**Module 2: a review of distributed ledger technology platforms**

1. **Distributed Ledger Technology Systems**

A distributed system involves a set of distinct processes (e.g., computers) passing

messages to one another and coordinating to accomplish a common

objective (i.e., solving a computational problem).

– by *Preethi Kasireddy*.

In *distributed computing*, the system is built to coordinate the work of different computing devices work together to achieve a unilateral goal. There are several system architectures and protocols to facilitate this process. These types of system protocols gave raise to consensus concepts like Byzantine Fault, Nakamoto Consensus etc. Distributed computing systems are known for independent failure of components or simple faults, lack of global clock, lack of concurrency. Below is a deeper explanation of most key concepts of distributed computing systems like Blockchain, Hyperledger and Corda networks.

1. ***Blockchain Basics (Blockchain Structures and Blockchain Features).***

Blockchain is referred as the most trusted database system. This is because it is an immutable decentralized ledger where records are replicated on every computer (node) that maintains a consistent state with the network protocol and with other computer peers on the same network. In terms of its governance, Blockchains can be categorize into a public or a private network and at times a hybrid network, also called a consortium.

Over the years, people have tried to standardize and make implement to this technology so as to solve the core current problems of scalability and decentralization which brings with itself the issue of network security. We see lots of different implementation in blockchain networks like Ardor Blockchain, Bitcoin, Ethereum, Cosmo, Zcash, Dash etc. Concepts like Child-chain and Sharding have been introduced to improve scalability without scarifying decentralization and security of the network.

The blockchain network is sustained by some key features which leave us with benefits such as immutability, fast action (in terms of finance: fast transactions), decentralization etc. The Blockchain Technology provides these benefits through its use of cryptographic hashing and encryption algorithms, the implemented consensus protocol (which makes it achieve decentralization). It is also very important to mention the Blockchain incentive mechanism or its reward system. Running a server can be expensive so an economic incentive is mostly attached to help keep participants who run these blockchain computers (especially when the network is a public blockchain).

1. ***Introduction to Hyperledger and Corda framework.***

The [Hyperledger](https://www.hyperledger.org/projects/fabric) and [Corda](https://www.corda.net/) technology are enterprise blockchain frameworks for implementing Permissioned Blockchains or Consortium networks. Blockchain, for what it is known for, comes with some features like decentralization, Publicity in a Pseudo anonymous way etc. This Technology is not in-line with the current enterprise operational. In recent times, a major focus of Co-operations in Blockchain is to find how they can collaborate and benefit from the features of Blockchain.

These frameworks are developed to help business enterprises collaboration in using blockchain. It enables companies to create a blockchain network which grants certain levels of access or enables participants to have different levels of involvement in the network. The activities of the participants (nodes) on the network is controlled by a central governing node.

Hyperledger is founded by the Hyperledger Foundation and Corda is supported by R3. They both are mainly implemented in Java. They don’t have Native Digital Currency but they both have Built-in Smart Contracts.

1. **Consensus Mechanism**
2. ***Consensus Mechanisms***

Distributed Ledger Technologies are in no way new but blockchain technology is a part of DLT that is new. Bitcoin blockchain started the paradigm shift in 2009 when Satoshi dropped the Bitcoin Whitepaper. For the first time, we could solve the problem associated with double spending of digital asset such as cryptocurrency.

There are several blockchain consensus models but the two most prominent is the

1. Proof of Work (PoW)
2. Proof of Stake (PoS)

**Proof of Work (PoW)**

The Bitcoin network is the general practical standard and reference for the PoW model. In PoW, all mining nodes on the network are tasked with the responsibly of maintaining the security of the blockchain. This is done by the miners iteratively guessing to get a solve of a mathematical hash puzzle. From a pool of verified (but not confirmed) transactions, a miner selects transactions which have the minimum transaction fees and these miner forms a data of block of a specific size. Miners’ computers compete to find a hash of this data block with has some specific properties. The miner who finds the first solution is allowed to add the new block of transactions to the blockchain and they are rewarded with some bitcoins.

**Proof of Stake (PoS)**

The large amount of computational power required for a PoW miner to guess faster than his peers and find the next block is costly and energy intensive. A whole industry has grown up around creating custom chips designed only for mining.

