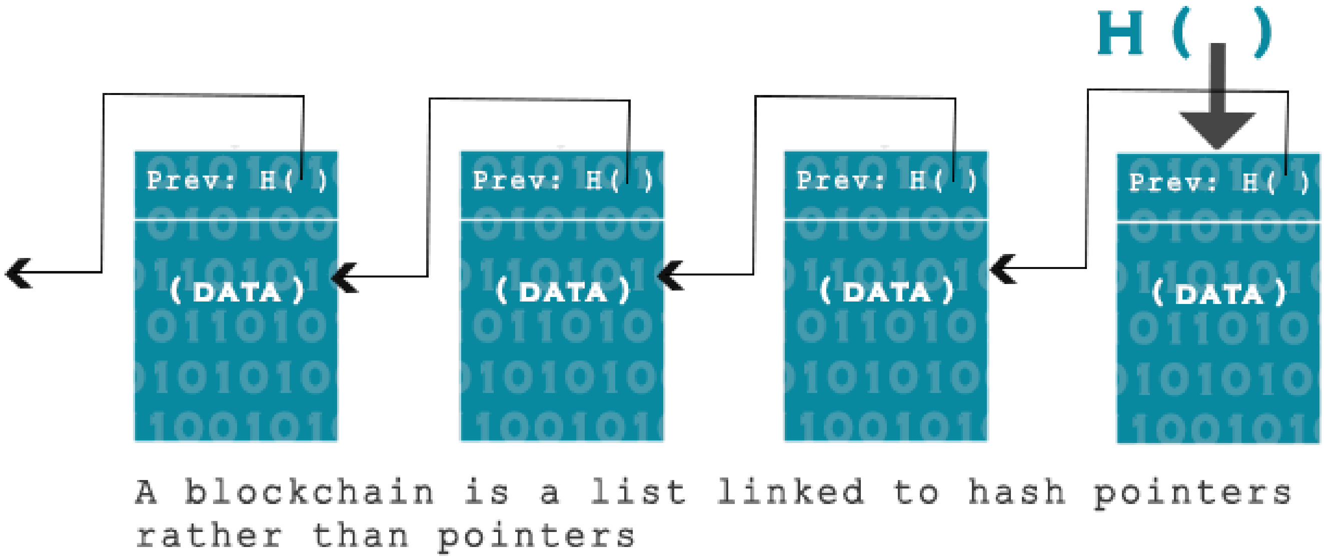


Figure 20 - Linked list Hash pointer

A blockchain's use case is a tamper-evident log. In other words, we want to create a structure that stores a large amount of data and allows us to add data to the log end. However, if someone changes the data in the log earlier, we will detect it.

To understand why this tamper-evident manipulation property is achieved by a block chain, let us suppose what happens if a rival needs to control information in the middle of the chain. Specifically, the objective of the adversary is to do it so that someone who only knows and recalls the hash pointer at the head of the blockchain cannot recognize the manipulation. To achieve this objective, the opponent changes some block k data. Since the data has been changed, it will not match the hash in block k+ 1, which is a hash of the whole block k. Clearly remember that we are statistically assured that the new hash does not match the altered material because the hash function is resistant to collisions. So we detect the inaccuracy between the new data in block k and the hash point in block k+ 1. Naturally, by changing the next block of hash, the rival can continue to try and cover the change. The rival can proceed, yet when he approached the head of the list, this methodology will fail. Specifically, as long as the hash pointer is put away at the highest priority of the list in a region where the rival cannot modify it, without being recognized a rival will be unable to change any block.[19]

You may have seen that the blockchain development is like the Merkle-Damgard development that we found in the past area. They are very comparable, and a similar security contention applies to them two.

Tamper-evident Log

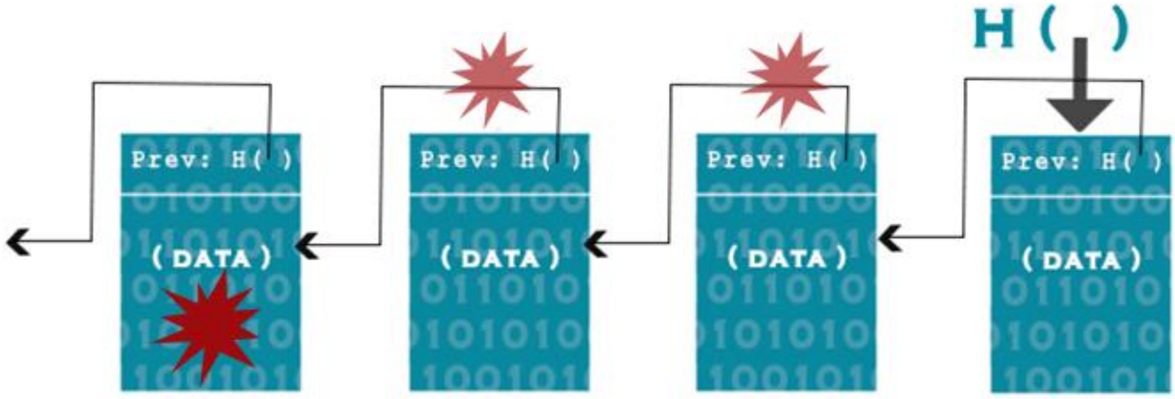


Figure 21- Tamper-Evident Log

In blockchain if any adversary try to modify the data anywhere, it will affect the hash pointer in that block showing as incorrect. If head of the list are also stored then even if the adversary alters every pointer to be constant with the modified data the header pointer will be still invalid and thus we can detect the tampered data easily[20].

5.10 Merkle-Tree

A binary tree is another valuable data structure that we can construct with hash pointers. Following its inventor Ralph Merkle, a binary tree with hash points is recognized as a Merkle tree. Assume we do have some data-containing blocks. These blocks are made up of our tree leaves. We organize these data blocks in pairs and then start building a data structure for each pair with two hash points, one each for these blocks. All these data structures create the tree's next level. We organize them into two groups and create a new data structure for each pair that includes each hash. We continue doing this until we achieve a single block, the tree's root

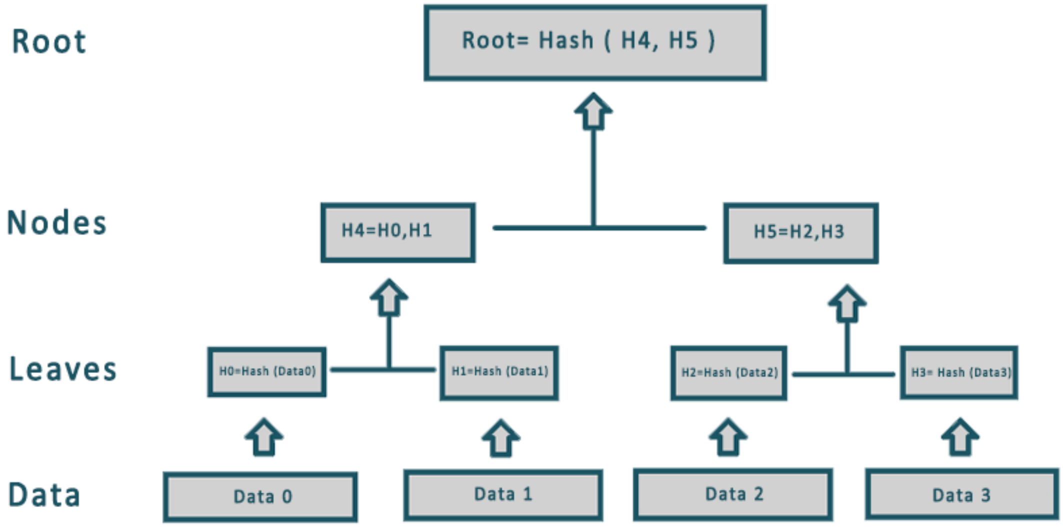


Figure 22- Merkle Tree

As before, we remember the hash pointer at the tree's head. Now we have the unique ability to navigate the hash pointers to any point in the list. This helps us make sure that the data has not been manipulated because, just as we've seen with block chain, when an opponent tampers with plenty of data block at the bottom of the tree, the hash pointer will not correspond to one level, and even if he continues to manipulate the block, the change will soon spread to the top of the tree, where he might not be able to manipulate wit. So once again, any obvious attempt to manipulate any data is identified by remembering the top hash pointer.[21]

Proof of membership:

One more decent component of Merkle trees is that, not normal for the block chain that we manufactured previously, it permits a brief proof of membership. When somebody needs to demonstrate that a specific information block is an individual from the Merkle Tree. Obviously, we recall only the root. At that point they have to appear us this information block, and the blocks on the way from the information block to the root. We can disregard the rest, as the blocks on this way are sufficient to enable us to check the hashes as far as possible up to the foundation of the tree. See figure 23 for a graphical portrayal of how this functions.

On the off chance that there are ***n*** nodes in the tree, just about *log (n)* items should be appeared. What's more, since each progression just requires figuring the hash of the block child, it takes about *log (n)* time for us to check it. Thus regardless of whether the Merkle tree contains an exceptionally vast number of blocks, we can in any case prove membership in a generally brief time. Check subsequently keeps running in reality that is logarithmic in the quantity of nodes in the tree.

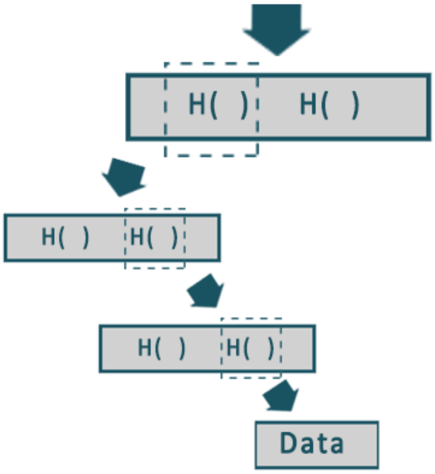


Figure 23- Proof of member ship

A sorted Merkle tree is only a Merkle tree where we take down the blocks and sort them with some other ordering mechanism. This might be alphabetical order, numerical, lexicographical or otherwise some other approved order.

Proof of non-membership:

With a sorted Merkle tree, it becomes conceivable to confirm non-membership in a logarithmic reality. That is, we can demonstrate that a specific block isn't in the Merkle tree. What's more, the manner in which we do that is basically by demonstrating a way to the thing that is simply before where the thing in question would be and demonstrating the way to the thing that is soon after where it would be. In the event that these two things are back to back in the tree, at that point this fills in as a proof that the thing being referred to is excluded. For on the off chance that it was incorporated, it would should be between the two things appeared, there is no space between them as they are consecutive.

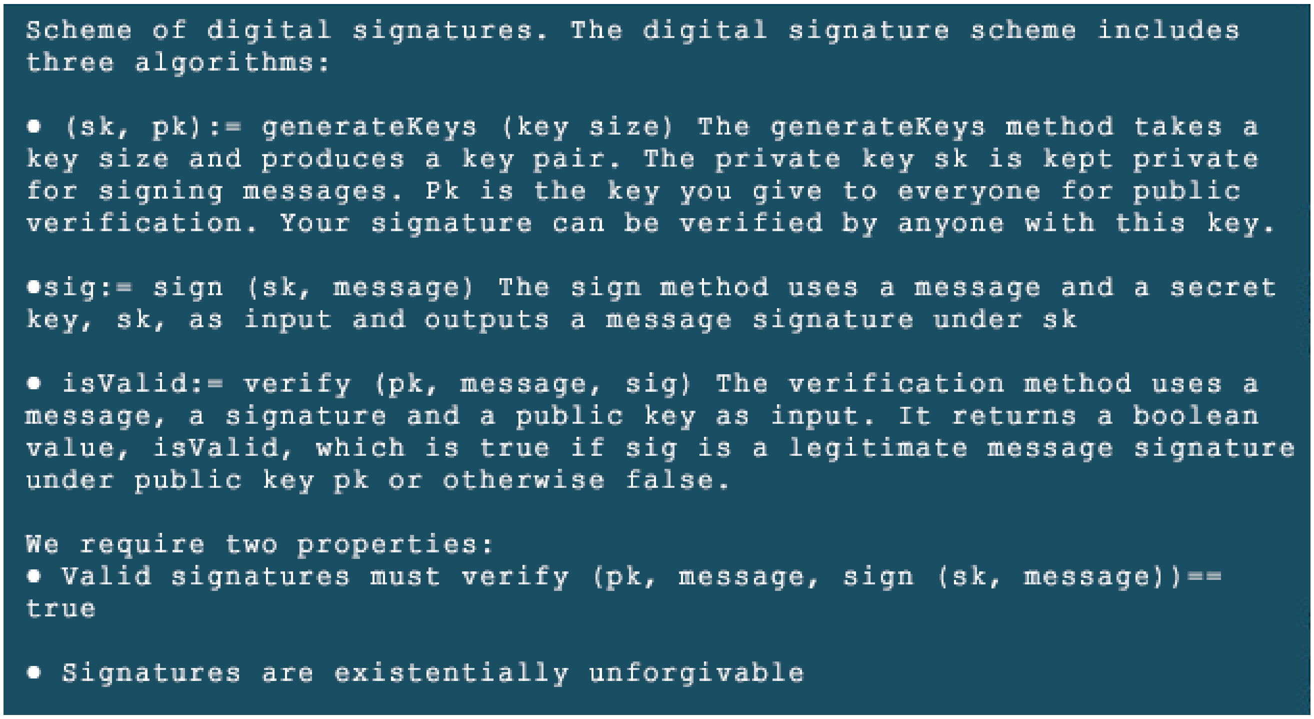
We have examined utilizing hash pointers in connected records and double trees, yet for the most part, it turns out that we can utilize hash pointers in any pointer-based information structure as long as the information structure does not have cycles. On the off chance that there are cycles in the information structure, at that point we will not most likely make all the hashes coordinate. Looking at the situation objectively, in a non-cyclic information structure, we can begin close to the leaves, or lose to the things that do not have any pointers leaving them, computing the hashes, and at that point work our way back toward the start. Be that as it may, in a structure with cycles, there is no closure we can begin with and compete once again from[22].

