**Blockchain 2.0**

* Interested by the Bitcoin revolution, mainstream companies are trying to use the central blockchain idea and build alternative systems around it for use in industry, manufacturing, supply chain, finance, governance, IoT etc.
* This movement of academia, industry, startups is termed Blockchain 2.0
* Blockchain is a powerful technology- capable of going much further than financial transaction
* A decentralized platform- can be utilized to avoid intermediates (The middle man)

**Smart Contracts:** An automated computerized protocol used for digitally facilitating, verifying or enforcing the negotiation or performance of a legal contract by avoiding intermediates and directly validating the contract over a decentralized platform- **faster, cheaper and more secure.**

**Advantages:**

**Immutable:**no party will be able to change the contract once it is fixed and written to the public ledger(the blockchain)

**Distributed:** all the steps of the contract can be validated by every participating party-no one can claim later that the contract was not validated

**Why blockchain?**

* + The blocks are immutable
  + The information is open-everyone can check and validate

**BLOCKS**

* Digitally signed and encrypted transaction **verified by the peers**
* **Cryptographic security-** ensures that participants can only view information on the ledger that they are authorized to see.

**STRUCTURE OF BLOCK**

Metadata about a block

* 1. Previous block hash
  2. Mining statistics used to construct a block
  3. Merkle tree root

**Previous block hash:** every block inherits from the previous block-we use previous blocks hash to create the new blocks hash- **make the blockchain tamper-proof**.

* **Mining**- the mechanism to generate the hash
* the mechanism needs to be complicated enough, to make the blockchain tamper-proof
* **bitcoin mining**: Hk = Hash(Hk-1||T||Nonce)
* find the nonce such that Hk has certain predefined **complexity** (numbers of zeros at the prefix)
* The header contains mining statistics-timestamp, nonce, and difficulty

**IPFS**

* Each File and all of the blocks within it are given a unique fingerprint based on a cryptographic hash
* IPFS removes duplication across the network and track version history for every file
* Each network node stores only the content it is interested in and some indexing information that helps to figure out who is storing what
* When looking up files, you ask the network to find nodes storing the content behind a unique hash
* Every file can be found by **human readable names** using a decentralized naming system called **IPNS(inter planetary naming system)**

**There can be various types of faults in a distributed system.**

* 1. Crash Fault: A node suddenly crashes or becomes unavailable in the middle of a communication
  2. Network or Partitioned Fault: A network fault occurs and the network gets partitioned
  3. Byzantine Faults: A node starts behaving maliciously