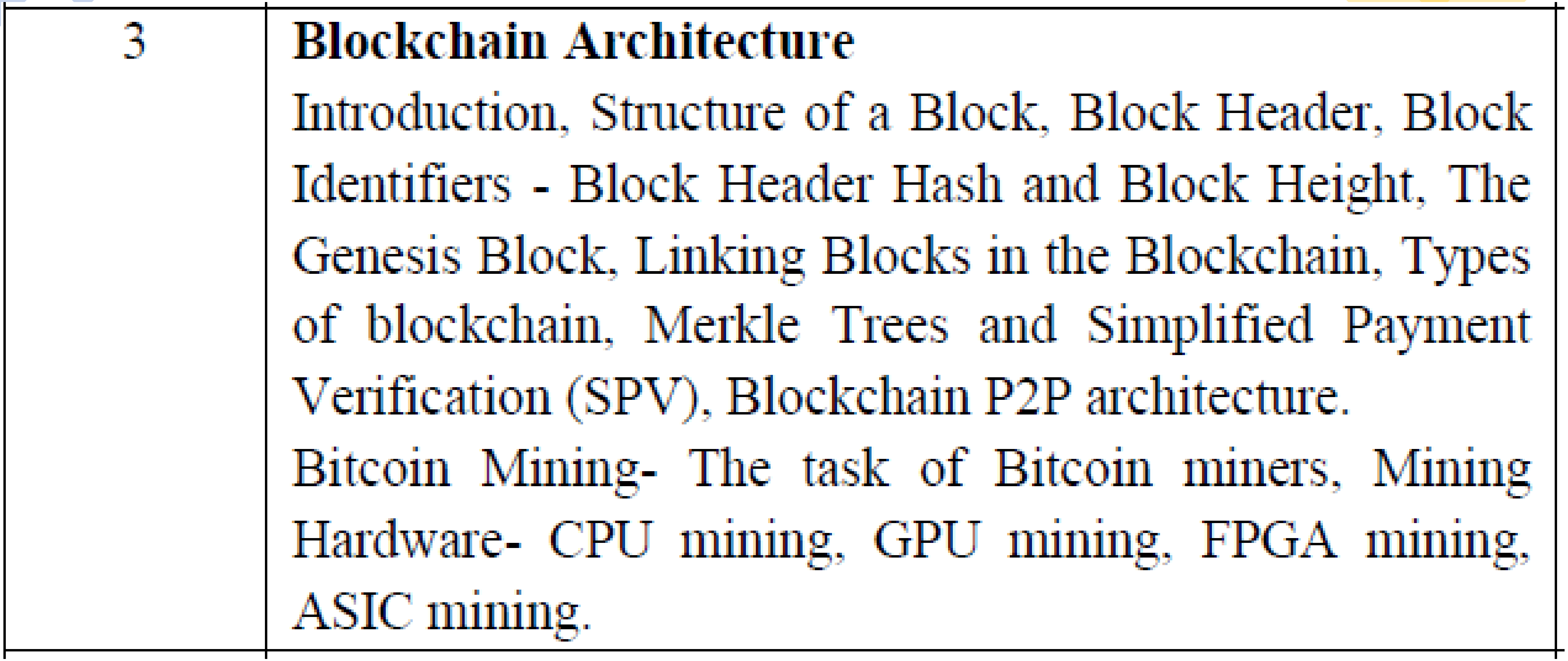
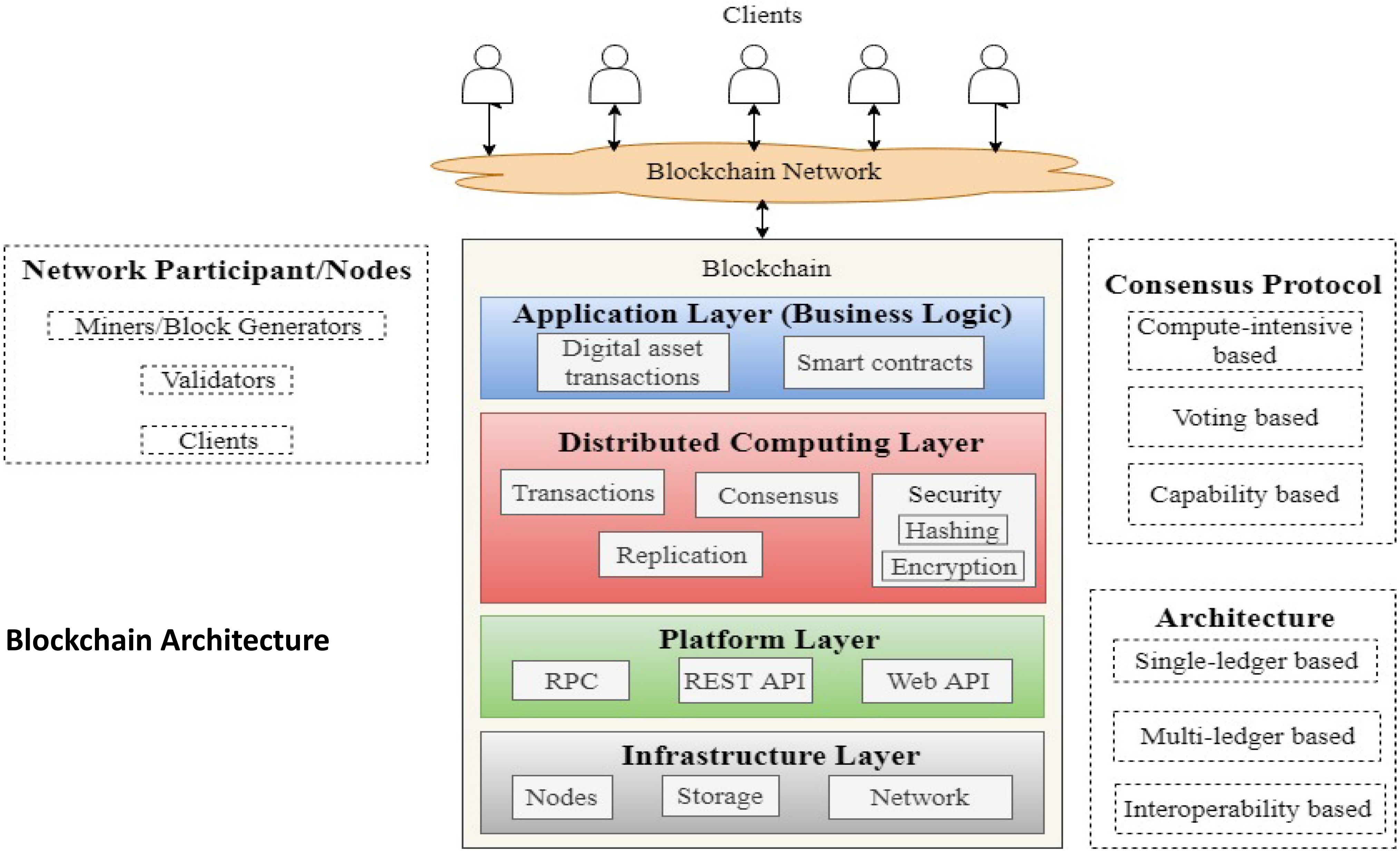


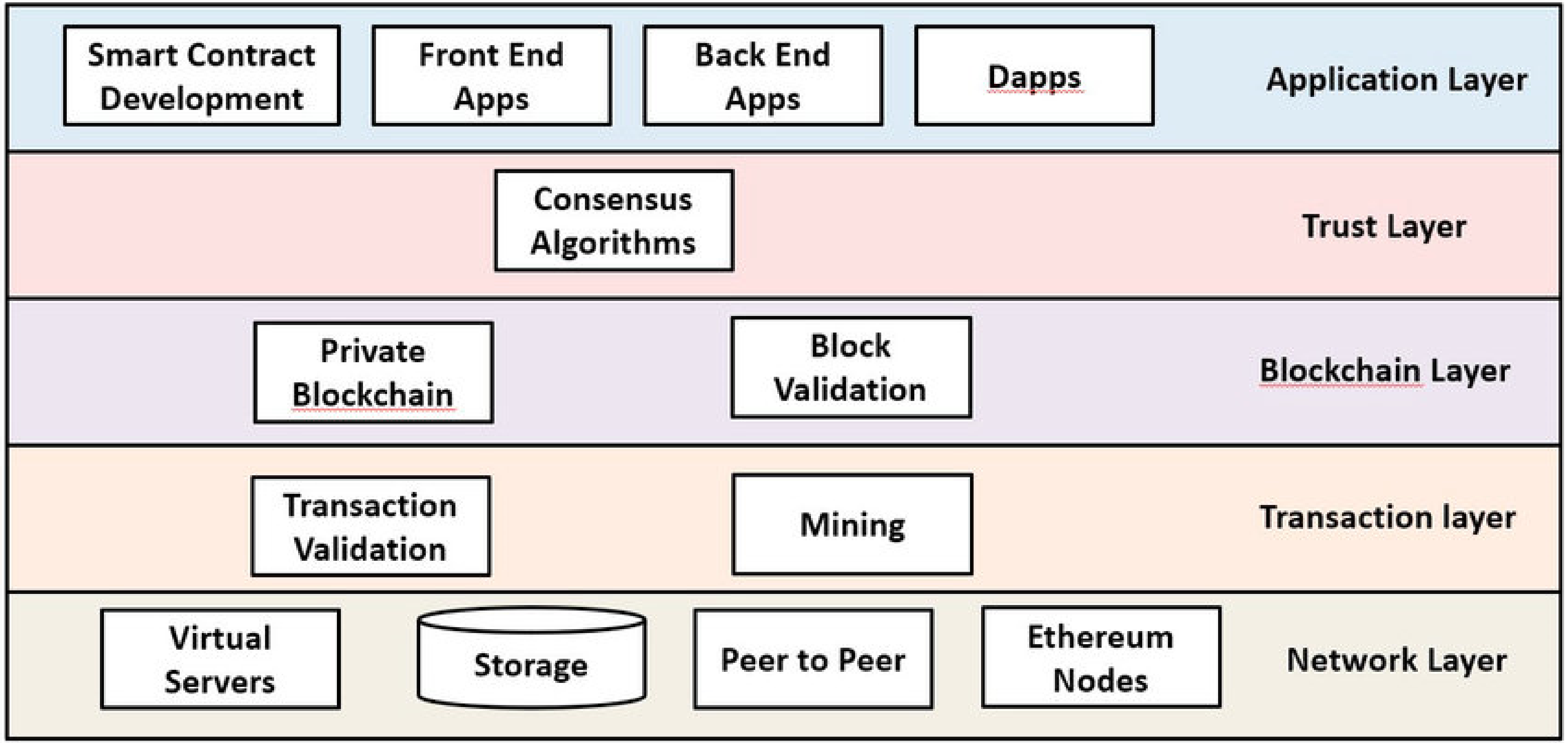
3	**Blockchain Architecture** Introduction, Structure of a Block, Block Header, Block Identifiers - Block Header Hash and Block Height, The Genesis Block, Linking Blocks in the Blockchain, Types of blockchain, Merkle Trees and Simplified Payment Verification (SPV), Blockchain P2P architecture. Bitcoin Mining- The task of Bitcoin miners, Mining Hardware- CPU mining, GPU mining, FPGA mining, ASIC mining.