PoS is an alternative approach that has gained popularity in recent years and that requires no specialist hardware. This approach was first implemented with NXT in 2013. In PoW, hashrate determines how likely a miner is to add the next block but in PoS, the miner’s staked coins stake his likelihood to find the next block. That is, each network node is linked to an address, and the more coins that address holds, the more likely it is that they will mine (or ‘stake’, in this instance) the next block.

Unlike in the case of PoW, an account with 51% of the coins can attack the system but buying this might push up the market price and making it more expensive for the attacker to harm the network. One can read more about the properties of distributed systems from the links below.

**Further Reading**

[Byzantine Fault](https://en.wikipedia.org/wiki/Byzantine_fault#Byzantine_Generals'_Problem)

[How does Distributed Consensus Work?](https://medium.com/s/story/lets-take-a-crack-at-understanding-distributed-consensus-dad23d0dc95)

[Review of Blockchain Consensus mechanisms](https://blog.wavesplatform.com/review-of-blockchain-consensus-mechanisms-f575afae38f2)

1. ***Blockchain challenges***

As a technology that is just 10yrs old but really arrived about 6yrs ago, there are lots of challenges which are still to be resolved. The *Blockchain Trilemm*a of Scalability, decentralization and security, remains a core challenge for distributed computer systems like Blockchain. The Proof of Stake protocol is a model that reduces energy waste without sacrificing decentralization and security but this doesn’t scale the speed of the system.

The Ardor Blockchain is one of such Proof of Stake implementations. Ardor uses the Parent-Child chain architecture to solve scalability issue. More explanation will be available in later chapters of this work.

**Further Reading:**

[The Blockchain Trilemma: Decentralized, Scalable and Secure](https://www.certik.org/blog/20191004_The-Blockchain-Trilemma:-Decentralized-Scalable-and-Secure).

1. ***Blockchain Hacks and Security***

There is no doubt that are systems where important information (like transaction records) are kept are susceptible to hacking, but the question remains that, if blockchains are hackable. In computer science, hacking generally refers to gaining of unauthorized access to data in a computer system or network. This is achieved by finding and exploiting the weaknesses of the system. Also, Hacking can be defined as the habitual activity of someone who possesses exceptional skills and enjoys researching and analyzing the innermost intricacies in computer programming. Considering both definitions hacking is not generally bad thing.

Technically speaking, Blockchains cannot be hacked with the state of today’s science, for the data is public and encrypted to the world most secured ciphers. The blockchain can be attacked. Attack in this sense is the hijacking of the system or network for some malicious benefits to a user.

These attacks can be categorized into

1. Peer-2-Peer network attacks,
2. Consensus protocol attacks,
3. Smart Contract attacks and
4. Wallet attacks.

The most popular of these attacked includes Mining malware, 51% attack, Sybil attack, Selfish mining, etc.

The security of the system is heavily depended on its decentralization but the issue with decentralizations is that is reduces scalability in terms of super-fast transaction verification, data block confirmation and message broadcasting. Some concepts are being developed through research, and some are being implemented. These are beyond the scope of this course.

**Further Reading:**

[What is Computer Hacking?](https://study.com/academy/lesson/what-is-computer-hacking-definition-prevention-history.html)

[10 Blockchain and New Age Security Attacks You Should Know](https://blogs.arubanetworks.com/solutions/10-blockchain-and-new-age-security-attacks-you-should-know/).

1. **Blockchain Primitives**
2. ***Cryptographic Hash Function***

In general, Cryptography is converting data into a hiding format which is unreadable or not accessible to an unauthorized user thereby allowing information to be kept secret. This ensures data integrity and security.

Cryptographic [Hash](https://simple.wikipedia.org/wiki/Cryptographic_hash_function) functions is the process of calculating an input or message to returns a data of fixed-size string. This string is called the hash value, message digest, digital fingerprint or signature, checksum. It can be in 16bits, 32bits, or 64bitd Hexadecimal format. The Bitcoin blockchain uses SHA-256, Ethereum uses KECCAK-256 and Ardor Blockchain uses SHA-256. There are other types of Hashing Algorithms like MD5, SHA-2, SHA-3 etc.

**The benefits of Cryptographic Hash functions:**

* It is main use is to compare two files for equality without opening the two or more document files.
* Hashing is also used to verify the integrity of a file after it has been moved or transmitted from one place to another. This can be to ensure the transferred file is not corrupted.