5.11 Digital signature

This is the second cryptographic primitive, alongside hash functions, that we need as building blocks for the cryptocurrency. We want two properties from digital signature.

1. No one but you can make your signature, yet any individual who sees it can confirm that it's legitimate.
2. We need the signature to be tied to a specific document so the signature can't be utilized to demonstrate your understanding or underwriting of an alternate document.

For transcribed signatures, this last mentioned property is closely resembling guaranteeing that someone can't take your signature and clip it off one archive and paste it onto the base of another.



Scheme of digital signatures. The digital signature scheme includes three algorithms:

● (sk, pk):= generateKeys (key size) The generateKeys method takes a key size and produces a key pair. The private key sk is kept private for signing messages. Pk is the key you give to everyone for public verification. Your signature can be verified by anyone with this key.

●sig:= sign (sk, message) The sign method uses a message and a secret key, sk, as input and outputs a message signature under sk

● isValid:= verify (pk, message, sig) The verification method uses a message, a signature and a public key as input. It returns a boolean value, isValid, which is true if sig is a legitimate message signature under public key pk or otherwise false.

We require two properties:
● Valid signatures must verify (pk, message, sign (sk, message))== true

● Signatures are existentially unforgivable

GenerateKeys and sign can be randomized calculations or algorithms. Undoubtedly, generateKeys would be wise to be randomized in light of the fact that it should produce distinctive keys for various individuals. Verifying on the other hand, will constantly be deterministic.

Now look at the two properties that we expect of a digital signature in more depth.

The first property is straight clear — that valid signature must confirm. If someone sign a message with **sk,** its mystery key, and somebody later endeavors to approve that signature over that equivalent message utilizing my public key, **pk**, the signature must approve accurately. This property is a fundamental prerequisite for signature to be useful[23].

Unforgeability. The second requirement is that it's computationally infeasible to fake signature. That is, an opponent who knows your public key and gets the chance to see your signatures on some other messages can't produce your signature on some message for which he has not seen your signature. This unforgeability property is commonly formalized regarding a diversion that we play with an enemy. The utilization of amusements is very normal in cryptographic security proofs.

In the unforgeability diversion, there is a foe who asserts that he can produce signatures and a challenger that will test this case. The main thing we do is we use generateKeys to create a secret signing key and a comparing public key. We give that secret key to the challenger, and we give the general public key to both the foe and the challenger. So the foe just knows data that is public, and his main goal is to try to fake a message. The challenger knows the secret key. So he can make signatures.[24]

Practical Concerns. There are various things that we can do to transform the algorithmic idea into a digital signature component that can be actualized practically. For instance, numerous signature algorithms are randomized (specifically the one utilized in Bitcoin) and a good source of randomness is required by us. This is important because a bad randomness will lead our secure algorithm to an insecure algorithm.

The size of the message is another practical concern. In practice, the size of the message you can sign is limited because real schemes will work on bit strings of limited length. This limitation is easy to overcome: sign your message hash instead of the message. If we are using a 256-bit output cryptographic hash feature, then a message of any length can be signed effectively until our signature scheme can sign 256-bit messages. Earlier as discussed, it is safe to use the message hash as a message digest because we know that the hash function is collision-resistant.

ECDSA Algorithm

Elliptic Curve Digital Signature Algorithm is a unique digital signature scheme which is used by Bitcoin. ECDSA is an update of the earlier DSA algorithm adapted for the use of elliptic curves. It is a U.S. government standard. These algorithms have been analyzed cryptographically extensively over the years and are generally considered safe.

Also, Bitcoin uses ECDSA over the standard elliptic curve "secp256k1" which is projected to provide security of 128 bits, i.e. this algorithm is as hard to break as performing 2^{128} symmetric-key cryptographic operations, for example, invoking a hash function. Although this curve has been issued as a standard, it is rare to be used outside of Bitcoin, with other applications using ECDSA (for example, in secure web browsing key exchange on TLS) commonly by using the "secp256r1" curve. This is only a peculiar bitcoin, as Satoshi has chosen it in the early system specification and is hard to change now.

It could be helpful to know the size of the different quantities:

The public key, uncompressed	***512 bits***
The public key, compressed	***257 bits***
Private key	***256 bits***
Message to be signed	***256 bits***
Signature	***512 bits***

Table 2: Size of different quantities

5.12 Public Keys

This trick goes with digital signatures. The aim is to use a public key from a digital signature scheme and match it with an identity of an actor or person in an individual system. If under the public key (*pk*), you notices a message with a signature which correctly verifies it, you can think about this as *pk* says the message. You can consider a public key like a party or actor in a system that can generate a statement simply by signing those statements. The public key is an identity from this viewpoint. To speak for identity *pk*, people must know the appropriate secret key, *sk*.

One result of treating public keys as identities is that you will be able to create a new identity everywhere you want which means you just create a fresh new key pair, *sk,* and *pk*, through our digital signature system using **generateKeys** operation. You can use *pk,* the fresh public identity and *sk* is the secret-key that is known by you only, and you can use for the identity *pk* as well. Since public keys are outsized, you can use *pk* hash as your identity. In that case, then to confirm and verify that your identity sends a message you have to check:

1- That *pk* actually hashes to your identity.
2- The message authenticates under public key *pk*.

In addition, by default your public key *pk* will essentially look random and no one can uncover your identity in the real world by analyzing your *pk*. You can actually create a fresh, random identity that can only be controlled by you.

Decentralized identity management. This leads to the concept of decentralized management of identity. Register as a user by yourself instead of having a central authority to register you in your system. There is no need for you to be issued a username or inform someone that you will use a specific name. You may generate an identity at any time if you want a new one, and you can create as many as you desire. There is no problem if you want to be known by six different identities. You can make a new identity, use it for a short while and then discard it if you want to. With decentralized identity management, all of those things are possible and that is what Bitcoin does with identity. In Bitcoin jargon, these identities are called ***addresses***. In Bitcoin and Cryptocurrencies, you will often hear the word "address," and that is a public key hash. It is an identity made up, as part of this decentralized identity management scheme.

Sidebar

It might seem counterintuitive that you can generate an identity without a centralized authority. After all, if someone else is lucky enough and he generates quite the same key as you, can't your bitcoins get stolen by them?

The solution to this is that the possibility if somebody else will produce the same 256-bit key as you are so small then we must not concern about it. We are sure that it won’t happen for all purposes.

In contrast to the intuition of beginners, probabilistic systems are unpredictable and difficult to understand, often the opposite is true. The statistical theory enables us to quantify precisely the probability of events we are interested in and to make confident claims about the behavior of such systems. Therefore, when generating keys, it is significant to use a reliable source of randomness to make sure that the practical guarantees should match with the theoretical.

However, there is a subtlety, the probabilistic guarantee is only true if keys are produced randomly. The random generation in real systems is often a weak point. If two users' uses the same random source or use predictable randomness, theoretical guarantees are no longer applicable.

At first, decentralized identity management seems to lead to a high level of anonymity and confidentiality. After all, without telling anybody your real-world identity, you can create a random identity by yourself. However, it is not that easy to grasp. The identity you create makes some statements over time. People see these reports, and they know that whoever possesses this identity has carried out certain actions. Therefore they can begin connecting them by using this series of actions to conclude information from your true identity. An observer can link these things over time and conclude by making inferences such as, "Danish, this person is acting a lot like Hammad. Maybe this person is Hammad."

In other words, you do not have to specifically register or reveal your real-world identity with Bitcoin, but your behavior's pattern could be identifiable itself.

5.13 Crypto-currency (Simplified)

We will discuss about the two very simple cryptocurrencies in this section.

GoofyCoin

The first is Goofy-Coin which is the simplest of crypto-currency we can think off. GoofyCoin has only two rules. In the first rule a nominated entity, Goofy, whenever he wants he can create new coin and those coins that have been newly created owned by him.

Goofy creates an ID to create a coin **uniqueCoinID** which he has never created before and constructs the string **"CreateCoin [uniqueCoinID]."** The digital signature of the string is then computed with his stealthy signing key. The string along with the signature of Goofy is a coin. Anybody can examine that the coin contains a valid CreateCoin statement signed by Goofy and therefore it's a valid coin. GoofyCoin's second rule is if anyone who owns a coin can transfer it to anyone else. It is not simple to transfer of a coin by simply transfer the coin data structure to the recipient; it is actually done by using cryptographic operations.

For example, if Goofy wants to send a coin that he created to Hiba. For this purpose, a new statement he will create that says, "Pay this to Hiba" where the word "this" is a hash pointer and that refers to the coin in question. Moreover, as we have seen before, identities are only public keys, so "Hiba" refers to the public key of Hiba. Goofy finally signs the string which represents the statement. Since Goofy is the owner of the coin so any of the transaction that spends the coin must have to be signed by him. When the data structure of Goofy's signed transaction exists, Hiba is the owner of that coin. She can show to anyone that she is the owner of the coin because she can show Goofy's valid signature to show the data structure. Also, it points to a valid coin which Goofy owned. Thus the system is self-evident in the validity and ownership of coins.

Once the coin has been owned by Hiba, then she can spend it. For this purpose, a new statement she will create that says, "Pay this coin to Safia's public key" where the word "this" is a hash pointer to that coin which she owned. Moreover, that statement will be signed by Hiba. Moreover, when this coin is presented, anyone can check and verify that the owner of this coin is Safia. They can go back following the hash pointer's chain to the creation of the coin and can verify that in each step the legitimate owner signed a statement that says "pay this coin to [new owner]."

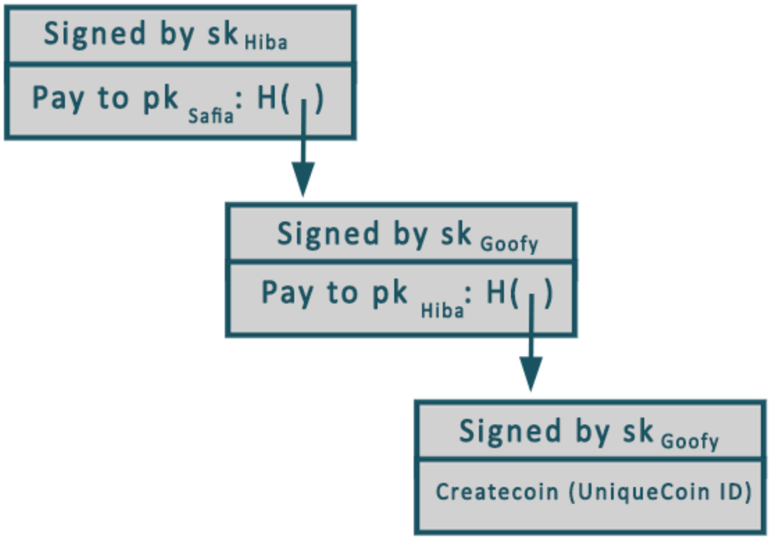


Figure 24 - Goofy Coin

It shows that coin has been created at the bottom and spent twice

In short, the GoofyCoin rules are:

- New coins can be created simply by signing a statement that Goofy is creating a new coin with a unique coin ID.
- Anyone who owns a coin, he or she can pass it on by simply signing a statement that says, “Pass on this coin to X” (where the public key is specified by X)
- Everyone can check the validity of the coin by following Goofy’s chain of hash pointers to verify all the signatures.