Blockchain Architecture

https://www.mdpi.com/2073-8994/11/10/1198

Blockchain Architecture



Smart Contract Development
Front End Apps
Back End Apps
Dapps
Application Layer

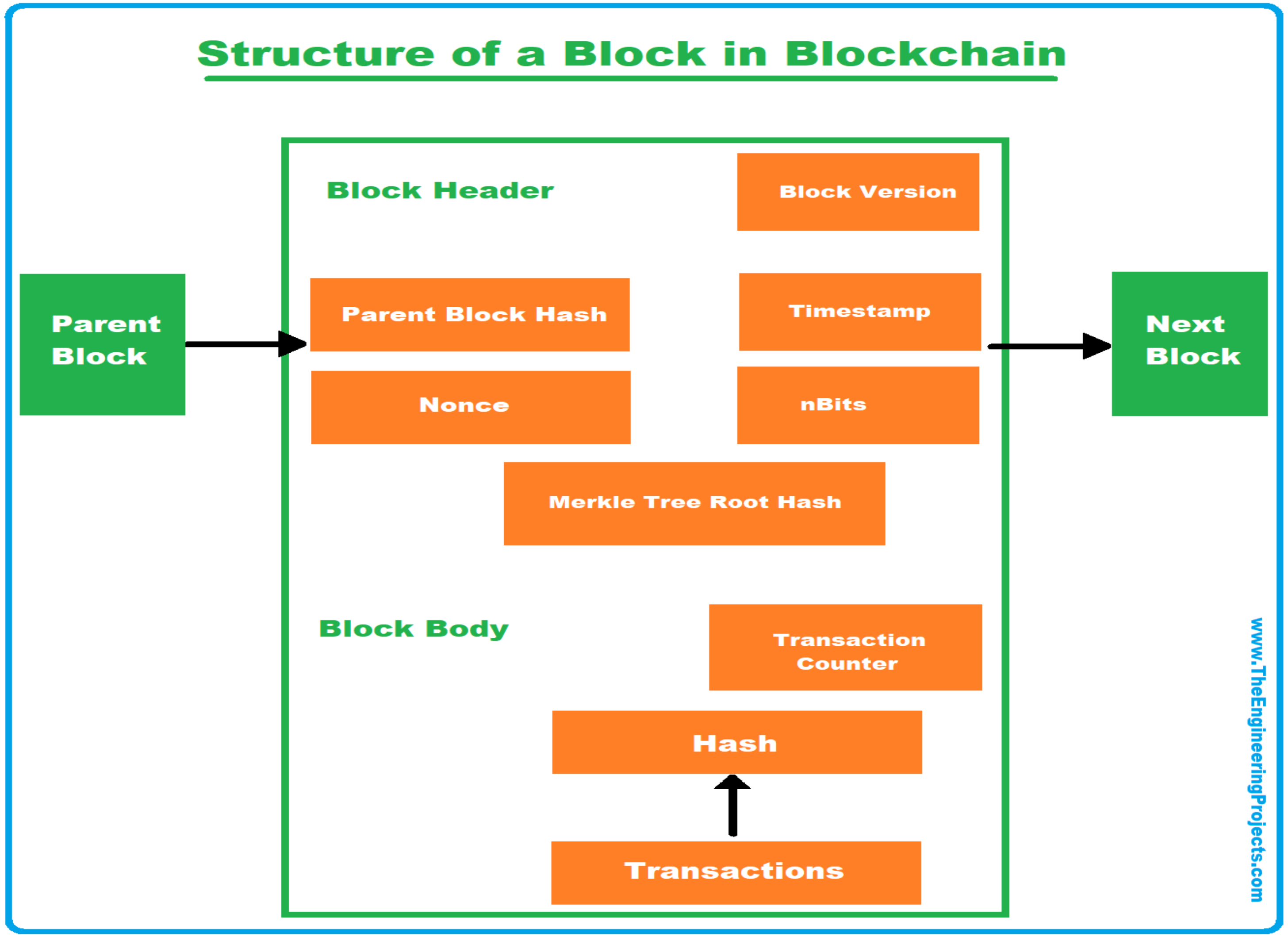
Consensus Algorithms
Trust Layer

Private Blockchain
Block Validation
Blockchain Layer

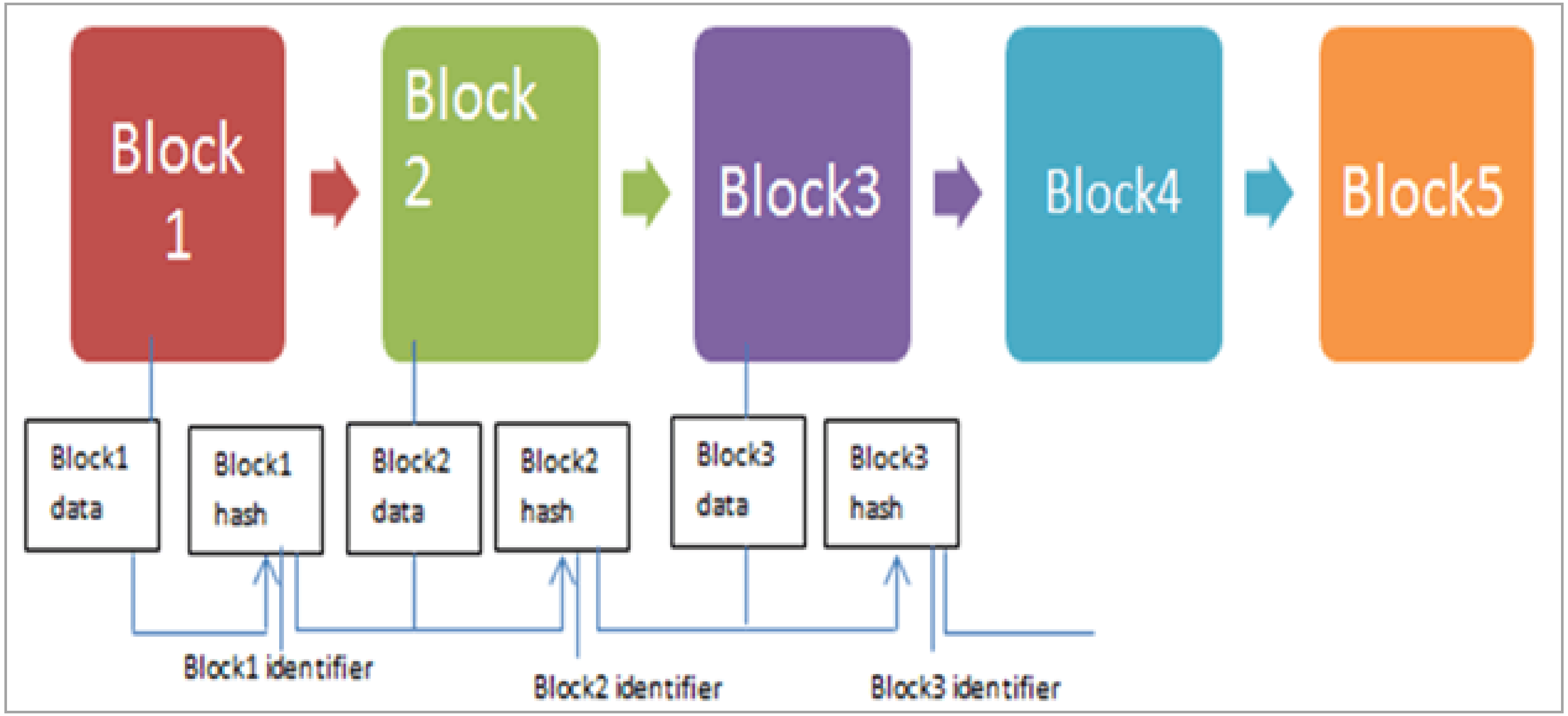
Transaction Validation
Mining
Transaction layer

Virtual Servers
Storage
Peer to Peer
Ethereum Nodes
Network Layer

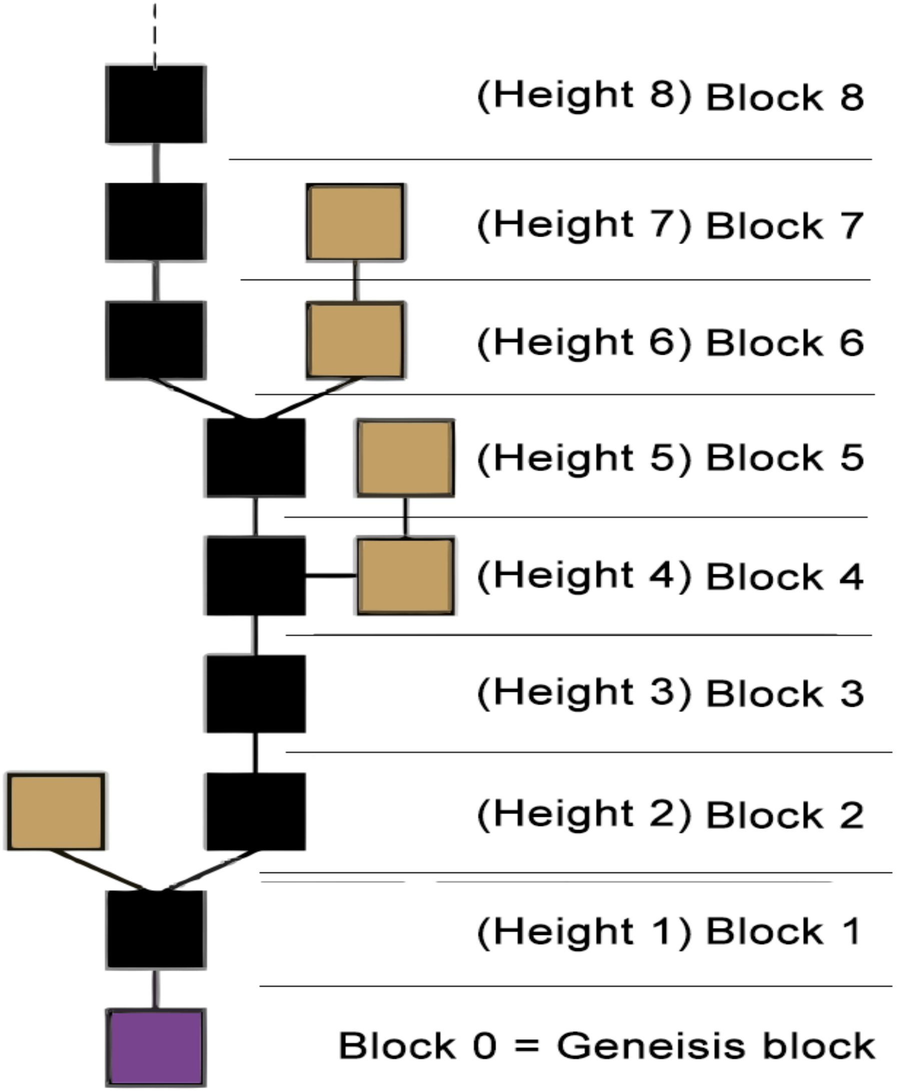
https://www.researchgate.net/publication/340700069/figure/fig3/AS:881305894002688@1587131107501/Blockchain-layered-architecture-31-Proposed-System-Model-Ethereum-Private-Blockchain_W640.jpg