**Properties of Cryptographic Hash functions.**

* Computational Efficient: That is, it should be fast enough but not too fast just to avoid security challenges.
* Consistence (Avalanche Effect): It should give you the same hash or any little change in the input hash should give a large change in the output hash.
* Collisions Resistance: It should be near impossible to find two inputs or messages with the same hash.
* Deterministic: They should be deterministic, in the sense that when you recompute the same message.

**In blockchain Cryptographic Hash functions are used for**

* Message authentication and Signing,
* Encryption
* Mining
* Identifying
* Public-key Cryptography etc.

**Further Reading:**

# [Hashing Algorithms and Security](https://www.youtube.com/watch?v=b4b8ktEV4Bg).

[Blockchain Underpinnings: Hashing](https://medium.com/@ConsenSys/blockchain-underpinnings-hashing-7f4746cbd66b).

[Cryptographic Hash Functions Explained: A beginner’s Guild](https://komodoplatform.com/cryptographic-hash-function/).

1. ***Public-key Cryptography and Digital Signatures****:*

Also known as Asymmetric cryptography is a system that uses a pair of Public and Private Keys. The system might be designed to generate some random input data from which it will create the private key and the public key. Effective security of this system requires the users to keep the private key really private. The public key of the same pair can be openly and widely distributed and there will be no security compromise. But the encryption strength is directly tied to the key size which increases exponentially with increase in key size.

In such a system, any person can encrypt a message using the receiver's *public key*, but this message can only be decrypted with the receiver's *private key*.

A sender can combine a message with a private key to create a [*digital signature*](https://en.wikipedia.org/wiki/Digital_signature) on the message. Anyone with the corresponding public key of the sender can combine this data, to verify whether the signature is valid, i.e. if it was signed by the owner of the corresponding private key. For instance, Alice wants to send a message to the network. She will hash this message and sign it by encrypting with it her private key. Any verifier can decrypt the message with Alice public-key and also hash a copy of the original message and then compare both data.

Some widely used public-key cryptography include: Rivest-Shamir-Adleman (RSA), Elliptic Curve Cryptography (ECC) and Elliptic Curve Digital Signature Algorithm (ECDSA).

**Public-key Cryptography features for Blockchains**

* **Encryption:** Only the owner of the private key can decrypt a message with the corresponding public key.
* **Authentication:** i) Pubic key can verify ownership and ii) Used for Signing transactions.

**Further Reading:**

[Public-key Cryptography](https://en.wikipedia.org/wiki/Public-key_cryptography).

[Public-key Infrastructure](https://en.wikipedia.org/wiki/Public_key_infrastructure).

# [Asymmetric cryptography (public key cryptography)](https://searchsecurity.techtarget.com/definition/asymmetric-cryptography).

1. ***Merkle Tree:***

This is also known as hash tree. They are a type of data structure that are derived from a set of data where every leave in the tree is a hash of a data set and every node is the hash of it two child elements. It simulates a hierarchical tree structure with the nodes labelled tail and head, and the data blocks are hashed to label the tails. The head is matched in pairs and hashed to produce the next level of tails. This is continued till we get one single hash



Merkle trees allow efficient and secure verification of the contents of a large data structure. They provide a cryptographically authenticated data structure.

**Feature of Merkle tree in blockchain:**

* Cryptographical Authentication.
* Efficient Verification

In blockchain, moving up the tree, each node is the hash of its child elements and this will result to a single master hash or the root hash.

**Benefits of Merkle Tree:**

The benefit that this provides is the concept that Satoshi described as “simplified payment verification”: instead of downloading *every* transaction and every block, a “light client” can only download the chain of *block headers*, 80-byte chunks of data for each block that contain only five things:

* A hash of the previous header
* A timestamp
* A mining difficulty value
* A proof of work nonce
* A root hash for the Merkle tree containing the transactions for that block.

If the light client wants to determine the status of a transaction, it can simply ask for a Merkle proof showing that a particular transaction is in one of the Merkle trees whose root is in a block header for the main chain.

This gets us pretty far, but Bitcoin-style light clients do have their limitations. One particular limitation is that, while they can prove the inclusion of transactions, they cannot prove anything about the current state (e.g. digital asset holdings, name registrations, the status of financial contracts, etc.). How many bitcoins do you have right now? A Bitcoin light client can use a protocol involving querying multiple nodes and trusting that at least one of them will notify you of any particular transaction spending from your addresses, and this will get you quite far for that use case, but for other more complex applications it isn’t nearly enough; the precise nature of the effect of a transaction can depend on the effect of several previous transactions, which themselves depend on previous transactions, and so ultimately you would have to authenticate every single transaction in the entire chain. To get around this, Ethereum takes the Merkle tree concept one step further.