Goofy-Coin has a fundamental security issue. Let us suppose that Hiba passed her coin with a signed statement to Safia, but she did not tell to anybody. Another signed statement she can create that pays the same coin to Hammad. It seems to Hammad that the transaction is perfectly valid and now he owns that coin. Both Safia and Hammad would have valid claims that they own this coin. This is known as a double-spending attack. The same coin is spent by Hiba twice. We know that coins should not work this way intuitively. We know that the coins cannot work in this way.

Double-spending attacks are in fact one of the main issues to solve for any cryptocurrency. The double-spending attack is not solved by GoofyCoin and is thus unsafe. GoofyCoin is simple, and its transference mechanism is very similar to Bitcoin. In fact, but because it is insecure, it will not be cut off as a cryptocurrency.

ScroogeCoin

Another cryptocurrency is designed to resolve the double-spending attack, which is the ScroogeCoin. ScroogeCoin is made from GoofyCoin, but in data structures, it is a bit more complicated.

The first important idea for a designated entity called Scrooge is to publish an append-only ledger that contains all the history of the transactions. Any data written in that ledger will be kept forever by the append-only property. When the ledger is genuinely append-only, we can use it to protect against double spending by requiring all transactions to be written before accepting them. Thus this will make it visible to the public if the coins have already been sent to another owner.

Scrooge can build a blockchain which he can sign digitally for the implementation of this append-only feature. There is a series of data blocks and each containing a single transaction (We would actually put several transactions in the similar block as Bitcoin in practice as an optimization.) Each block has the transaction ID, the contents of the transaction and a hash pointer to the previous block. The final hash pointer, which connects all data in the entire structure, will be digitally signed by Scrooge and published along with the blockchain.

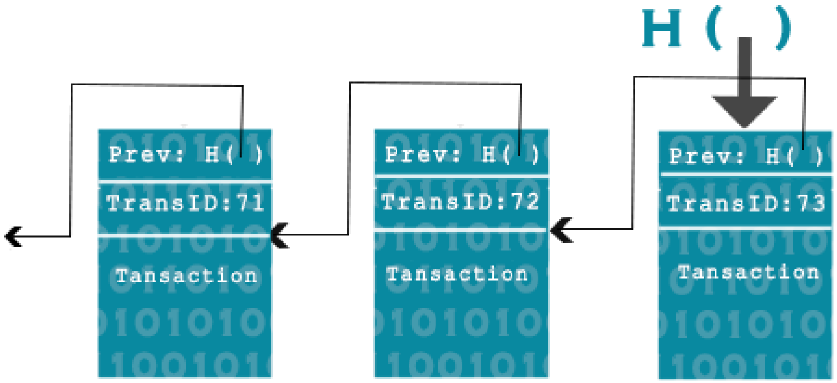


Figure 25- ScroogeCoin Block

A transaction in ScroogeCoin counts only when it is signed by Scrooge in the blockchain. Anyone can check that Scrooge approves a transaction by checking the signature of Scrooge in the block it appears. Scrooge will ensure that he will not approve that transaction that attempts to double-spend a coin already been spent.

Why do we need a hash pointer blockchain, besides, to have each block signed by Scrooge?

The append-only property is ensured by this. If the Scrooge attempts to add, remove, or change an existing transaction, all the following blocks will be affected because of the hash pointer. As long as someone monitors Scrooge's latest hash point, the change will be clear and easy to grasp. You would have to keep track of every Scrooge signature ever issued in a system where Scrooge signed blocks individually. A block chain makes it very convenient for two people to check whether they have followed the same history of Scrooge transactions.

There are two types of transactions in ScroogeCoin. The first type is CreateCoins, which is just the same as a Goofy operation that made a new coin in GoophyCoin. We will extend the semantics with ScroogeCoin a bit so that several coins can be created in a single transaction.

Type :CreateCoins Trans ID : 73

Numbers	Value	Recipient
0	3.2	0 x…. **[CoinID 73(0)]**
1	1.4	0 x…. **[CoinID 73(1)]**
2	7.1	0 x…. **[CoinID 73(2)]**

Table 4: CreateCoins transaction

This transaction CreateCoins creates several coins. Within the transaction, each coin has a serial number. Every coin also has a value; some ScroogeCoins are worth. So there is a receiver for every coin, which is a public key when the coin is created. Therefore CreateCoins creates a series of new coins with different values and assigns them to the first owners. We refer to coins by coined. The coined is a combination of a transaction ID and a serial number of the coin.

PayCoin is the second kind of transaction. It consumes certain coins and destroys them, thus creating new coins of the same total value. The new coin may belong to different people (public key). Everyone that pays in a coin must sign this transaction. So when you own one of the coins that will be consumed, you need to sign the transaction digitally to say that you are actually okay with spending the coin.

ScroogeCoin rules say that if four things are correct, the PayCoins transaction is valid. The rules are:

- The coins used are valid, i.e., in previous transactions, they were created.
- If the coins consumed in some previous transactions were not already consumed, then it is not a double-spend.
- The overall value of the resulting coins is equal to the total value of the entered coins, i.e., Scrooge can only create a new value.

- The transaction shall be validly signed by the owners of all the coins consumed.

<table>
<tr><th colspan="3">Type :PayCoins Trans ID : 73</th></tr>
<tr><th colspan="3">Coin ID Consumed: 68(1),42(0),72(3)</th></tr>
<tr><td>Numbers</td><td>Value</td><td>Recipient</td></tr>
<tr><td>0</td><td>3.2</td><td>0 x…….</td></tr>
<tr><td>1</td><td>1.4</td><td>0 x…….</td></tr>
<tr><td>2</td><td>7.1</td><td>0 x…….</td></tr>
<tr><th colspan="3">Signatures</th></tr>
</table>

Table 5 : Transaction of Paycoins

This PayCoins transaction will be valid, and Scrooge will accept it only if all of the conditions are fulfilled. He will write it in the history of the block chain then everyone can see that that transaction has occurred. Only then can the participants accept that the transaction has actually taken place. Until published, a double - spending transaction may be pre-empted, even if the first three conditions apply otherwise.

Coins are immutable in this system; they are never changed, combined or subdivided. Every coin is once created in one transaction and then consumed in another one. However, we can have the same effect as subdividing or combining coins through transactions. For Example, Hiba creates a new transaction to subdivide the coin that consumes that one coin, which then produces two new coins that have the same total value. She could be assigned those two new coins again. Even if the coins of this system are immutable, they do have the flexibility of a system with no immutable coins.

With ScroogeCoin, we come to the core issue now. The functioning of ScroogeCoin is to show which coins are valid. This prevents double-spending, as everyone can look into the blockchain and see that all transactions are valid and each coin is taken once. However, the The problem is that Scrooge has too much influence. He cannot create fake transactions since he cannot forge the signatures of other people. However, he can stop endorsing certain users' transactions, by denying them services and by making their coins un-spendable. If Scrooge is greedy, he may refuse to publish transactions if he does not receive any mandatory transaction fee. If Scrooge wants he can create as many new coins for himself. Alternatively, Scrooge may get bored with the whole system and may not update the blockchain completely.

Centralization is the problem here. Users might not be pleased with this system. Cryptocurrencies largely failed to take off in practice, with the central authority. There are some reasons for this, but in hindsight, it seems hard to get people to accept a centralized authority cryptocurrency. In order to accomplish this, we need to figure out how everyone can agree on a single block chain published in the history of transactions. They all need to agree which transactions are valid and which transactions actually have occurred. They must also be able to decentrally assign IDs to things. Finally, it is necessary to control the minting of new coins in a decentralized way. We can build a currency, like ScroogeCoin, but without the centralized party, if all of these problems are solved. This would actually be a very similar system to Bitcoin.

Chapter 6 – Consensus Models

6.1 Introduction

An essential part of Blockchain technology is that a greater part of nodes must check and confirm transactions within a block before the block can be confirmed and recorded. So, it is not possible for anyone to change the ledger, everybody can examine it, and it tends to be trusted. This is known as “reaching consensus”.

Another part of consensus is figuring out which mining node has the right to the public the following block in the linear blockchain. There are few models for accomplishing this. The idea for blockchain technology was under the presumption that, generally, the user would not know one another and, accordingly, have a common doubt in each other. Blockchain innovation is likewise proposed to be self-policing and distributed, which guarantees a central authority is not required, and that will help in reducing the transaction cost.

These model allow users to carry transactions among themselves and to accomplish self-policing state. However, a wide range of blockchain applications has been generated or proposed for the future, which has fluctuating dimensions of security and trust. This incorporates the idea of permissioned ledgers, where some dimension of trust is probably going to exist.
Following are significant Consensus Models that exist today practically. Software application automatically handles these models without manual mediation.

6.2 Proof of work consensus model (PoW)

The PoW model, the most commonly used consensus model also used by bitcoin platform it requires that for the mining nodes to post blocks in the blockchain. The processing resources should be expanded in order to solve the puzzle. The precise solution support as the proof that they have performed the work necessary to publish the block. This puzzle-solving process cost money in terms of electricity and time.

When a user finishes the work, they forward their block to alternate nodes on the network. Alternate nodes at that point verify that the work has been finished and that the transaction content and the block are valid. After validation, the nodes add the block to their duplicate ledger and convey the block all through the network.

The (PoW) model is appropriate for permissionless ledgers, which enable anybody to contribute information to the ledger and for everybody possessing the ledgers to have the duplicates copies. As anybody can contribute, there is a universal distrust among the users. The (PoW) model guarantees that every user generally has a similar probability of solving the puzzle, thereby avoiding specific users from having control over it.

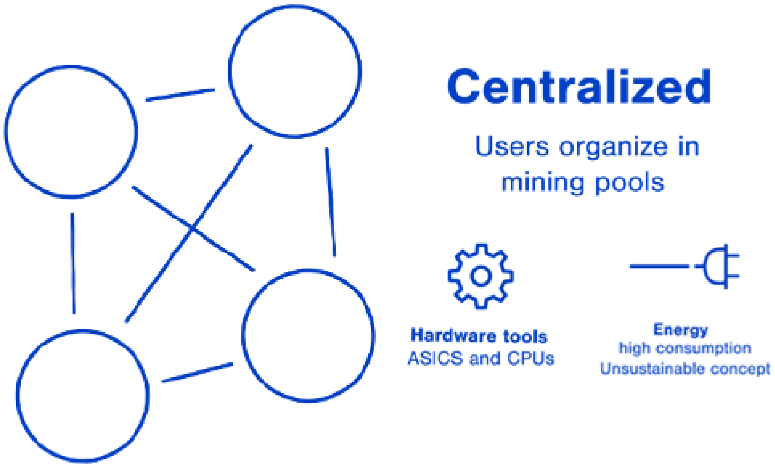


Figure 26- Proof of work:[25]

6.3 Proof of Stake Consensus model (PoS)

The concept of Stake Proof (PoS) states that a person may mine or validate block transactions depending on how many cryptocurrencies they hold. This implies that the more a miner owns Digital currency, the more he or she has mining authority.

There are various methods for basically executing Proof of Stake. A few techniques incorporate randomly picking users proportionate to their amount of money, multi-round voting, and coin aging where users with more old digital currency are given significance over as compared to others. With coin aging, when a user made a block, the age of their digital currency resets to "0" to prevent users from gaining control over decision power.

PoS consensus model, there is no need for resources computation such as electricity, time, processing power) as compare to proof of work. PoS uses fewer resources, some blockchain have chosen to forego a reward for new square creation. These frameworks are structured with the goal that all the cryptographic money is as of now circulated among users rather than new coins being produced at a steady pace [26].