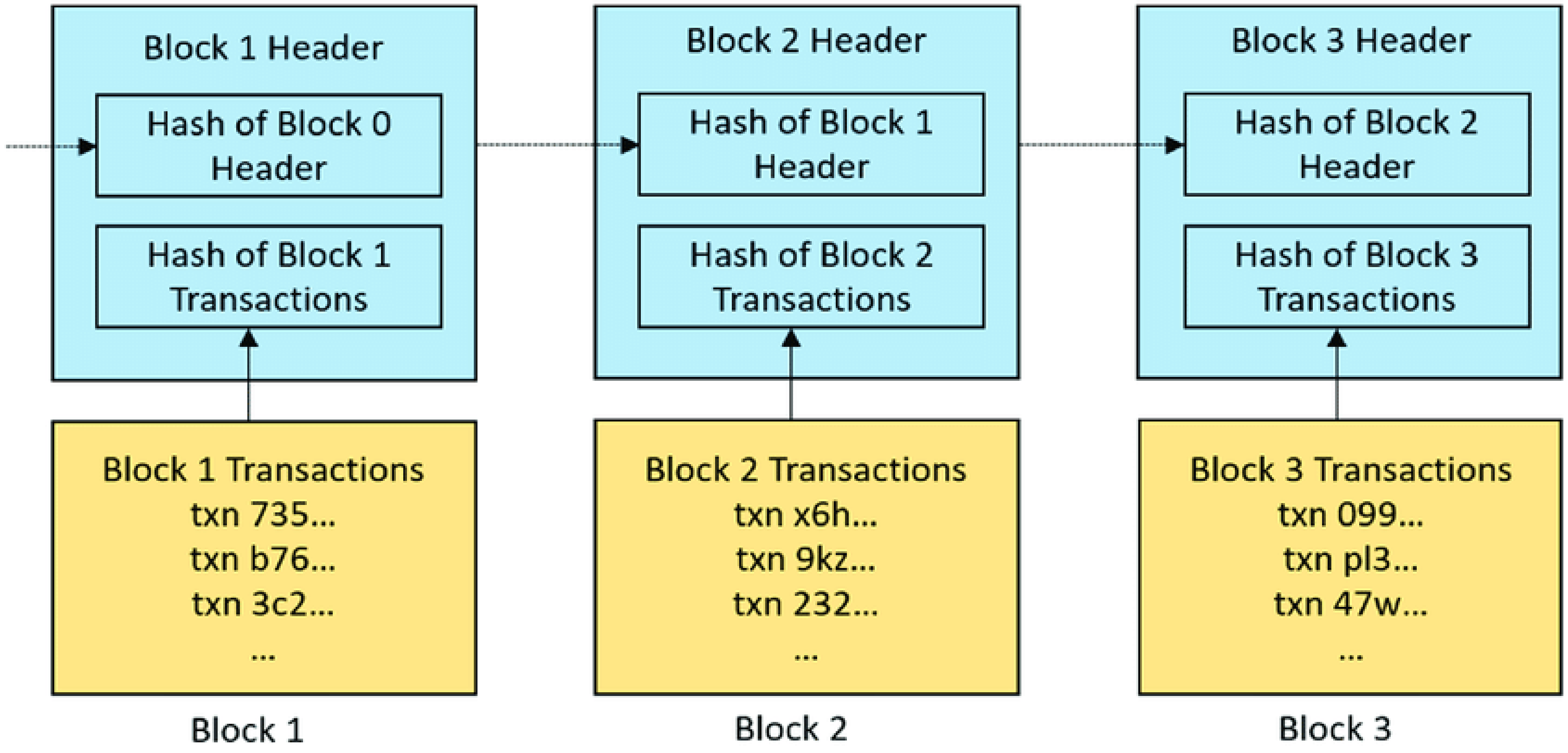
https://www.theengineeringprojects.com/2021/06/structure-of-a-block-in-blockchain.html



https://www.softwaretestinghelp.com/blockchain-security/



https://coinguides.org/block-height/



Block 1 Header

Hash of Block 0 Header

Hash of Block 1 Transactions

Block 1 Transactions
txn 735...
txn b76...
txn 3c2...
...

Block 1

Block 2 Header

Hash of Block 1 Header

Hash of Block 2 Transactions

Block 2 Transactions
txn x6h...
txn 9kz...
txn 232...
...

Block 2

Block 3 Header

Hash of Block 2 Header

Hash of Block 3 Transactions

Block 3 Transactions
txn 099...
txn pl3...
txn 47w...
...

Block 3

https://www.researchgate.net/publication/332215097_Blockchain_Technology_in_Healthcare_A_Systematic_Review/figures?lo=1

Genesis Block

A Genesis Block is the name given to the first block a cryptocurrency, such as Bitcoin, ever mined.

A blockchain consists of a series of so-called blocks that are used to store information related to transactions that occur on a blockchain network.

Each of the blocks contains a unique header, and each such block is identified by its block header hash individually.

These blocks get layered—one on top of the other, with the Genesis Block being the foundation—and they grow in height until the end of the blockchain is reached and the sequence is complete.

The layers and deep history of each sequence is one of the things that makes a blockchain-based cryptocurrency so secure.

Genesis Block

- Bitcoin's Genesis Block was the first instance of a proof-of-work blockchain system and is the template for all other blocks in its blockchain.
- In 2009, Bitcoin's pseudonymous developer, Satoshi Nakamoto, created the Genesis Block, which launched the cryptocurrency boom that is ongoing today.
- The Genesis Block forms the foundation of the Bitcoin trading system and is the prototype of all other blocks in the Bitcoin blockchain.

The following identifier hash belongs to the genesis block:

https://blockchain.info/block/000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f

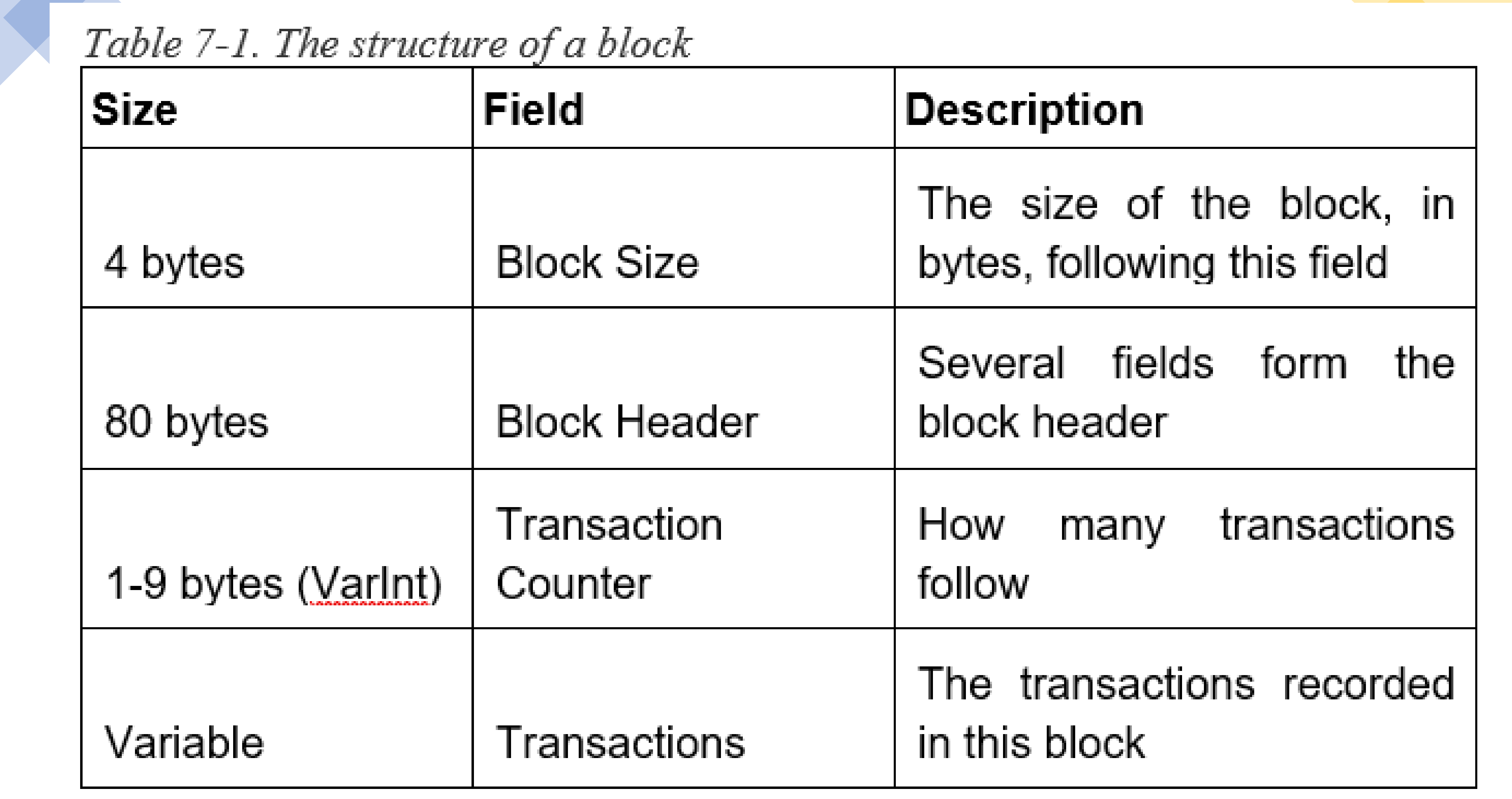


Table 7-1. The structure of a block

Size	**Field**	**Description**
4 bytes	Block Size	The size of the block, in bytes, following this field
80 bytes	Block Header	Several fields form the block header
1-9 bytes (VarInt)	Transaction Counter	How many transactions follow
Variable	Transactions	The transactions recorded in this block

What Is a Cryptocurrency Block Header?

- ❑ A block header is used to identify a particular block on an entire blockchain and is hashed repeatedly to create proof of work for mining rewards.
- ❑ A blockchain consists of a series of various blocks that are used to store information related to transactions that occur on a blockchain network.
- ❑ Each of the blocks contains a unique header, and each such block is identified by its block header hash individually.