**Further Reading:**

[Merkle Tree Structure](https://en.wikipedia.org/wiki/Merkle_tree)

1. ***Nodes***

In computer science, a node can be any active electronic device with a software protocol that can enable it connected to other similar protocol over the internet. These devices include computers, Tabs, Pods and phones. The role of nodes in a blockchain network is to create a network by maintaining a copy of the blockchain, process and verify transactions, and in some cases create new blocks by mining or forging.

Unlike in some networks like the Dash network, every node is considered equal but they have various length they can get involved in maintaining the network. These they can do by participating by

* keeping a full-copy of the blockchain
* keeping a shallow-copy of the blockchain
* confirming transactions (by Mining or Forging)

1. ***Smart Contracts***

In 1996, a well-respected Computer Science Researcher and Cryptographer named Nick Szabo introduced the concept of a Smart Contract and he wrote an entire research paper which he titled: “Smart Contracts: Building Blocks for Digital Markets”. His definition states that,

*“A smart contract is a set of promises specified in digital form including*

*protocols with which the parties perform on these promises.”*

Smart Contracts are business logic which are executable by computers. When used on the blockchain, they allow for real world constrains to be realized on the blockchain. A Smart Contract has the following characteristics:

* **Trustless**: Removes third parties or at least reduces them.
* **Trackable**: Transactions can be trackable and Auditable.
* **Irreversible**: Transactions are final and Immutable.
* **Self-executing**: Reduce cost and self-enforcing.

**Further Reading:**

# [Smart Contract: Building Blocks for Digital Markets](http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html)

# [Everything You Need to Know About Smart Contracts: A Beginner’s Guide](https://hackernoon.com/everything-you-need-to-know-about-smart-contracts-a-beginners-guide-c13cc138378a)

1. ***Blockchain Forks***

**What is a Blockchain Fork?**

A fork can be said to be an intentional or unintentional blockchain state upgrade that introduces new rules to the network (i.e. making network changes) which is not compatible with the older version of the network’s software. This can be a case where the nodes do not agree on the same state of the system. For instance, a new rule that allows block size to be 2MB instead of 1MB would require a hard fork.

Unintentional fork might arise from two nodes on the same chain finding different blocks at the same time. This will split the chain into two chains and nodes will start working on both chains. However, the node protocol always has it that the chain with most block solutions (i.e. the longest chain), becomes the main chain. A node who decides to mine on both chains is only halving its mining power.

There are basically, two types of forks:

* 1. The Hard Fork and
  2. The Soft Fork.

**Hard Forks**

There major difference between Hard and Soft fork is that the hard fork gives

the node the option to opt in while the soft fork allows node no opting at all.

* By [Vitalik Buterin](https://vitalik.ca/general/2017/03/14/forks_and_markets.html).

Hard fork is a system upgrade where the nodes has to choose either to be on the old chain or to join the new chain. This is to say that hard forks introduce new rules which are not compatible with the old chain. Nodes that continue to run on the older version will see the new transactions as valid but will not to be able to mine on this new chain till they upgrade their node. We have some types of Hard forks like, Bilateral Hard Fork which happened in the case of [Bitcoin Core](https://bitcoin.org/en/) and [Bitcoin Cash](https://www.bitcoincash.org/).

**Soft Fork**

A soft fork is a software upgrade that implements protocol rules that affects or set criteria for valid transactions and nodes on the old chain can still get on the new chain provided that majority of the nodes and miners are on the new chain. For instance, nodes who did not upgrade will still see the new the new transaction which is set by the new rule and can still mine on this chain. However, if the soft fork upgrade succeeds (i.e. majority of the nodes and miners are on the new chain), then the unforked chain does not exist.

**Further Reading:**

[What are Blockchain Forks?](https://www.youtube.com/watch?v=boARRLlu5Uc)

[Learn what are Bitcoin Forks](https://blockgeeks.com/guides/bitcoin-forks-guide/)

A Short Paper: [An Empirical Analysis of Blockchain Forks in Bitcoin](https://fc19.ifca.ai/preproceedings/24-preproceedings.pdf)