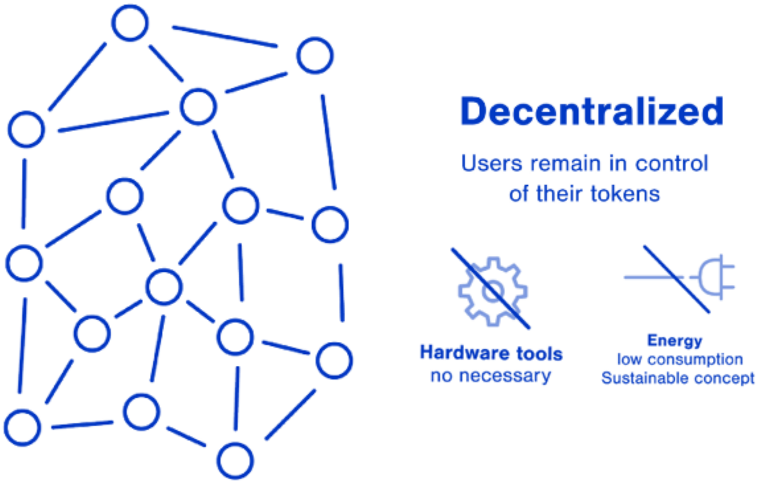


Figure 27- Proof of Stake

PoW model is appropriate for conditions where there are large amounts of common distrust, for example, in permissionless ledgers.

6.4 Proof of Authority Consensus model (PoA)

Proof of Authority (PoA) gives the facility to approve and distribute new blocks to the blockchain for approved users who are known as validators. Different consensus model such as Proof of Stake and Proof of Work, a user's character must be identified and verified. That's critical because identity is the only confirmation of a user's ability to add new blocks to the chain. Comparing PoW with PoA, Proof of Authority is a lot quicker model for handling new blocks, as there is no requirement for long and resources intensive processing. PoA could be utilized in both permissionless and permissioned ledgers.

This model may appear more familiar to clients who have practice and understanding about databases in which just precise users may alter or add information to a database. In this way, it might be the most relevant for some uses of blockchain technology in the public-sectors, as it can be adjusted to reflect the complexity of government decision – making and review processes.

6.5 The round robin Consensus model

There is some degree of trust between mining nodes in some blockchain systems. In this case, to determine which user adds the next block to the chain, there is no need for a complex consensus mechanism. This consensus model is frequently used for private blockchain and is referred to as round robin, where nodes create blocks by their turns. When mining nodes are not generally available on their turns, these systems might include a random element to allow available nodes to compile blocks so that unavailable nodes do not cause block production to stop. This model guarantees that the majority of blocks are not created by any specific node, it benefits from a simple approach, and it lacks cryptographic puzzles and low power demands.

Deplorably, due to the requirement for some dimension of trust among nodes, round robin does not function admirably in the permissionless open systems utilized by most blockchain based cryptocurrencies because mischievous nodes can persistently add extra nodes to build the chances of subverting the system.

6.5 List of further Consensus Model

- Proof-of-Activity
- Proof-of-Weight
- Proof-of-Importance
- Proof-of-Burn
- Proof-of-Capacity
- Delegated Proof-of-Stake
- Leased Proof-Of-Stake
- Proof of Elapsed Time
- Simplified Byzantine Fault Tolerance
- Delegated Byzantine Fault Tolerance
- Practical Byzantine Fault Tolerance
- Directed Acyclic Graphs

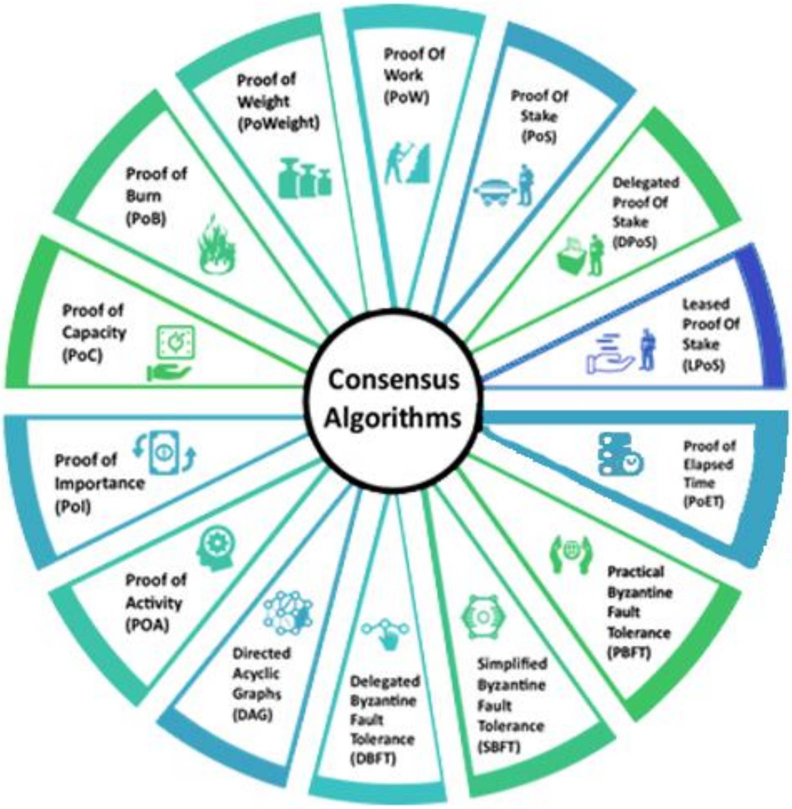

Consensus Algorithms
Proof Of Work (PoW)
Proof Of Stake (PoS)
Delegated Proof Of Stake (DPoS)
Leased Proof Of Stake (LPoS)
Proof of Elapsed Time (PoET)
Practical Byzantine Fault Tolerance (PBFT)
Simplified Byzantine Fault Tolerance (SBFT)
Delegated Byzantine Fault Tolerance (DBFT)
Directed Acyclic Graphs (DAG)
Proof of Activity (POA)
Proof of Importance (PoI)
Proof of Capacity (PoC)
Proof of Burn (PoB)
Proof of Weight (PoWeight)

Figure 28- List of other consensus model[27]*.*

Chapter 7 – DLT/Blockchain Case study in different field

7.1 Registration of birth and death.

The Indian Civil Registration System is subject to the 1969 Birth and Death Registration Act. The important aspects of the registration of birth and death in India are:

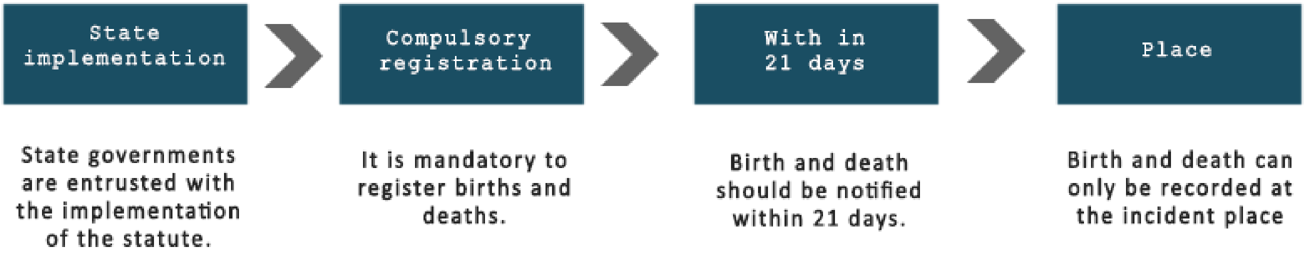

State implementation
Compulsory registration
With in 21 days
Place
State governments are entrusted with the implementation of the statute.
It is mandatory to register births and deaths.
Birth and death should be notified within 21 days.
Birth and death can only be recorded at the incident place

Figure 29-Indian process of birth and death registration

The number of civil enrollments in India has become in the course of the most recent two decades. Birth enlistment has gone up from 58% to 88.8% in the middle of 2001 and 2014; likewise, registered incident of death have gone up from 54% to 74.3% in the equivalent period[28]. While these are certain pointers, numerous fundamental issues still exist with respect to precision, consistency, fulfillment and timeliness. Some of the keys related with these issues are given beneath:

Incomplete coverage

The relatively low use of registration certificates, the use of additional documents rather than birth certificates and a general sense of apathy and lack of knowledge and understanding among citizens inhibit the spread of the registration of birth and death, resulting in incomplete coverage. As of 2014, the death rate in some of the lowest performing countries, such as Arunachal Pradesh (28.7 percent), Bihar (24.1 percent) and Nagaland (27.3 percent), ranged from 20 to 30 percent.[29]

Interdepartmental storehouses

While a couple of states have moved to computerized stages for registration, numerous still preserve manual records. The techniques for enrollment, information gathering and confirmation change enormously.

Precision of enlisted data

Issues identified with validness of birth endorsements lead to an assortment of issues, for example, issues with allocation of resources through public dispersion activities and alteration of indispensable data for transaction or value-based purposes.

Disparities in large scale level pointers

A mixture of inaccurate and fragmented data makes errors in deduced macro-level indicators, for example, baby death rates and statistic proportions. These markers normally structure the information point for pertinent arrangement advancement and decision making. Their distortion impacts the designation of assets for future activities and arrangement improvement.

A couple of the issues in birth and passing enlistment can be settled by blockchain-based digitization. Blockchain gives the benefit of permanence, prompting genuine provenance and conclusiveness, in this manner making it a stage of decision for the digitization of birth and demise records. The figure beneath delineates the key points of interest in utilizing blockchain-based answers for the location the issues related to birth and demise enrollment.

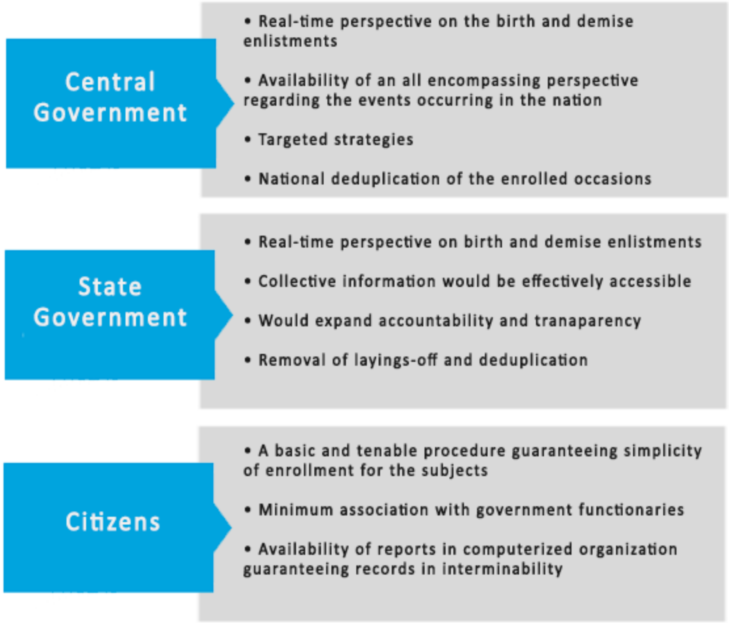

Central Government
• Real-time perspective on the birth and demise enlistments
• Availability of an all encompassing perspective regarding the events occurring in the nation
• Targeted strategies
• National deduplication of the enrolled occasions
State Government
• Real-time perspective on birth and demise enlistments
• Collective information would be effectively accessible
• Would expand accountability and tranaparency
• Removal of layings-off and deduplication
Citizens
• A basic and tenable procedure guaranteeing simplicity of enrollment for the subjects
• Minimum association with government functionaries
• Availability of reports in computerized organization guaranteeing records in interminability