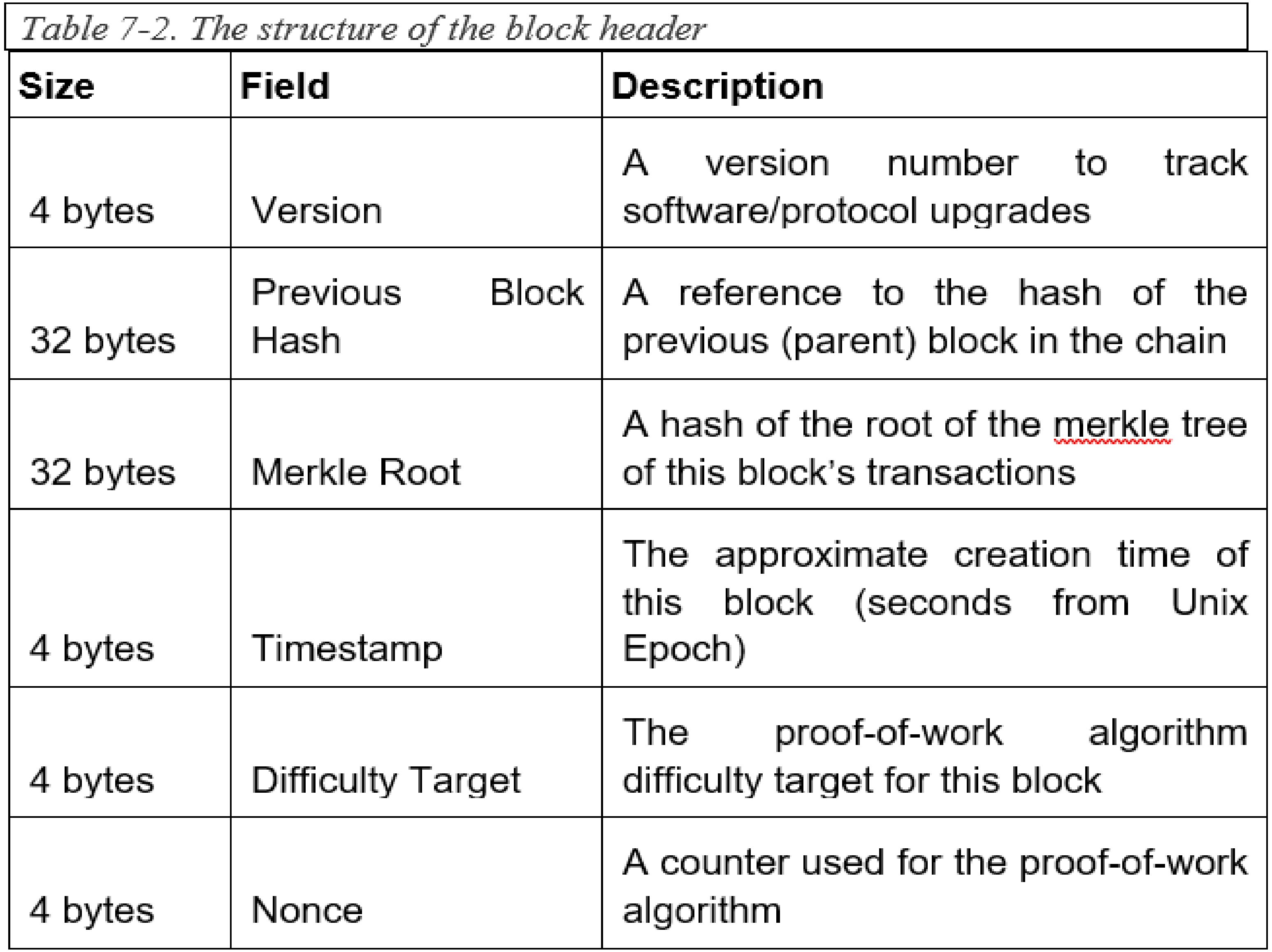
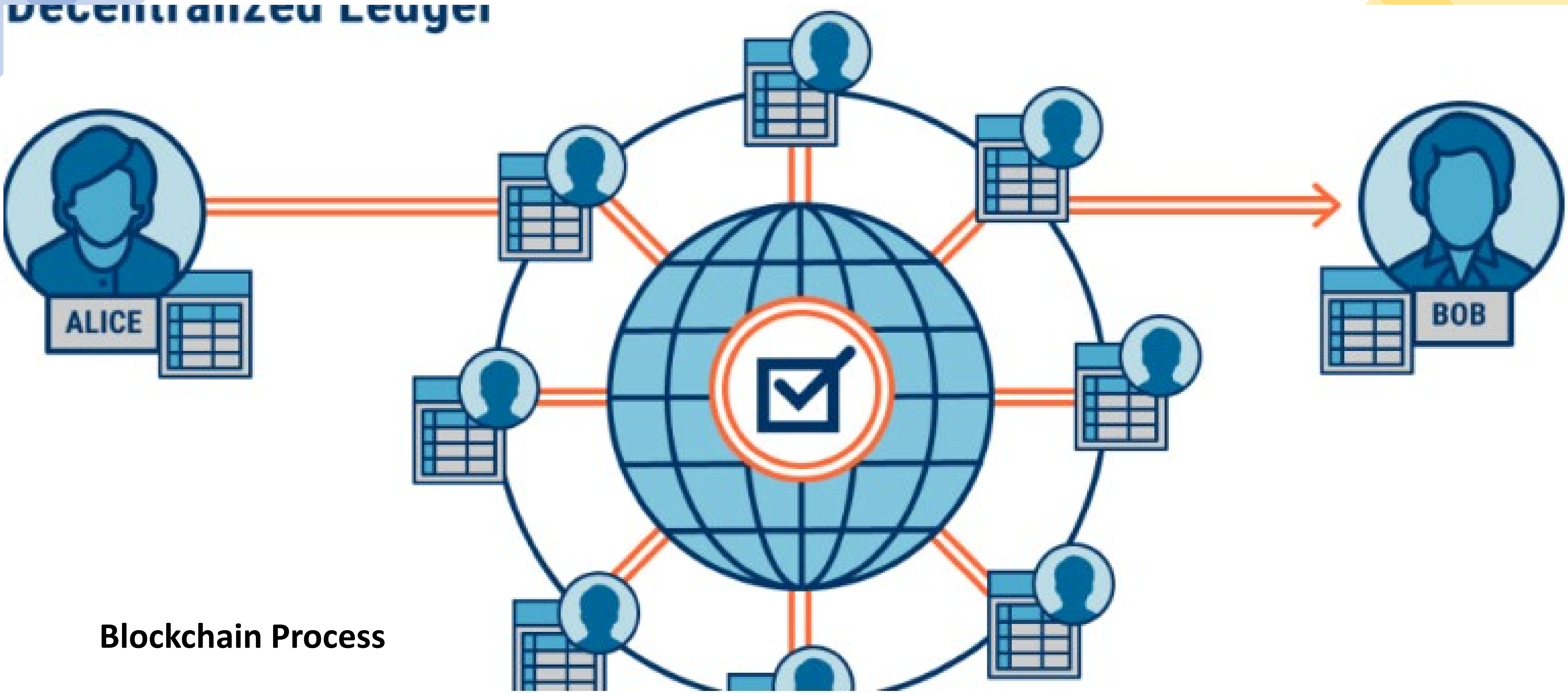


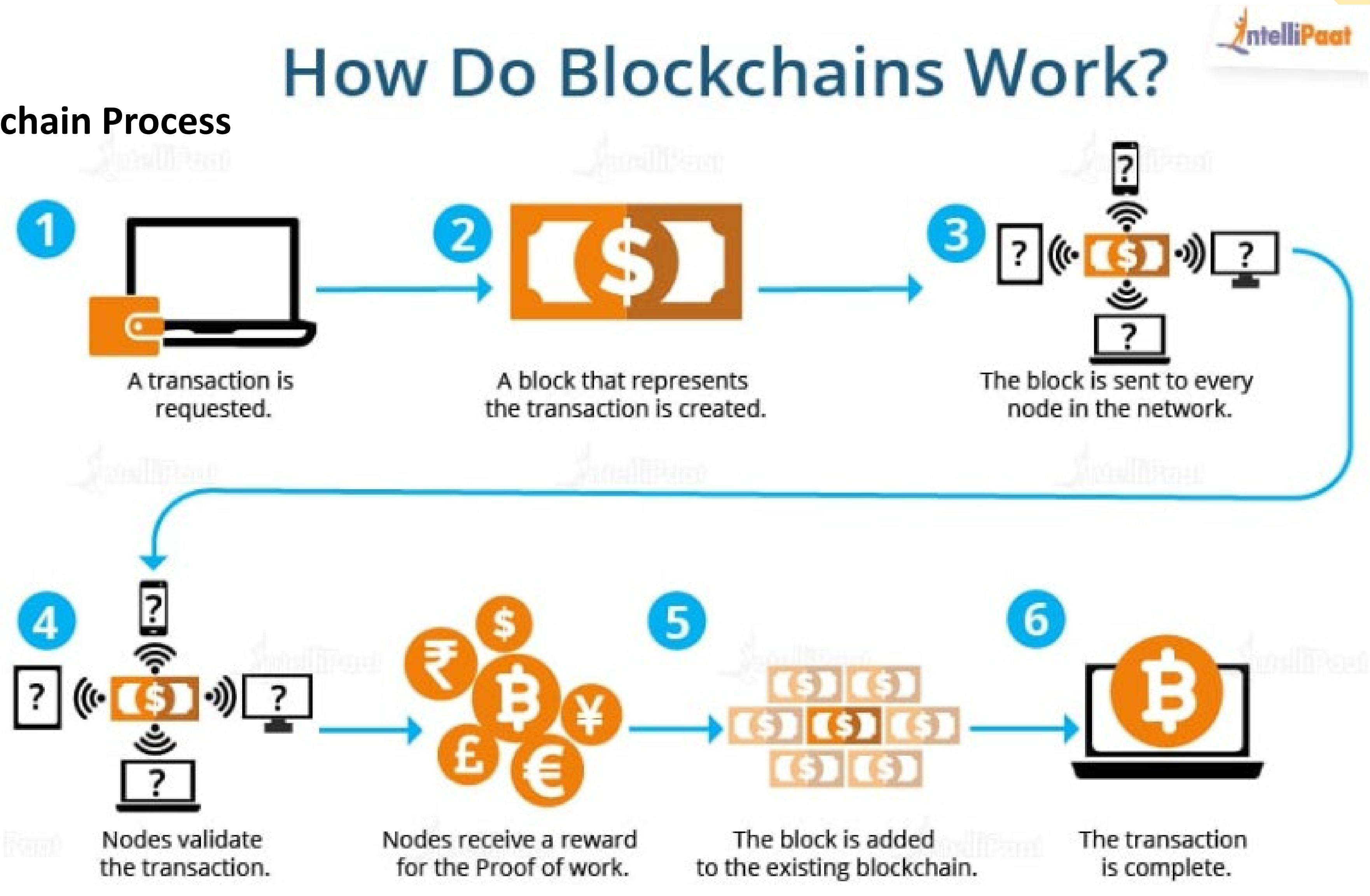
Table 7-2. The structure of the block header

Size	Field	Description
4 bytes	Version	A version number to track software/protocol upgrades
32 bytes	Previous Block Hash	A reference to the hash of the previous (parent) block in the chain
32 bytes	Merkle Root	A hash of the root of the merkle tree of this block's transactions
4 bytes	Timestamp	The approximate creation time of this block (seconds from Unix Epoch)
4 bytes	Difficulty Target	The proof-of-work algorithm difficulty target for this block
4 bytes	Nonce	A counter used for the proof-of-work algorithm



Blockchain Process

Blockchain Process

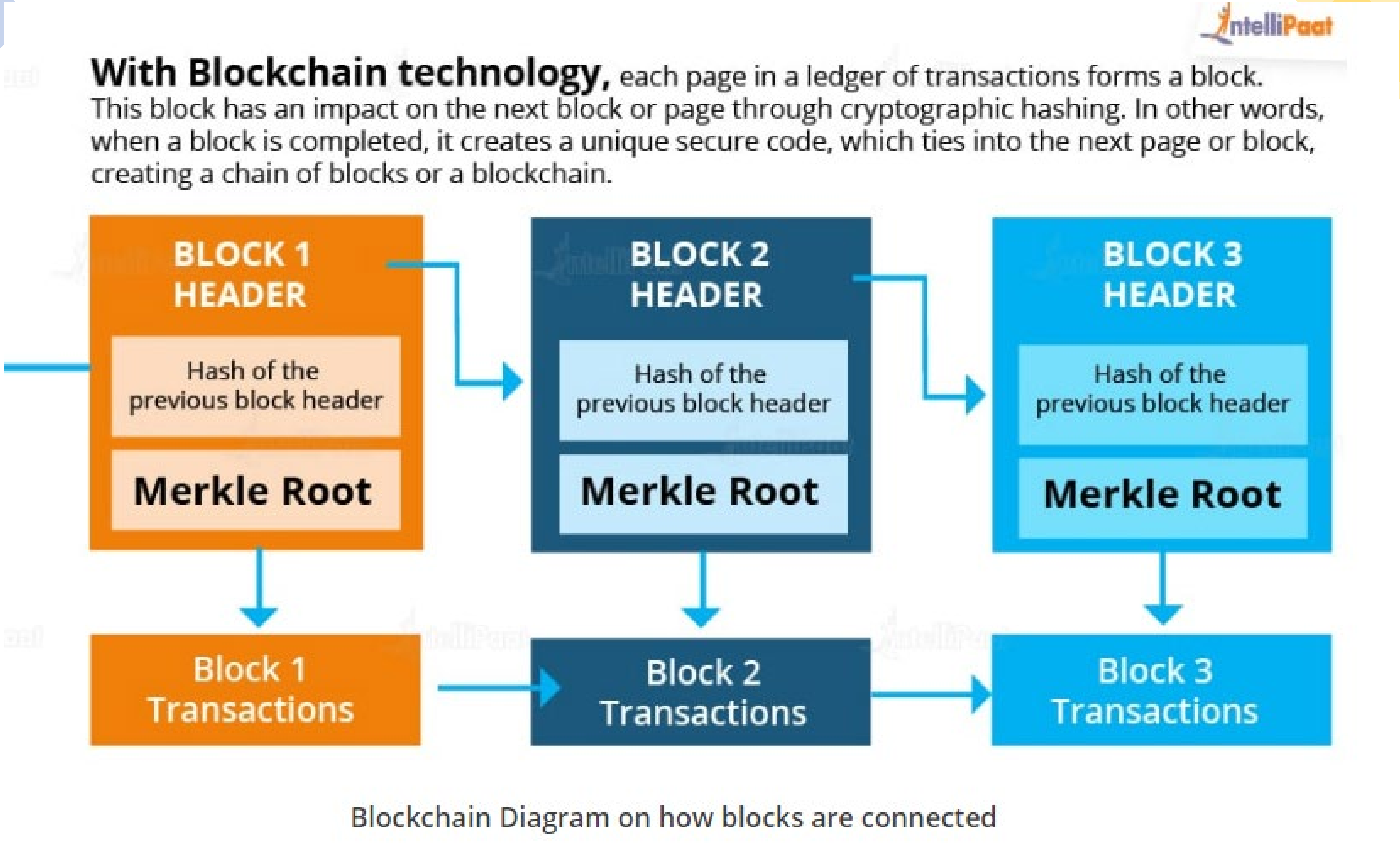


How Do Blockchains Work?

1. A transaction is requested.
2. A block that represents the transaction is created.
3. The block is sent to every node in the network.
4. Nodes validate the transaction.

Nodes receive a reward for the Proof of work.

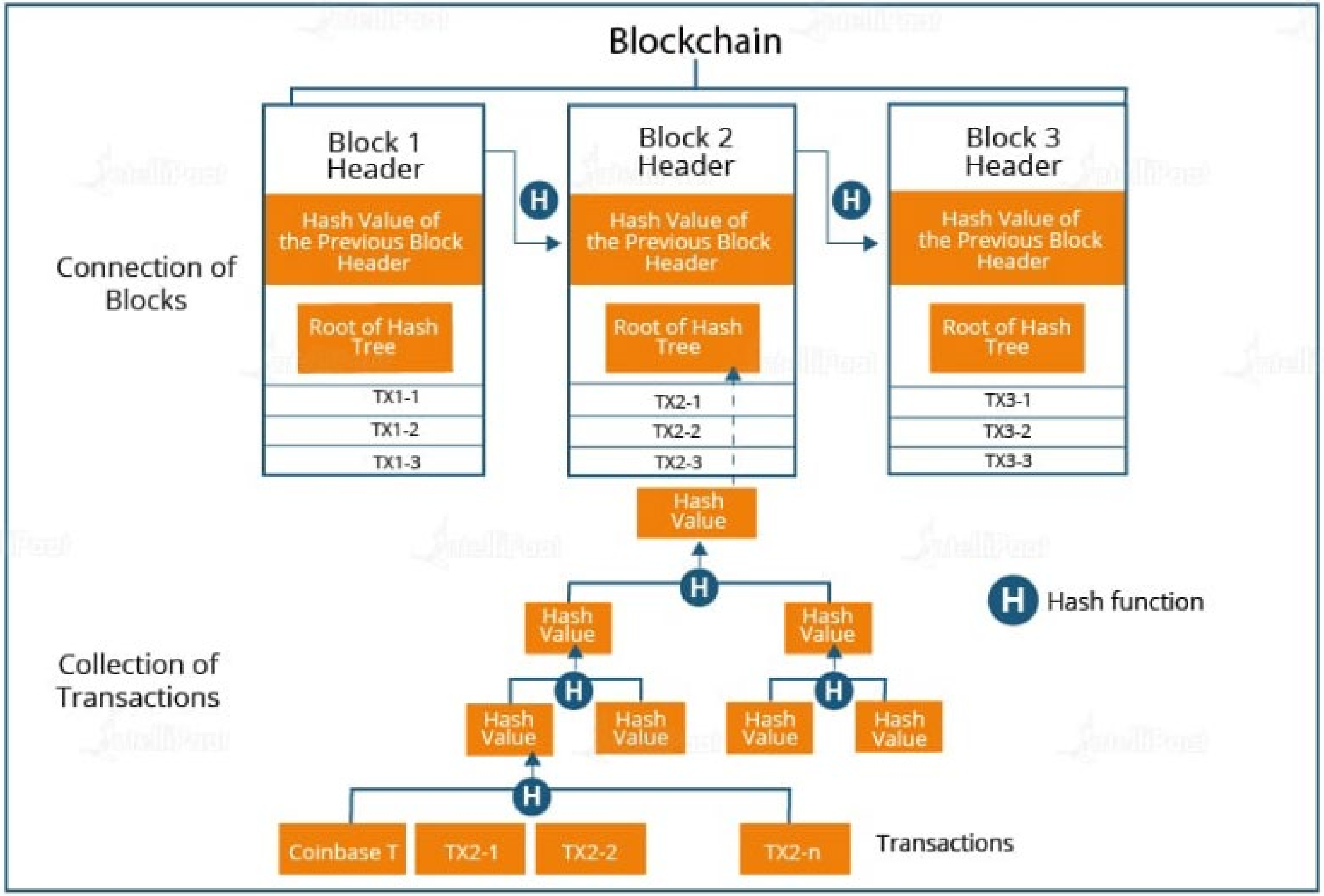
5. The block is added to the existing blockchain.
6. The transaction is complete.



With Blockchain technology, each page in a ledger of transactions forms a block. This block has an impact on the next block or page through cryptographic hashing. In other words, when a block is completed, it creates a unique secure code, which ties into the next page or block, creating a chain of blocks or a blockchain.