Figure 30- Advantages of Blockchain in registration

Use case: Birth registration through blockchain

Potential process flow through a blockchain for birth registration

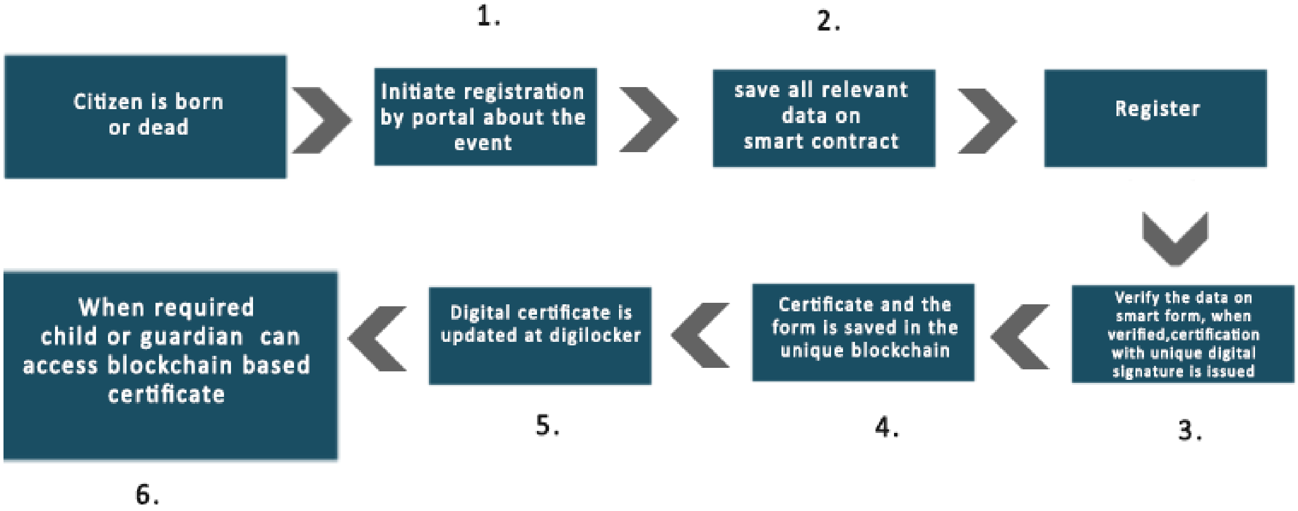

Citizen is born or dead
1.
Initiate registration by portal about the event
2.
save all relevant data on smart contract
Register
3.
Verify the data on smart form, when verified,certification with unique digital signature is issued
4.
Certificate and the form is saved in the unique blockchain
5.
Digital certificate is updated at digilocker
6.
When required child or guardian can access blockchain based certificate

Figure 31-Blockchain in registration

These are the key steps for enlistment and registration of birth and death by means of blockchain:

1. Upon the birth or death of an individual, the Doctor or ANM in control will sign into a portal which is connected to a smart contract in the blockchain system. The specialist will utilize a remarkable property reference number (UPRN), while the *ANM* will utilize an *ANM-ID* for signing in.

2. All important data (date of birth, name, blood group, habitation, guardian data, and etc.) will be updated to editable fields in the smart contract.

3. The smart contract moves to guardians' node, where the guardians of the individual audit the data and carefully sign the form.

4. The smart contract moves to registrar node, where the recorder approves each field in the smart form with a binary response (invalid or valid).

5. If all fields in the forms form are valid, a completely unique digital signature is generated, which is stamped and updated on the blockchain. A cumulative record is generated as a certificate that is now updated to the DigiLocker / similar account of the citizen, stamped with a completely unique digital signature.

7.2 Agriculture field

With 40% of the worldwide workforce utilized in agriculture, the segment is the world's the biggest supplier of employments and can possibly impact billions of lives.

Further, with expanding customer requests for transparency in the food supply inventory network, manufacturers and producers feel difficult to precise information from start to end. Blockchain innovation guarantees to improve discernibility and transparency in agriculture chains. High transaction fees, unfair price and late payment are some different difficulties that exist in agriculture supply chains. Also, food supply transportation and coordination are unpredictable and on occasion require intra continent supply chains. Such supply chains include many staff and several communications with high probabilities for human error[30].

Blockchain innovation can possibly make the agriculture chain increasingly secure, productive and transparent. It guarantees start to finish production visibility and enables one to follow the beginning of production and track an item/produce amid its expedition in a store network. Blockchain arrangements, if actualized, may prompt the disposal of mediators or third part or middle-men, along these lines prompting improved estimating, diminished exchange charges, along these lines dispense with issues of accumulating, and so on. The figure on the following page outlines the key focal points of utilizing a blockchain-based key in the agriculture supply chain.

Use-case: Transparency in agriculture supply-chain through blockchain

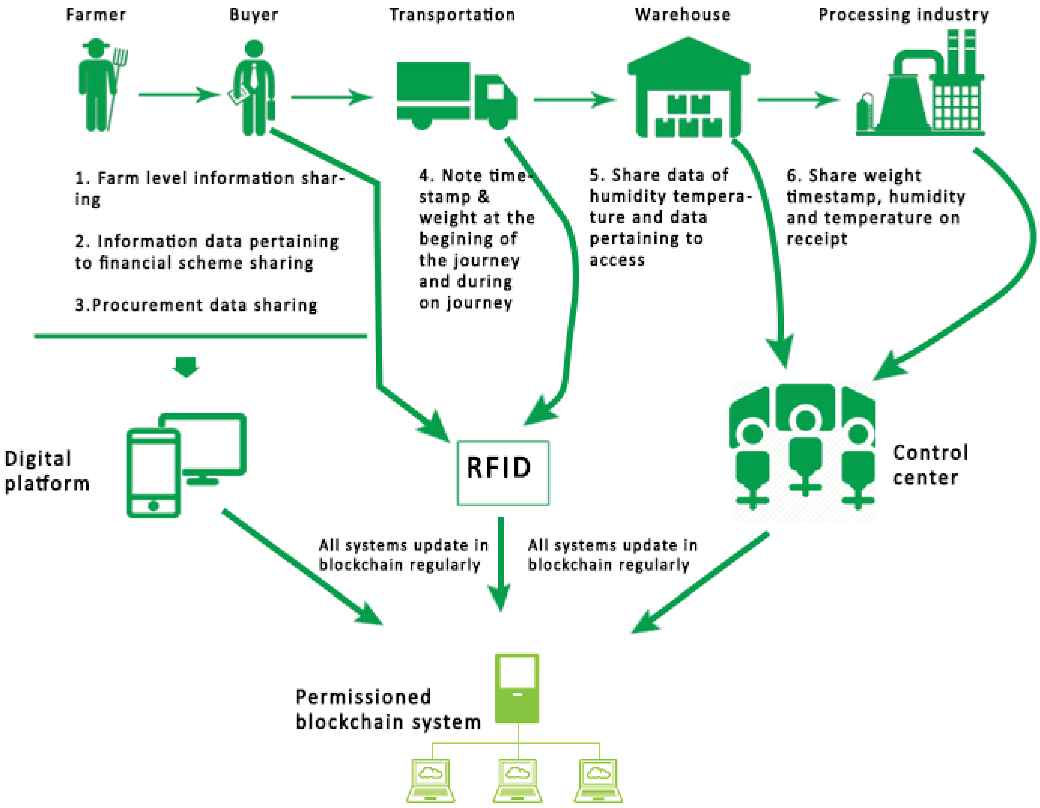

Farmer
Buyer
Transportation
Warehouse
Processing industry
1. Farm level information sharing
2. Information data pertaining to financial scheme sharing
3.Procurement data sharing
4. Note time-stamp & weight at the begining of the journey and during on journey
5. Share data of humidity temperature and data pertaining to access
6. Share weight timestamp, humidity and temperature on receipt
Digital platform
RFID
Control center
All systems update in blockchain regularly
All systems update in blockchain regularly
Permissioned blockchain system

Figure 32- Agriculture blockchain

The arrangement sends a blockchain application intended to work with a basic cell phone interface, either through the application itself or by connecting it with existing interfaces and frameworks for capture data. The accompanying advances portray one of the potential use instances of a blockchain-based inventory network:

1. During harvest, a farmer takes a geo-tagged image of his farm. The harvest is loaded in other means of storage where the farmer takes a geo-tagged picture of the packing area. The farmer also can use a smartphone to share the data on financing or benefit initiatives. Then the farmer takes the gunny bags to the closest buyer.

2. The purchasing agent then records the farmer's product. The agent confirms and registers the product, together with the following information:
   a. Info of the farmer
   b. Quantity obtained
   c. Geo-tagged farm image
   d. Other product information

3. When the produce is enlisted, a QR code is created consequently. The purchasing specialist prints the QR code and sticks it on the bags.

4. The purchasing operator at that point examines different enlisted delivers and makes a dispatch. Every dispatch box is verified with a reed switch which creates an alarm after being altered. The transfer box is verified with a thermocouple and polymeric sensor and an RFID tag.

5. The weight of the shipment is checked before it is transported. This weight is then bolstered into the blockchain by the purchasing specialist. The shipment gets a QR code, which is permanently connected to the QR codes of the packs in the relegation. A smart contract is started for the shipment.

6. The sensors track the capacity states of the transfer during transportation. The RFID label enters a period stamp toward the beginning of the journey and a period stamp toward the finish of the journey at the warehouse and processing plant.

7. Any breach of the normal time taken, storage conditions or altering will be managed as an infringement of a smart contract on the blockchain. For instance, whenever solidified organic products are being transported and the temperature of the transfer is required to be underneath zero degree Celsius consistently, an infringement will be recorded by the keen contract if anytime amid the voyage the temperature contacts zero degree Celsius.

8. Similar parameters are likewise caught at the distribution center such as warehouse before the relegation is delivered to the processing plant.

9. The shipment is conveyed to the preparing plant. The weight of the shipment is checked and recorded in the blockchain and checked against its unique weight. A smart-contract is executed. On the off chance that the heaviness of the shipment isn't equivalent to the weight of the shipment recorded by the purchasing operator, the shipment is rejected.

10. If no damage were raised by the smart-contract and the weight match, the shipment is accepted.

11. At the processing plant industry, a worker will most likely read the QR code of every bag to trace the produce to the geo-labeled farm.

7.3 Social Welfares.

Governments throughout the world spend noteworthy portion of their financial aids on social advantage plans to deal with their sick, debilitated, poor, and the old, underprivileged and minimized. In any case, a significant part of such plans is to guaranteeing that beneficiaries spend the money wisely and smartly. Financial consideration of citizen is one of the key empowering influences for the disbursal of sponsorships specifically to recipient financial balances, thereby decreasing leakages.

An expected adults around 2.5 billion worldwide are rejected from the formal money related framework. In India, more than 600 million individuals need access to managing an account administrations and near 300 million individuals live beneath the official poverty. These individuals to a great extent depend on government welfare installment plans, for example, the National Rural Employment Guarantee Act, 2005, widow benefits, old age pension, scholarships, limited LPG cooking gas and different endowments for their sustenance.

The Government of India makes these installments through the Direct Benefit Transfer (DBT) plot which covers recipients under 407 unique plans with a combined payout of 2.68 lakh crore INR (aggregate till December 2017)[31].DBT means to dispense with debasement, wasteful aspects, and spillages to give comprehensive development, convey better welfare measures and annihilate destitution. Under DBT, every recipient builds up his/her character and delivers different archives to check his/her qualification prior to a few specialists.

Still few difficulties exist, including money related misfortunes through misrepresentation and blunder, an extensive number of unbanked welfare petitioners, cost of unapproved exchanges, high transaction costs and organizing the most defenseless citizens[32].

By evaluating third party to monitor or manage transaction and records, blockchain innovation can enormously lessen exchange expenses and help relieve these difficulties. Utilizing blockchain innovation for social advantage plans will strengthen the government wide policy of supportability, along these lines diminishing destitution and creating an incentive for money in public consumption.