Blockchain Diagram on how blocks are connected

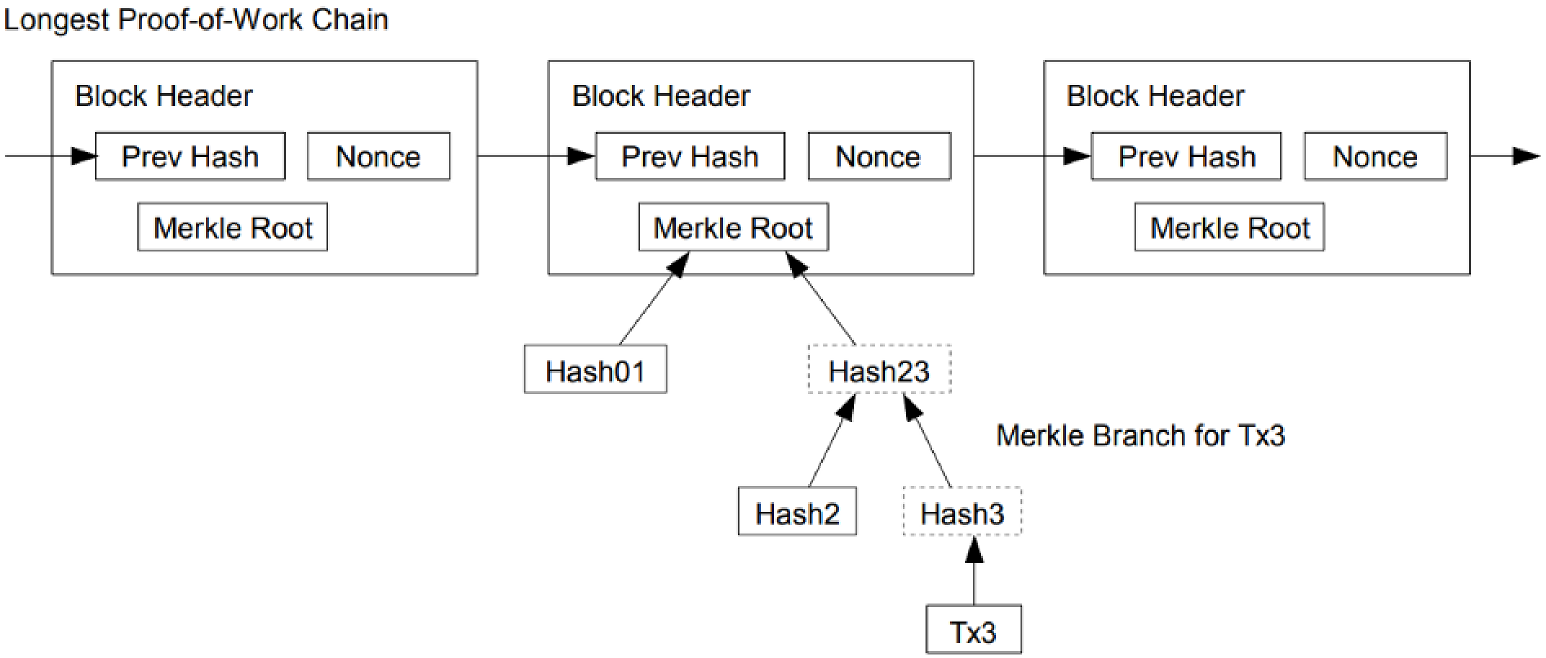
Structure of Blockchain



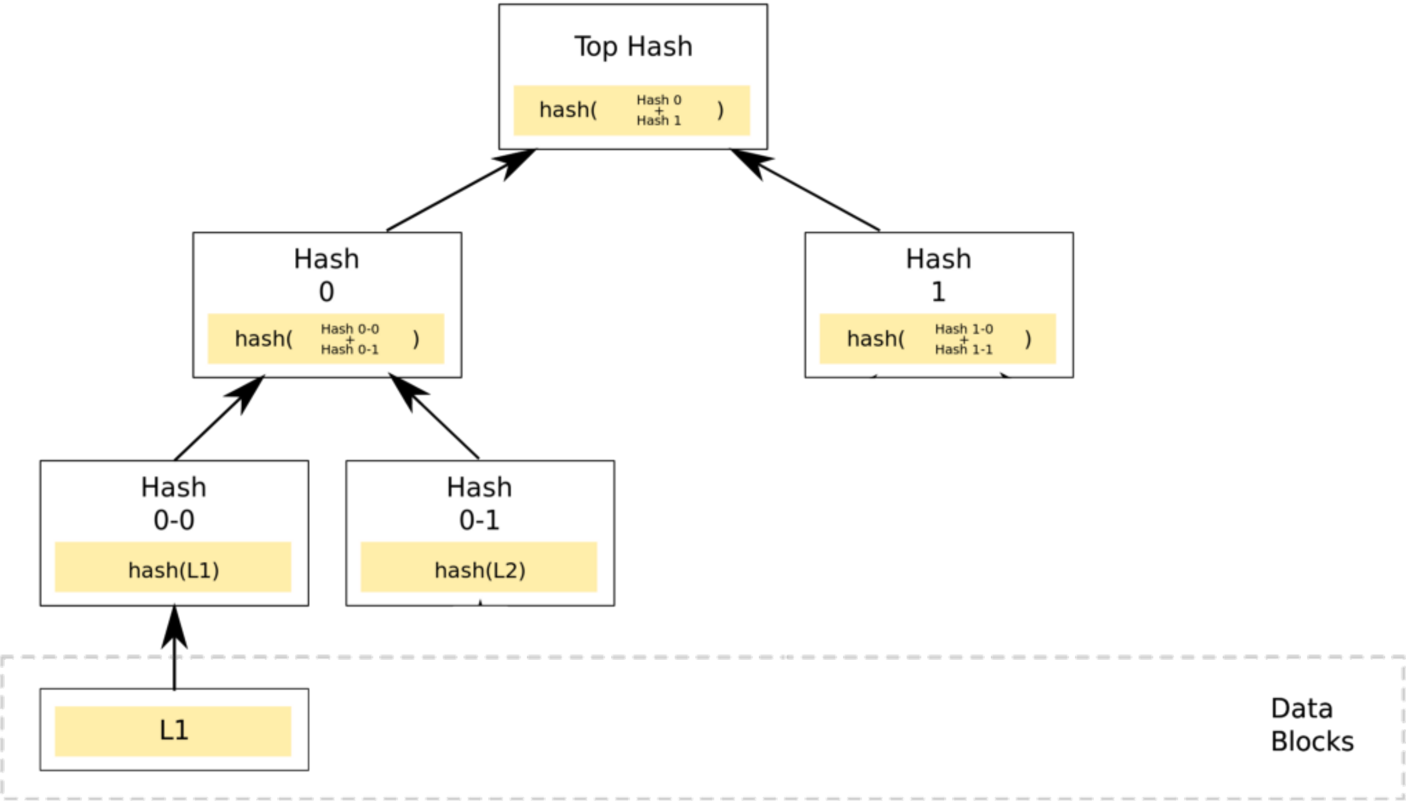
Simplified Payment Verification

- Simple Payment Verification, usually abbreviated to SPV, is a system outlined in the original Bitcoin Whitepaper that enables light clients (wallets running on low-end systems) to verify that a transaction has been included in Bitcoin and therefore a payment has been made
- Now, the cool thing about Merkle trees is that someone that only knows the Merkle root/top hash can verify if a transaction is part of the tree, that is, if it’s been included into a Bitcoin block.
- This is done by taking the nodes that are in the path that connects the Merkle root with one of the bottom transactions and bundling them together to create a proof.

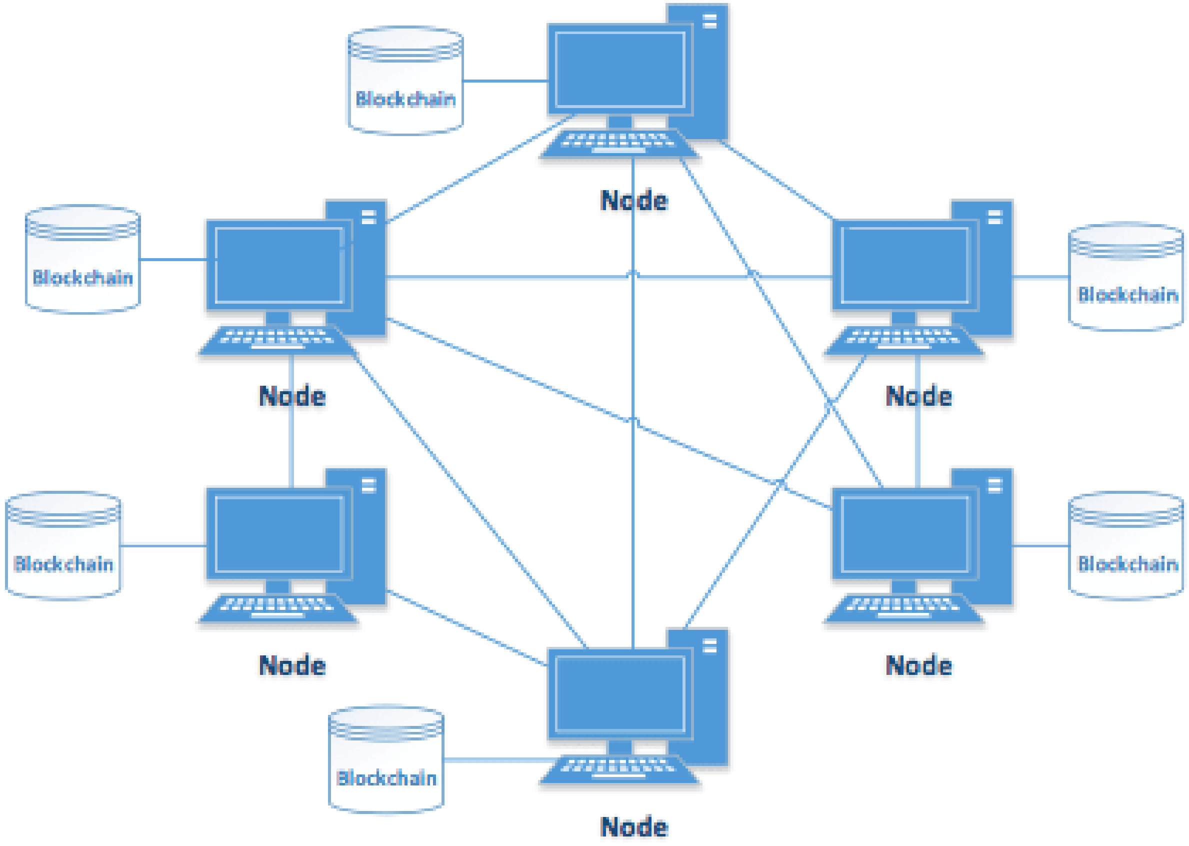
Simplified Payment Verification



Simplified Payment Verification



Blockchain P2P architecture

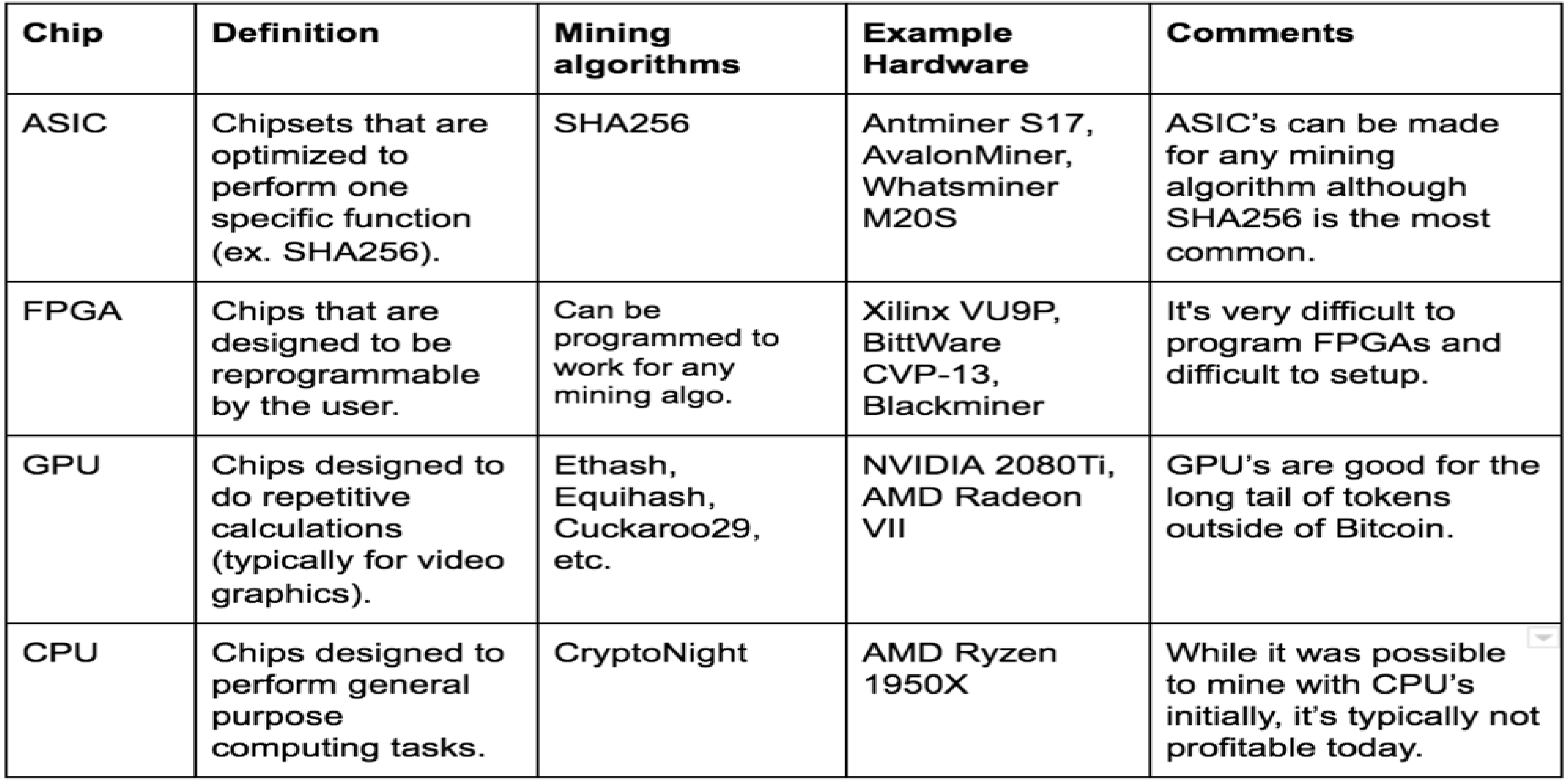


https://www.researchgate.net/publication/320127088_Blockchain_Implementation_Quality_Challenges_A_Literature_Review

P2P architecture

- The term P2P refers to decentralized networks of interconnected computer systems containing peers, or nodes. All nodes are equal, and the exchange of data occurs without a central server — that is, each computer or node can act as both a file server and a client. For example, when acting as a client, a node downloads data from other participants; and when it’s acting as a server, it can be a downloading source.
- Put simply, the peers or participating computer systems can simultaneously consume and provide resources on the same network. These resources can be files, storage, access to a scanner or printer, or processing power. There is no centralized authority, and no single point of failure. All interconnected nodes can engage in storing, distributing and uploading files. Transactions are peer-to-peer — P2P — meaning that they take place directly between the two parties involved, *sans* intermediary.

Hardware required for Mining in Blockchain



Chip	Definition	Mining algorithms	Example Hardware	Comments
ASIC	Chipsets that are optimized to perform one specific function (ex. SHA256).	SHA256	Antminer S17, AvalonMiner, Whatsminer M20S	ASIC's can be made for any mining algorithm although SHA256 is the most common.
FPGA	Chips that are designed to be reprogrammable by the user.	Can be programmed to work for any mining algo.	Xilinx VU9P, BittWare CVP-13, Blackminer	It's very difficult to program FPGAs and difficult to setup.
GPU	Chips designed to do repetitive calculations (typically for video graphics).	Ethash, Equihash, Cuckaroo29, etc.	NVIDIA 2080Ti, AMD Radeon VII	GPU's are good for the long tail of tokens outside of Bitcoin.
CPU	Chips designed to perform general purpose computing tasks.	CryptoNight	AMD Ryzen 1950X	While it was possible to mine with CPU's initially, it's typically not profitable today.

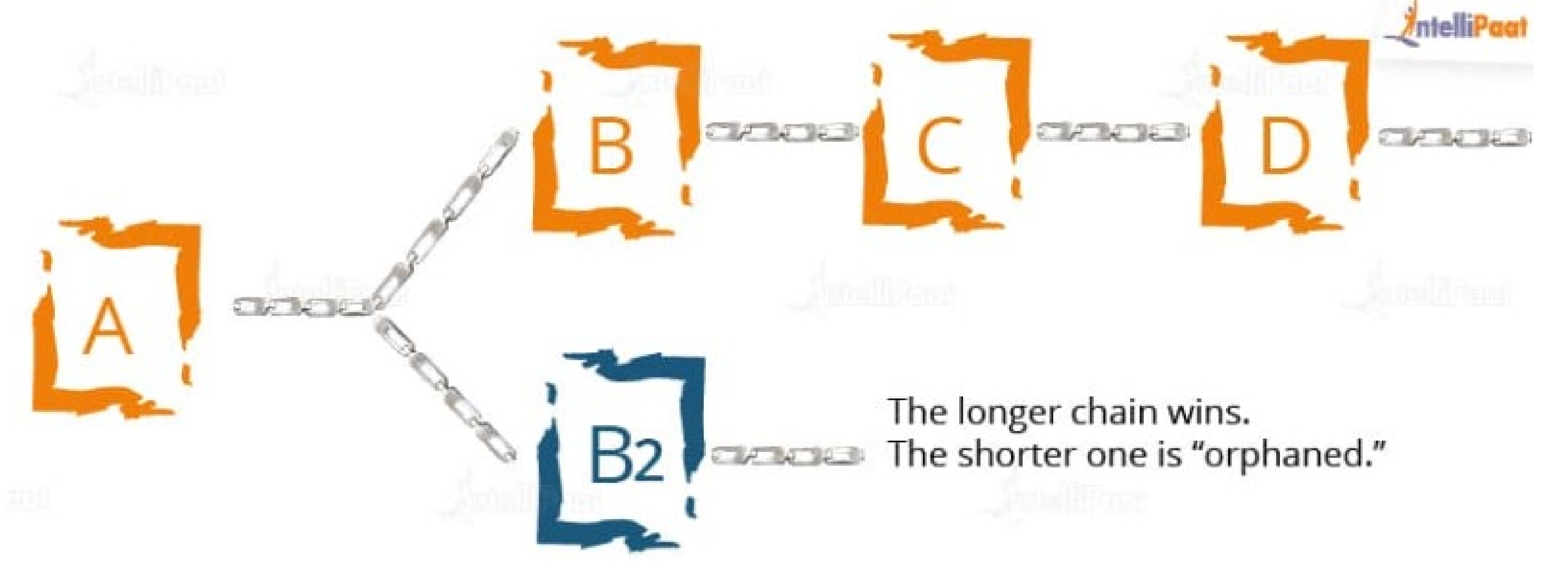
Blockchain Forks

A fork is **a change to the blockchain protocol.** It is essentially a divergence from the previous version of the blockchain.

The decentralized nature of public blockchains means that participants on the network must be able to come to an agreement as to the **shared state** of the blockchain.

The unanimous consensus among the network nodes results in a single blockchain that **contains verified data that the network asserts to be correct.**

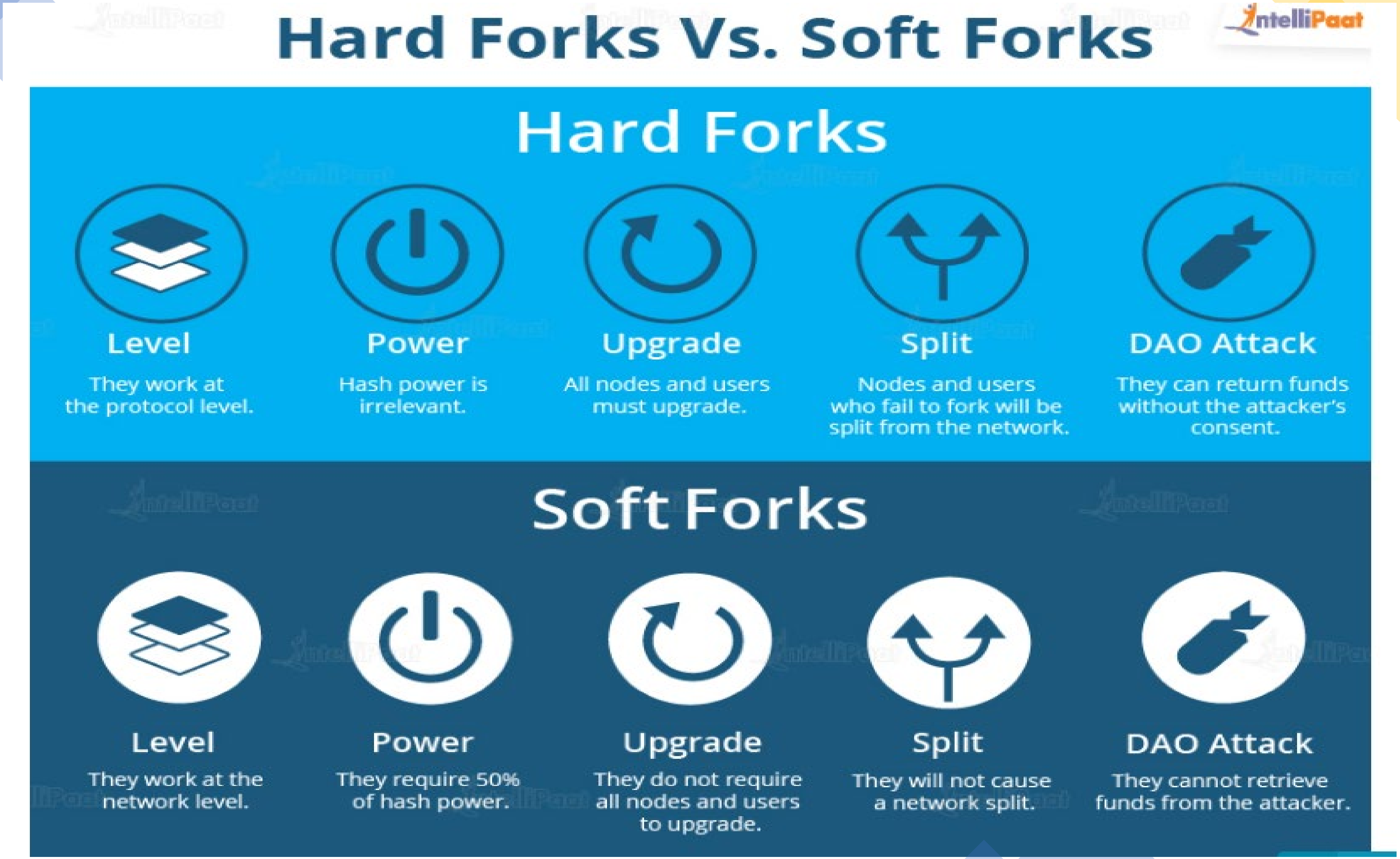
However, many times, nodes in the network can’t come in unanimous consensus regarding the **future state of the blockchain**. This event leads to **forks**, meaning that it leads to a point in which the ideal ‘single’ chain of blocks is split into two or more chains which are all valid.



The longer chain wins.
The shorter one is “orphaned.”

Reasons for the occurrence of a blockchain fork

- Adding new functionality
- Fixing security issues
- Reversing transactions



Hard Forks Vs. Soft Forks

Hard Forks

Level
They work at the protocol level.

Power
Hash power is irrelevant.

Upgrade
All nodes and users must upgrade.

Split
Nodes and users who fail to fork will be split from the network.

DAO Attack
They can return funds without the attacker's consent.

Soft Forks

Level
They work at the network level.

Power
They require 50% of hash power.

Upgrade
They do not require all nodes and users to upgrade.

Split
They will not cause a network split.

DAO Attack
They cannot retrieve funds from the attacker.

Hard Forks

- ❑ When there is a change in the software that runs on full nodes to function as a network participant, new blocks mined based on new rules in the blockchain protocol are not considered valid by the old version of the software.
- ❑ When hard forks occur, **new currency comes to existence.** An equivalent quantity of currency is distributed to the full nodes who choose to upgrade their software so that no material loss occurs.
- ❑ **The final decision to join with which chain rests with the full nodes.** If full nodes choose to join with the new chain, the software is upgraded to make newer transactions valid while the nodes who do not choose to upgrade their software continue to work the way they used to work.

Example:
Suppose, there is a new update in the Ethereum blockchain in which the consensus protocol will change from a type of Proof of Work to a type of Proof of Stake. The full nodes which install the update will use the new consensus protocol, and the ones who do not choose to install the update will become incompatible in the blockchain.

Soft Forks

❑When there is a change in the software that runs on full nodes to function as a network participant, new blocks are mined **based on new rules in the blockchain protocol** and are also **considered valid by the old version of the software**. This feature is also called backward compatibility.

Example:
Suppose, there is a new update in the Ethereum blockchain in which the consensus protocol will change from a type of Proof of Work to a type of Proof of Stake. The full nodes which install the update will use the new consensus protocol, and the ones who do not choose to install the update will still stay compatible with other nodes in the blockchain.