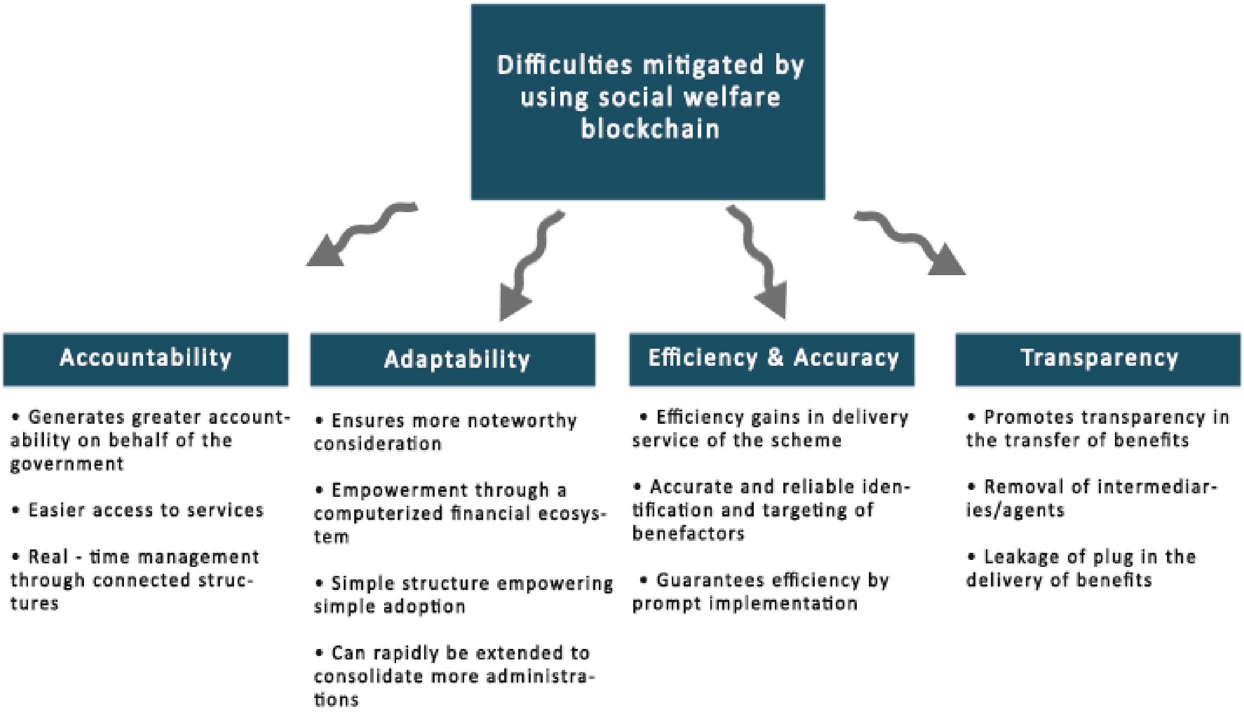

Difficulties mitigated by using social welfare blockchain
Accountability
• Generates greater accountability on behalf of the government
• Easier access to services
• Real - time management through connected structures
Adaptability
• Ensures more noteworthy consideration
• Empowerment through a computerized financial ecosystem
• Simple structure empowering simple adoption
• Can rapidly be extended to consolidate more administrations
Efficiency & Accuracy
• Efficiency gains in delivery service of the scheme
• Accurate and reliable identification and targeting of benefactors
• Guarantees efficiency by prompt implementation
Transparency
• Promotes transparency in the transfer of benefits
• Removal of intermediaries/agents
• Leakage of plug in the delivery of benefits

Figure 33- Difficulties mitigated by using social welfare blockchain

Use-case: Social benefits through blockchain

Blockchain innovation can be utilized to make a protected and very productive welfare infrastructure that can avoid fraud. One of the possible use cases for blockchain-based social advantages installments is appeared as follows. Recipients will get welfare benefits utilizing a framework that contains a versatile application and blockchain framework to record installments, payments sent or gotten by welfare beneficiaries[33].

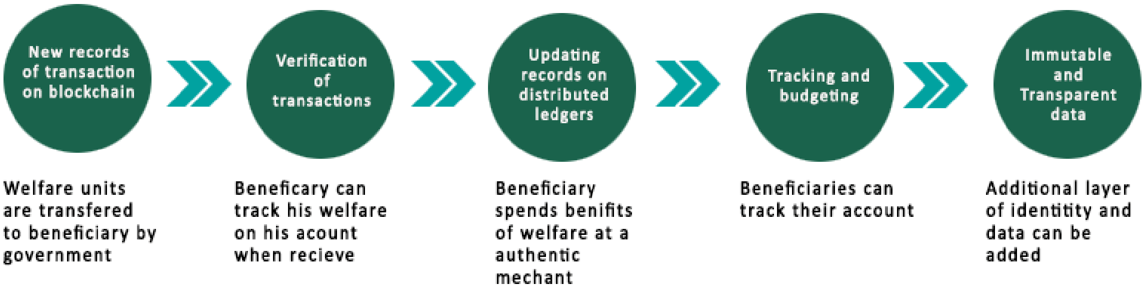

New records of transaction on blockchain
Welfare units are transfered to beneficiary by government
Verification of transactions
Beneficary can track his welfare on his acount when recieve
Updating records on distributed ledgers
Beneficiary spends benifits of welfare at a authentic mechant
Tracking and budgeting
Beneficiaries can track their account
Immutable and Transparent data
Additional layer of identitity and data can be added

Figure 34- Social benefits through blockchain

Several steps when a transaction is done on blockchain[34]:

- Beneficiaries are enrolled dependent on the significant enrolment criteria. The enrolment criteria can conceivably be overseen utilizing a smart-contract where enrolment/de-enrolments are dealt with automatic utilizing the smart-contract.

- Beneficiaries utilize a portable application on their telephones/tablets through which they get their benefits. The framework utilizes a private permissioned distributed ledger to enable clients to store their exchanges.

- Beneficiaries spend their welfare credits/money at pre-checked authentic merchants.

- Beneficiaries at that point see and track their transactions safely on the portable application, which enables them to monitor their spending and plan their spending.

- Their exchanges are recorded on a DL to help their monetary management. This strand of secure, alter safe information on the conveyed record would help in progressively viable data exchange.

- Smart contracts naturally check for change of conditions and refresh and update the ledger occasionally.

Also, the structure makes a steady layer of more extravagant information onto installments so that a more profound and increasingly successful relationship can be built up between the legislature and recipients.

7.4 Land registration management

Land registration indicates to the registration management of all the transaction of lands in any geographical authority of a governing consultant. In many nations, these records are kept up with different government sector or anyone linked to them. Be that as it may, the process of land records does not give 'land title', which now and again prompts title question and overpriced prosecution.

Landowners around the world are defenseless to control by people or gatherings with personal stakes who can undermine the legitimacy of their possession. It is assessed that over 70% of the general population on the planet who possess land have a fragile title against it[35]. On a basic level, a legitimate land title is a vital image of financial portability. This remains constant since one can't borrow against property that isn't legitimately one's own. The circumstance gets increasingly confounded on considering the effect on most by far of individuals that rely upon essential financial exercises, for example, agricultural business for their sustenance, particularly in agrarian economies like Pakistan and India. There are incalculable models everywhere throughout the world where land has been usurped by false records

Key issue with land title:

- Tenuous land titles
- Information silos
- Inefficiencies in record keeping

Use-case: Land registration through blockchain

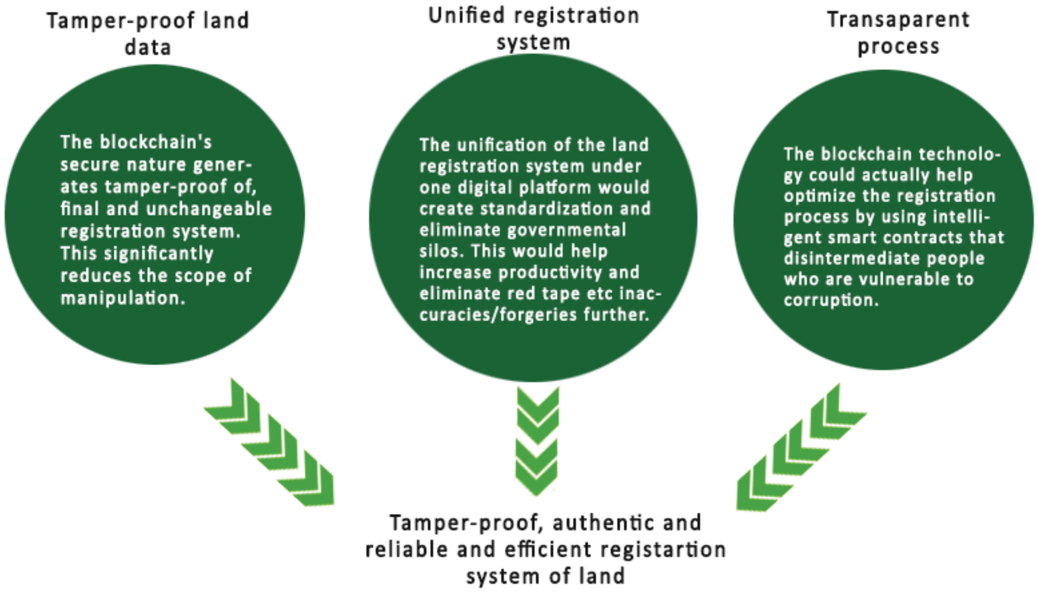

Tamper-proof land data
The blockchain's secure nature generates tamper-proof of, final and unchangeable registration system. This significantly reduces the scope of manipulation.
Unified registration system
The unification of the land registration system under one digital platform would create standardization and eliminate governmental silos. This would help increase productivity and eliminate red tape etc inaccuracies/forgeries further.
Transaparent process
The blockchain technology could actually help optimize the registration process by using intelligent smart contracts that disintermediate people who are vulnerable to corruption.
Tamper-proof, authentic and reliable and efficient registartion system of land

Figure 35-Land registration through blockchain

The management of land registration has been embraced by different governments as one of the first areas to generate a proof of concept (PoC) blockchain. Delaware's state in the United States announced recently two blockchain proposals under the banner' Delaware is completely open for blockchain business,' that included the archiving of state records in an wide open distributed library. Many other significant examples of the use of blockchain for land registration management come from Sweden, Georgia, and Ghana.

A land registration system predicated on blockchain can go a long way to address some of the problems that exist today in land registration.

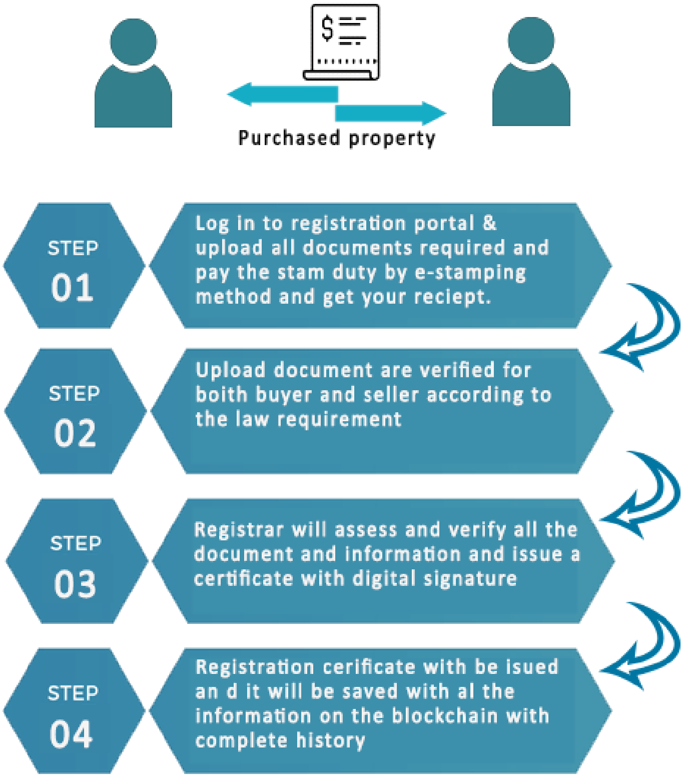
The following actors would be involved in a possible approach to conducting a land registration transaction between parties over a blockchain:

1. **Seller and buyer**
2. **State agency authority**

   The name of the deed is validated against the identity of the buyer and the receipt of the stamp duty is verified. It is liable for verifying the buyer and seller's identity and ownership of the property on the basis of federal regulations.

3. **Office of the sub-registrar of assurances**

   The receipt is verified against the paid stamp duty, property deed, property card and legitimate proof of identity and if the registration process has been completed in a reasonable amount of time.



Log in to registration portal & upload all documents required and pay the stam duty by e-stamping method and get your reciept.

Upload document are verified for boith buyer and seller according to the law requirement

Registrar will assess and verify all the document and information and issue a certificate with digital signature

Registration cerificate with be isued an d it will be saved with al the information on the blockchain with complete history

Figure 36- Land registration process on blockchain

- When the property or land is purchased and required stamp duty is paid, a smart form is triggered after the purchaser/seller uploads legal documents confirming his/her identity, he get the notification of receipt[36].

- The form is forwarded to the relevant land registry authority, which authenticates the documents previously uploaded. The authority shall provide a response (valid / invalid) to each specific field in the intelligent form corresponding to the identity of the buyers and sellers and the truthfulness of the relevant data uploaded. If all fields are legitimate, the smart contract carries out the first business rule to generate a digital property card then the card is marked with a unique digital signature[37].

- Upon verification at the relevant land registry entity, the sub-registrar's office shall receive the form. If the form is submitted in a timely manner, the second business rule is complied with and the form is executed to digitally register the property title to the blockchain network. The purchaser receives a digital registration certificate and the deed is now stored permanently on the blockchain.

7.5 Implemented Case Studies

7.5.1 Case Study 1

Founder & CEO: Emmanuel Buetey Noah
Name: BenBen
Location: Accra, Ghana

Problem

BenBen addresses two structural problems related to the Ghana land registry:
The absence of appropriate and systematic tracking and the lack of digital storage of information prevents officials and property owners from having specific certainty and visibility about who they belong to;

• The connection between property proprietors, government offices (more explicitly the Ghanaian Land Commission) and money related organizations appears to be frail and wasteful. In the expressions of Noah (2017): "you need to physically go to the Commissioner of land to seek in existing registries, at that point bring the right legal documents to the bank". Thus, it could take as long as a year or more– therefore showing immense dangers to the two moneylenders and borrowers.

Solution

BenBen provides all land registries throughout Ghana with a digital register system run by Ethereum. It can certify land information by combining satellite imagery and on-site verification, working together with local stakeholders in the land market. All information is aggregated so that financial institutions and the Lands Commission have access to the data in real time. Based on a

Business-to-business (B2B) model, BenBen doesn't really deal directly with owners of properties. Instead, companies use the BenBen platform to initiate a transaction, confirm a sale, access credit and prove true ownership by both the Lands Commission and financial institutions. In this light, BenBen acts during the entire land transaction process as a risk mitigation tool for financial institutions, governments and property owners.

It gives information instantly, reliably and tampered-free to evaluate the legality of a land ownership claim.

BenBen utilizes three key to assess its effects and victories: the figure of digitalized records, the amount of exchanges signed on the Blockchain; and the amount of records checked with on-the-ground affirmation and satellite imaging. As of 2017, BenBen checks 10,000 records incorporated to its advanced register – with many driving to effective exchanges. Public and budgetary foundations supports the BenBen activity and a few pilots have now been kept running with the Land Commission and Barclays Bank of Ghana.

Result

- ✓ 70% of court debate in Ghanaian national courts are land-related.

- ✓ Average time to get continuous land data from the Lands Commission was one month. This has been diminished to at least 3 days with BenBen's administrations.

- ✓ Average time to get to affirm land privilege was one year. This has been diminished to a normal of three months with BenBen's administrations.

7.5.2 Case study 2

Founder & CEO: The Danish Tax Administration (SKAT)
Name: Vehicle Wallet
Location: Copenhagen, Denmark

Launched: as Proof of Concept (PoC) in 2017

Problem

A car undergoes different phases and activities during its life cycle, such as loan, insurance, repair, MOT testing, and ownership shift. The Danish Tax Administration (SKAT) is a frequently involved stakeholder, as this often includes registrations and levies.

One of the basic exercises identified with a vehicle's lifecycle is the move of private ownership when a vehicle is exchanged and the proprietorship changes starting with one individual then onto the next. For this to occur, the included gatherings are required to round out an official re-enlistment so that SKAT knows the proprietor and consequently can gather the related charges.

When exchanging a vehicle an unevenness of info shows up among dealer and purchaser. The purchaser must trust that the dealer provides him\her valid registration documents. This can be an inborn danger of the vehicle being unfortunately re-manufacture, owing debtors or even stolen property. The dealer then again needs to believe that the purchaser reregister the vehicle. In addition to other things, this infers a danger of the purchaser driving on tolls paid by a merchant or further that purchaser utilizes the vehicle for unwanted issues, in most pessimistic scenario unlawful matters.

Solution

Vehicle Wallet is a joint venture between the installment specialist co-op SKAT and Nets where Blockchain-based advancement is utilized to create a PoC on registered computerized resource for taking care of a vehicle's life cycle process. All information concerning the vehicle is spared in one distributed ledger and makes one concurred and shared the record of the vehicle history as it is exchanged over the store network. This implies no vehicle data irregularity, prompting expanded efficiencies, improved strength with relief from cyber security and extortion dangers. At all stages security, respectability and legitimacy of vehicle data are guaranteed utilizing proven cryptographic services.

The administration controller makes and populates the enlistment for the new vehicle, which is stacked onto the Blockchain. The smart contract convention guarantees that just the controller can do this. The controller at that point exchanges the responsibility for vehicle to the producer by conjuring an exchange on the Blockchain. The exchange is confirmed on the off chance that an agreement exists, for example in the event that every single important gathering concur. The maker includes the make, demonstrate, VIN, and so forth to the vehicle layout, as allowed by the smart contract. This update is obvious to all individuals of the supply chain with the correct authorization. This procedure proceeds over the supply chain network

Transfer of a vehicle's proprietorship is done safely through Vehicle Wallet when dealer starts the transfer by utilizing the VIN number of the vehicle, the beneficiary's id or VAT and the terms of exchange, for example, cost and expiration time. Along these lines the recipient is advised in his or her own wallet and can transfer a bank ensure and acknowledge the deal or reject. At the point when beneficiary satisfies all terms, an "Approved" button will appear. The vehicle's sender can seal the deal. The vehicle will therefore be transferred to a new owner and will appear in his wallet.

Result

The development of a Vehicle Wallet PoC is part of a larger research project focused on the usage of blockchain technology in the Danish Tax Authority. The PoC had a number of valuable results:

- ✓ Hands-on involvement with Blockchain innovation and its advantages so as to make a reasonable business case concerning usage of Blockchain innovation inside the Danish Tax Administration.

- ✓ From SKAT's perspective, a Blockchain arrangement will no doubt wipe out manual procedures attached to re-registration and reducing existing operational costs.

- ✓ A reasonable exhibit of how Blockchain innovation can possibly upgrade certainty and trust among dealer and purchaser when a vehicle changes possession. This is done through consensus mechanism, cryptography, and transaction real-time, totally validation and transparency of the vehicle history.

- ✓ Proof of how SKAT can decrease fraud regarding Vehicle Registration Declarations and different exercises, for example, MOT test and fix using Blockchain innovation since transferring and authorization of false or non-existing information won't be conceivable.

KEY POINTS

During a one-month run, Vehicle Wallet was created in a co-creation process among Nets and SKAT and included four engineers and one planner. Moreover, a few significant experts furnished the task group with their effort and advices.

7.6 Areas where DLT/Blockchain are useful and implemented

7.6.1 Healthcare

The industry of healthcare needs an increasingly proficient and secure framework for overseeing preauthorizing expenses, restorative records, insurance claims, and recording other complex transactions. Blockchain guarantees to give much needed help.

Electronic medicinal records

Electronic medicinal records are as of now kept up in server farms (in a cloudlike domain), and access is restricted to medical clinic what's more, care supplier systems. Centralization of such data makes it powerless against security breaks and can be costly.

Blockchain holds the total restorative history for every patient, with numerous granularities of control by the patient, specialists, emergency clinics, back up plans, etc. giving a protected component to record and keep up far reaching medicinal narratives for each patient. In light of this, the accompanying advantages are figured it out:

- Complete quiet restorative history for exact medication suggestions by doctors
- Tamper-safe methods for putting away restorative narratives
- Reduced time in protection claims goals and expanded effectiveness in giving protection cites

Healthcare installment preauthorization

The expression *clinical attachment* is an idea encompassing the requirement for extra clinical data when a payer is settling a healthcare insurance. Cases are frequently submitted without all required supporting subtlety. Subsequently, payers need to ask extra detail, which adds expenses and deferrals to the settlement process. Further, coordinating and matching those claims with supporting data is quite difficult for different parties involved.

Blockchain can simply handle and restructure this time consuming and annoying process, which would automate and program the gathering and sharing of information. Few more advantages incorporate:

- The framework can recommend elective administrations that have better inclusion.
- Claims can be inspected and paid all the more proficiently and rapidly.

7.6.2 Government

Most government involves tracking assets and monitoring and recording transactions which can all be made increasingly proficient and transparent by using blockchain.

Creating trusted identity remains a concern because of costly background check and forgery required in validation and verification. Millions of individuals worldwide may have tampered their identity and may not be actually who they state they are. Millions upon a large number of refugees and their children go undocumented or missing. Individuals in the less fortunate part of the world might not have adequate confirmation to set up a way of life as required by certain service provider for instance, banks commonly require evidence of residence or bills to create identity, neither of which may exist in the developing nation.

Government can apply blockchain by issuing computerized authentic birth certificate that are time-stamped, tamper free and available to access anywhere in the world. The advantages of this incorporate

- Reduction in human trafficking
- Reduced expenses and time in identity confirmation
- Transparency in grant allotments
- No manipulation in voting system.

7.7 Overview of Potential DLT\Blockchain Application.

DLT based application in other sectors

Financial Services and Infrastructure	• Insurance (with smart contracts) for the automation of insurance payments and the validation of insured events • Loans • Collateral and assets registries • Mortgages services • Trading commodities • Capital market such as trading & settlements of securities ,digital issuance
Trade and Commerce	• Rewards & loyalty programs • Supply chain management • Trade Finance • Product provenance & authenticity (e.g. diamonds, artworks, pharmaceuticals) • Intellectual property registration • Post-trade processing • Invoice management
Government	• Reducing tax fraud • E-voting systems • Record-keeping, such as criminal records • E-Residence • Reducing fraud in government payments • Storing personal records: birth, marriage & death certificates • Protection of critical infrastructure against cyber attacks

Payments & Money	• Payment clearance and authorization • International and national payments and exchanges • Foreign exchange • Digital currencies
Internal system of financial service providers	Replacing internal ledgers maintained by large multinational financial service providers that collect information from different departments, geographies or subsidiaries

Table 6: DLT based application in other sectors

Chapter 8 - Limitation and opportunities in DLT/Blockchain

8.1 Introduction

Understanding the capability of DLT/Blockchain innovation and its apparent capacity to change existing frameworks, business, and procedures, still, challenges exist for the acknowledgment of advantages to some sector recognized by stakeholders. To recognize more extensive of DLT/Blockchain advancements and the role that it could play in its adoption and improvement, it is compulsory to understand the difficulties faced by DLT/Blockchain and possible opportunities also from this technology. This segment of the report starts by examining a portion of the imperative difficulties to the more extensive improvement and usage of DLT/Blockchain.

Few of the challenges and opportunities are discussed below:

Challenges

- Distributed nature of the system.
- Wholly inadequate clarity about the terminology and its unreliable knowledge, create hurdles to the broader acceptance of DLT/Blockchain.
- The risk factor in adopting new technology and merging it with existing modules
- Energy consumption and its cost
- Preserving the privacy and security of data
- Guaranteeing reliability of information through encryption
- Lacking proof on corporate profit and extensive impact on economics
- Lacking information about implementing DLT/Blockchain and how they will support smart contracts
- Less support on how this technology would be overseen by the government
- Insecurity around ruling
- Various non-interoperable implementations and fragmentation[38]

Opportunities

- Eliminating the need of the third-party intermediate which will be effective in reducing the cost also for both business and users
- The acceptance of DLT / Blockchain technologies could provide companies with new sources of revenue
- Opening doors for new business and financial models
- Decentralization technology can improve the adaptability and security and stability of transnational systems[39].
- Improving security and trust in the transactional system and giving the power to end users
- DLT/Blockchain technology reduce the propensity for scam
- Beneficial for recording and reporting activities and data
- Through public key cryptography, it enables digital identities

8.2 DLT/Blockchain Challenges

Multiple sources reiterate the difficulty of understanding what DLT and Blockchain stand for and what technology can actually do. The utilization of the terms *Distributed Ledger Technology* (DLT), and *Blockchain* is regularly conflated. The various approaches and differences in the implementation of DLT/ Blockchain contribute to the lack of transparency in terms of terminology.

There seems, by all accounts, to be an absence of comprehension among organizations, buyers and specialists about how the technology implemented and operate. The apparent immaturity of the innovation also makes difficulties for organizations that conceivably need to utilize DLT/Blockchain.

Early Adoption Risk

Regardless of whether financial advantages are reasonable, the expenses of adoption and usage of DLT/Blockchain for existing organizations might be extensive. This is especially the situation for officeholders with expansive existing complex IT frameworks, back-office processes, or processes made to conform to the available standard which could require costly architecture. Furthermore, the running expenses related to the selection of DLT/Blockchain are up 'til now not clear. Some back-office methods may not be replaced or removed in the near term by a DLT / Blockchain solution. DLT/Blockchain solutions might need to implement well-known sector-specific business practices, both operational and technical, as well as standards, and overcome the cultural resistance of industry leaders in order to accomplish greater market recognition.

Lacking proof on corporate profit and wide impact on economics

In some cases, it is questionable whether a DLT/Blockchain proposed solution improves over a more conventional, centralized ledger, for example, other transactional limitations or in performance. Until further verifications and proof of concept are implemented and steered, the vulnerability in regards to which use case is practical and reasonable. Without more extensive selection among organizations, it is difficult to make an adequately clear evaluation of DLT/Blockchain's wider impact on economics in the medium to long haul.

The enormous limitation in embracing blockchain is 'What are you going to utilize it for, and if you do, are you going to set aside extra cash with it or get more cash-flow that you presently do?' Existing frameworks may be old, yet on the off chance that they carry out the job, for what reason would we pay to withdraw them and explore new technology?

Considerable doubt for new technology, especially in the case of Blockchain.

Insecurity around ruling

Seeing how activities on DLT/Blockchain identify with the more extensive administrative condition or to the advancement of explicit guideline with DLT/Blockchain, it will be a key component in the adoption and improvement of DLT also. Deloitte and Smart Contracts Alliance feature that, from an administrative point of view, the capacities and effect of DLT/Blockchain will could really compare to the innovation itself. Furthermore, administrative bodies will need to build up the abilities required to comprehend and translate the action occurring on the ledger, to distinguish potential threats, and to guarantee existing regulation and user consistency.

Various non-interoperable implementations and fragmentation

To understand the advantages of DLT/Blockchain, it will be tough for ledgers to exchange data with different ledgers and with inheritance IT frameworks. In the short and medium term, it is indistinct whether huge organizations would be willing to update their current operating methods, DLT/Blockchain will be required to exist together with inheritance IT structures and business forms. *De Meijer* described several fragmented DLT/Blockchain frameworks challenging, each with their restrictive, non-interoperable standard, which raises difficulties for competition and interoperability.

Preserving the privacy and security of data

Associations should consider keeping up the security and integrity of stored information on the ledger and of the information relating to ledger activity and transactions. Essentially, for some records, a transparent record might be favored or purposeful although one with capability of restricting user facility to get to viable and sensitive information. Associations should guarantee that information can be accessed by those with proper authorizations. Distributed ledger are seen to be more secure than centralized frameworks, this does not generally convert into security for each and every account

While DLT / Blockchain may offer opportunities in this respect, for example, numerous duplicates of a ledger in the incident of a system failure or cyber-attack, the distribution of access and the executive's rights over numerous nodes may in itself present the risk of security, in that vindictive substances have numerous indirect accesses or back-door, through which to attack the framework. The matter of trust in the framework, determining the integrity of different users in the distributed ledger, and completing information exchanges and transaction in a reliably secure way are in this way key difficulties to more extensive DLT/Blockchain adoption.

Energy consumption and its cost

Certain ledger design of distributed DLT/Blockchain in which changed are made to several ledgers at the same time might consume more energy as compared to centralized one. This is probably going to be a huge issue for permissionless ledgers as compare to permissioned ones, in which scaling can be managed and planned. With extensive amounts of stakeholders and technology with various ways to deal with DLT/Blockchain implementation, the energy consumption expenses of running such a framework and guaranteeing, that effective cost-estimation components are set up especially on the server side may represent a huge challenge

8.3 DLT/Blockchain Opportunities

Eliminating the need of the third-party intermediate which will be effective in reducing the cost also for both business and users

DLT/Blockchain can convert various procedures into automating operation which are at present done through human or trusted third-party interactions, thus providing opportunities for productivity. DLT/Blockchain can expel the requirement for effectively intermediated information synchronization and simultaneousness control by a trusted third-party in a supply chain network, and this could likewise convert into proficiency gains. Similar perceptions were made by Lunn and Brennan, who contend that the chance for sectors which right now depend on trusted third party intermediation could be as improved transactions, cost removal, and innovative revenue.

Opening doors for new business and financial models

Adopting DLT/Blockchain could empower new business and financial models. The facility to exchange resources without a third-party could make open doors for peer-to-peer transactions and therefore encourage the development of the *sharing economy*. There might be openings for the more widely monetary and financial framework, for example, bringing more unique numbers of individuals into the mainstream financial framework.

Example: Royal Mint Gold (trading digital gold through blockchain platform)

Decentralization technology can improve the adaptability and security and stability of transnational systems

DLT/Blockchain can expand the adaptability of frameworks and storage of data because of its distributed nature and less chance of central point failure. DLT/Blockchain gives a modernization that isn't claimed or owned by any single user and that in this manner in the occasion of failure everybody can keep their very own duplicate data. This type of strength and security gives the chance to make a new identity system where users possess the data, which remains all around consistent and can't be altered or destroyed.

It improves trust in Transactions without the involvement of third-party intermediation. It can put end-users responsible for their own transactions and data, as the centralized database does not store their personal information that is progressively helpless against being hacked. Most discussed area of opportunities for DLT/ Blockchain is authentication of trust, cybersecurity, verification and validation of personal identities, and doing as such through transparent automation.

Conclusion

Distributed ledger and blockchain technology can completely change systems and processes surrounded by financial services. The technology could eliminate the need or trusted third-parties, reduce costs and increase revenue or profit for different players inside the industry. Nevertheless, it is not a one-fit solution, as many use-cases need to fit the technology's explicit prerequisites and characteristics. Presently, we still have a debate on whether public or private blockchain systems are more suitable for business use-cases and research is still going on.

Even though public blockchain gives more transparency and data security, if a large number of transactions are to be processed, they are relatively slow whereas private blockchain provides more privacy and enables high transaction speeds but comes along with a low standard of security. Since both system designs have their one of a kind of strength and weaknesses, according to the experts that private and public blockchain are going to converge in the future. Also, the technology is still at an early stage. The estimated period for this technology accessibility for extensive use in economic services is evaluated to be 10 years.

The technology embraces the solid potential for the various sector of financial services, especially in transaction payment, it could reshape the present banking processes. The blockchain could initiate the desperately required digital transformation in trade finance. It enhances the segment by delivering low cast security, risk mitigation, and fast processes[40]. In over-the-counter markets, the technology has the potential to reform the infrastructure of the industry furthermore, lead to the discarding of outdated market contributors. Additionally, it could empower the automation of smart contracts. The existence of numerous uses in these areas underlines their significant potential. For example, in the loaning business, protection, substantial bequest, and considering are further promising sectors. One crucial challenge and requirement also, while making and restructuring of new business model is to deal with the change phase from old to new procedures that integrate DLT/blockchain technology effectively. One method for accomplishing this will without a doubt be the collaboration with organizers all together to build up the lawful structure that is greatly required.

References

[1] https://www.newyorker.com/business/currency/we-need-to-know-who-satoshi-nakamoto-is

[2] http://www.fatf-gafi.org/publications/fatfgeneral/documents/guidance-rbavirtual-currencies.html

[3]Adapted from: “Dubai Aims to Be a City Built on Blockchain”, By Nikhil Lohade, 24 April 2017, Wall Street Journal
https://www.wsj.com/articles/dubai-aims-to-be-a-city-built-on-blockchain-1493086080

[4] Distributed Ledger Technology -
http://documents.worldbank.org/curated/pt/134831513333483951/pdf/WP-PUBLIC-Distributed-Ledger-Technology-and-Blockchain-Fintech-Notes.pdf

[5] Distributed Ledger Technology -

[6] Amended from Dave Birch (Consult Hyperion) in: UK Government Office for Science report “Distributed Ledger Technology: beyond block chain”,

[7] Examples include discussions on platforms like Reddit and GitHub, or decisions taken within a boardroom

[8] https://www.slideshare.net/MichelRauchs/180926-conceptualising-dlt-systems

[9] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[10] Iddo Bentov, Charles Lee, Alex Mizrahi, and Meni Rosenfeld. Proof of activity: Extending bitcoin's proof of work via proof of stake. ACM SIGMETRICS Performance Evaluation Review, 42(3):34{37, 2014.

[11] Sunny King and Scott Nadal. PPCoin: Peer-to-peer crypto-currency with proof-of-stake. Self-Published Paper, August, 19, 2012.

[12] A Blockchain-Based Approach to Health Information Exchange Networks

[13] https://www.omg.org/cloud/deliverables/CSCC-Cloud-Customer-Architecture-for-Blockchain.pdf

[14] https://lisk.io/academy/blockchain-basics/how-does-blockchain-work/what-is-a-peer-to-peer-network

[15] https://nodes.com/

[16] https://medium.com/@Rashwan/bitcoin-and-cryptocurrency-technologies-online-course-summery-lecture-1-f3bb3727ebcb

[17] http://learningspot.altervista.org/a-simple-cryptocurrency/

[18] https://www.dis.uniroma1.it/~querzoni/corsi_assets/1516/GreatIdeas/lesson-bitcoin-slides-1.pdf

[19] https://medium.com/@Rashwan/bitcoin-and-cryptocurrency-technologies-online-course-summery-lecture-1-f3bb3727ebcb https://www.free-ebooks.net/computer-sciences-textbooks/Bitcoin-and-Cryptocoin-Technologies/pdf?dl&preview

[20] https://blockchain-blog.github.io/

[21] https://medium.com/@Rashwan/bitcoin-and-cryptocurrency-technologies-online-course-summery-lecture-1-f3bb3727ebcb

[22] https://medium.com/@Rashwan/bitcoin-and-cryptocurrency-technologies-online-course-summery-lecture-1-f3bb3727ebcb

[23] http://assets.press.princeton.edu/chapters/s10908.pdf

[24] 2 In asymptotic terms, we allow the attacker to try a number of guesses that is a polynomial function of the key size,but no more (e.g. the attacker cannot try exponentially many guesses).

[25] https://cryptocurrencyhub.io/an-introduction-to-consensus-algorithms-proof-of-stake-and-proof-of-work-cd0e1e6baf52

[26] https://medium.com/s/story/how-does-the-blockchain-work-98c8cd01d2ae

[27] https://101blockchains.com/consensus-algorithms-blockchain/#6

[28] Office of the Registrar General & Census Commissioner, India (n.d.) Civil Registration System Division. Ministry of Home Affairs, Government of India. Retrieved from http://www.censusindia.gov.in/vital_statistics/CRS/CRS_Division.html (last accessed on 4 Jan 2018)

[29] Programme Evaluation Organisation, Planning Commission, Government of India. (February 2011). Evaluation study on National Rural Health Mission (NRHM) in seven states. Retrieved from http://planningcommission.gov.in/reports/peoreport/peo/NRHM_v1.pdf

[30] Blockchain technology. (n.d.). Agriculture. Retrieved from http://www.ccgrouppr.com/practical-applications-of-blockchaintechnology/sectors/agriculture/#section1

[31] Direct Benefit Transfer, Government of India. Retrieved from https://dbtbharat.gov.in/ (last accessed on 4 Jan 2018)

[32] Asian Development Blog. (26 September 2017). Direct benefit transfer – a game-changer for financial inclusion in India.Retrieved from https://blogs.adb.org/blog/direct-benefit-transfer-game-changer-financial-inclusion-india

[33] Benjamin Cheah Kai Wai. (24 May 2017). Can blockchains revolutionise social welfare programmes? Retrieved from https://www.benjamincheah.com/2017/05/24/can-blockchains-revolutionise-social-welfare-programmes/ (last accessedon 4 Jan 2018)

[34] UK Government pilot uses blockchain tech for welfare distribution https://bravenewcoin.com/insights/uk-government-pilot-uses-blockchain-tech-for-welfare-distribution

[35] Tapscott, D. (June 2016). Don Tapscott: How the blockchain is changing money and business [Video file]. Retrieved from https://www.ted.com/talks/don_tapscott_how_the_blockchain_is_changing_money_and_business/transcript (last accessedon 4 Jan 2018)

[36] https://blog.bankbazaar.com/stamp-duty-and-registration-at-the-time-of-property-purchase/

[37] https://www.revenue.ie/en/property/stamp-duty/filing-and-paying-stamp-duty/stamp-certificates.aspx

[38] https://www.rand.org/randeurope/research/projects/blockchain-standards.html

[39] https://www.rand.org/randeurope/research/projects/blockchain-standards.html

[40] https://www.huffingtonpost.com/ameer-rosic-/goodbye-corrupt-charities_b_13207806.html