Blockchain Parameters

Permission restrictions

Permission restrictions decide whether transaction processors (miners) who submit data and are eligible to create blocks of data can do so without permission or are restricted to do so and need permission from a central authority. Two models exist:

(1) **Permissioned blockchains**: Transaction processing is performed by predefined users. This is the case for example in Hyperledger Fabric.

(2) **Permissionless blockchains**: There are no restrictions on the identities of processors, thus everyone can start mining to create blocks. This is the case in Bitcoin and Ethereum.

Blockchain Parameters

Restricted access to data

This refers to who can view (read) transaction data from the blockchain network. Two models exist:

(1) **Public blockchains**: There are no restrictions on reading transaction data. Everyone can download the blockchain ledger and view all transactions. This is the case with Bitcoin and Ethereum.

(2) **Private blockchains**: Direct access to blockchain data is limited to predefined users. Thus, only participants that are registered on the blockchain network can download the ledger. This is the case with Hyperledger Fabric.

Blockchain Parameters

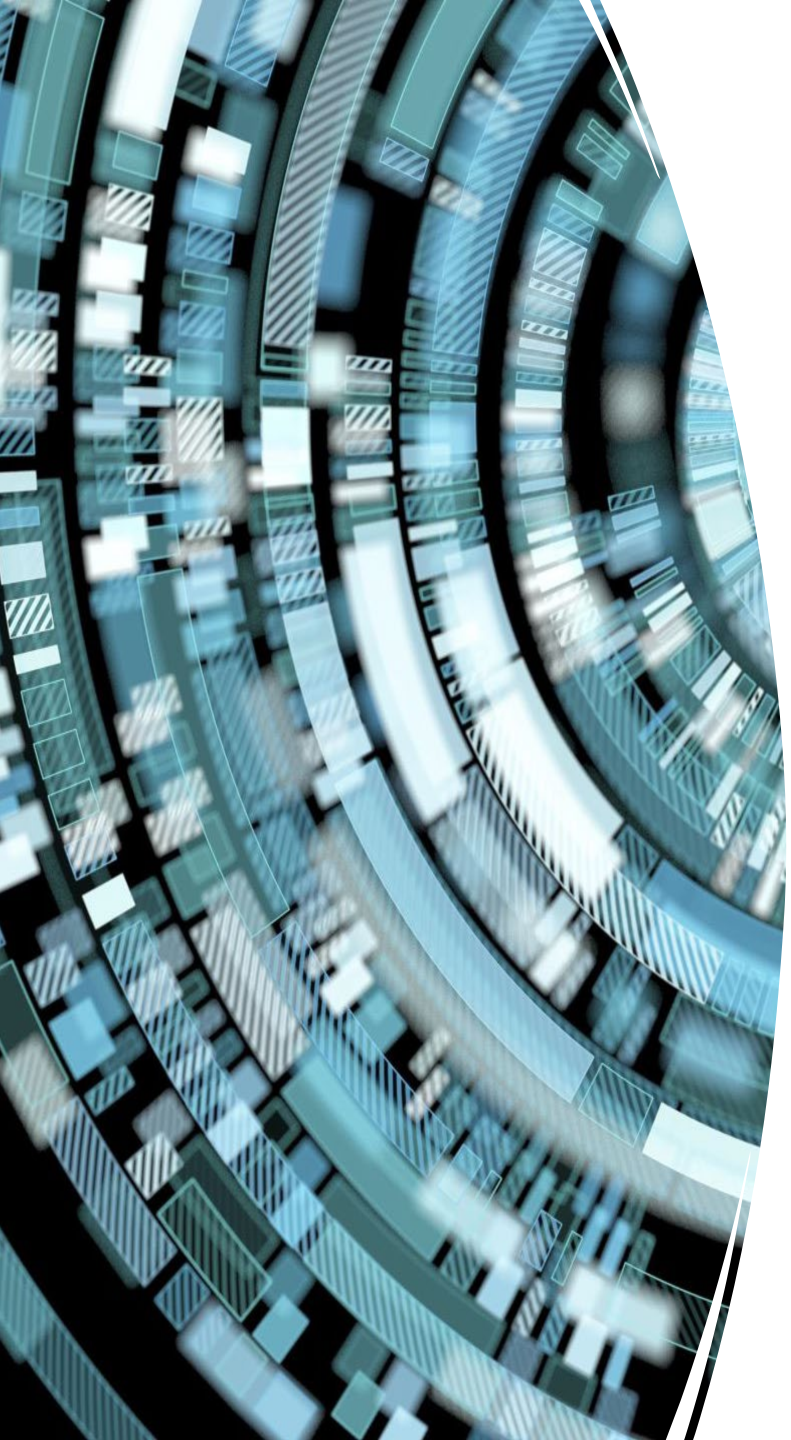
The consensus mechanism used

The consensus mechanism is a means to determine consensus about all transactions and the current state of the system. The mechanism ensures that transactions will only be added to the blockchain if valid and never recorded more than once. Three models dominate currently:

(1) **Proof-of-Work (PoW)**: Miners have to solve a computational difficult problem to ensure the validity of new transactions. (Bitcoin/Ethereum)

(2) **Proof-of-Stake**: Miners can create a new block depending on their investment and ownership to the system. This will be explained in more detail when talking about Ethereum later.

(3) **Practical Byzantine Fault Tolerance (PBFT)**: This consensus mechanism is one of many that can be used in permissioned blockchains, where a new block is added if more than 2/3 of all validating peers submit the same response. Hyperledger Fabric out of the box does not provide PBFT, but offers its users to add this consensus mechanism modularly.



Blockchain Parameters

Scalability

Scalability of blockchain systems is composed of two factors:

(1) **Node-scalability** in blockchain networks refers to the extent to which the network can add more participants without a loss in performance.

(2) **Performance scalability** refers to the number of transactions processed per second impacted by the latency between transactions and each block size.

A blockchain is considered scalable if it can add thousands of globally distributed nodes while still processing thousands of transactions per second. Currently, none of the existing blockchains are really scalable

Blockchain Parameters

Governance

Governance refers to the degree to which decision making power is distributed in the blockchain community. It tries to answer the question of who can make what decisions on a blockchain platform.

As you can imagine, every blockchain platform needs to be developed and maintained. Usually a core developer team performs this job. As there are many stakeholders in a blockchain network (core developers, miners, currency-exchanges, Dapp developers), decision making for new changes to the blockchain core protocol is very important and often controversial.

This is a strong factor where blockchain platforms differ from each other and will be important for each of the three blockchain platforms.

Blockchain Parameters

Anonymity

Anonymity on the blockchain refers to whether the identity of a user is openly transparent.

In public permissionless blockchains, such as Bitcoin and Ethereum, users are pseudonymous because they hide their identity behind a pseudonym, their public wallet address. In private permissioned blockchains, such as Hyperledger Fabric, users usually know each other.

Native Currency

Native currency refers to whether the blockchain has an inherent currency. For example, Bitcoin uses its currency “bitcoin” as medium for exchange. Ethereum uses “ether”.

But Hyperledger Fabric does not use an own currency.

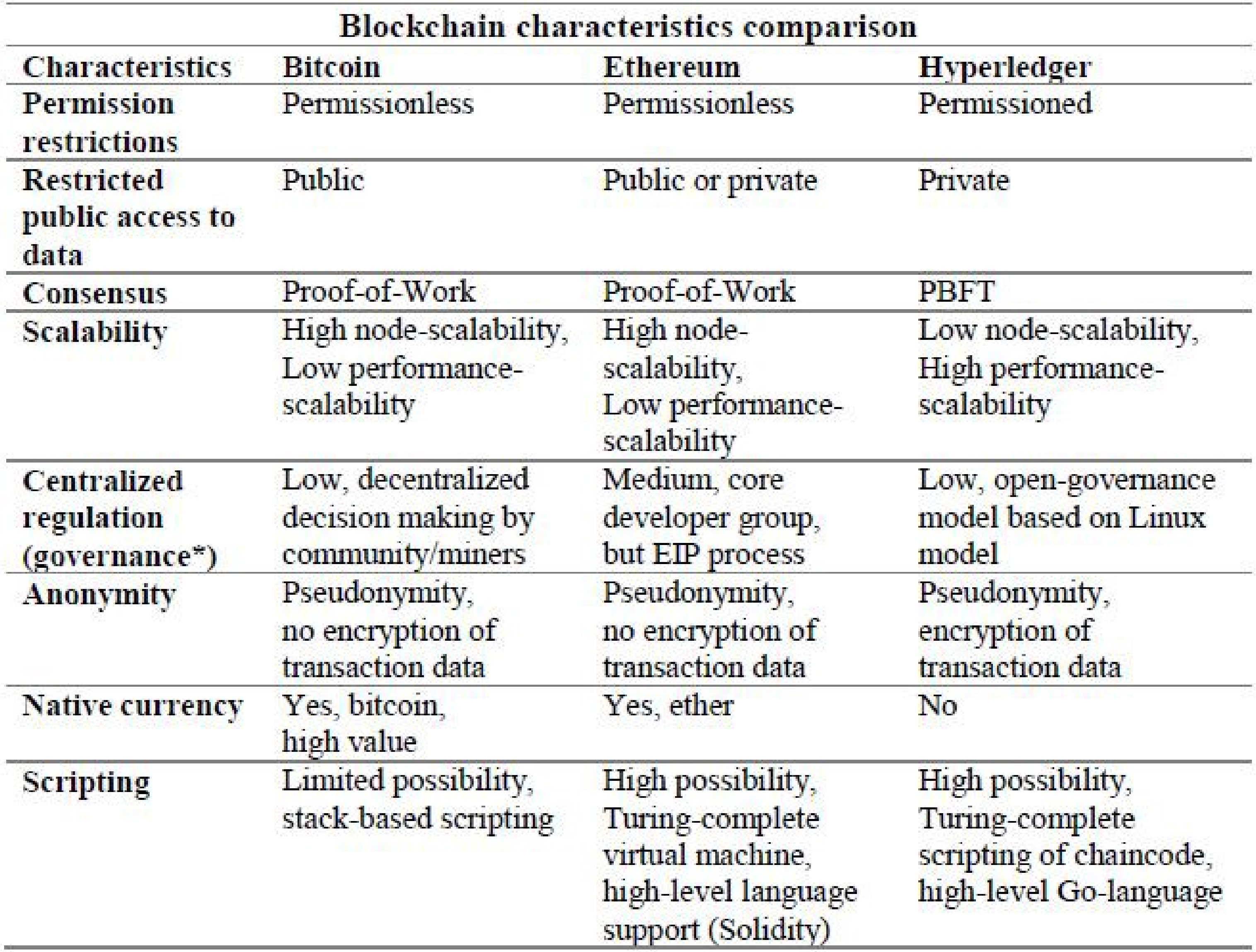
Blockchain Parameters

Scripting

Scripting refers to the degree to which a blockchain's programming features support the development of decentralized applications (Dapps).

Some blockchains such as Ethereum and Hyperledger Fabric provide developers with a Turing-complete scripting language. That way developers can create smart-contracts that can interact with each other and form decentralized applications.

Other blockchains, such as Bitcoin, only provide a very limited stack-based programming possibility. This makes application development very difficult and sometimes not viable.



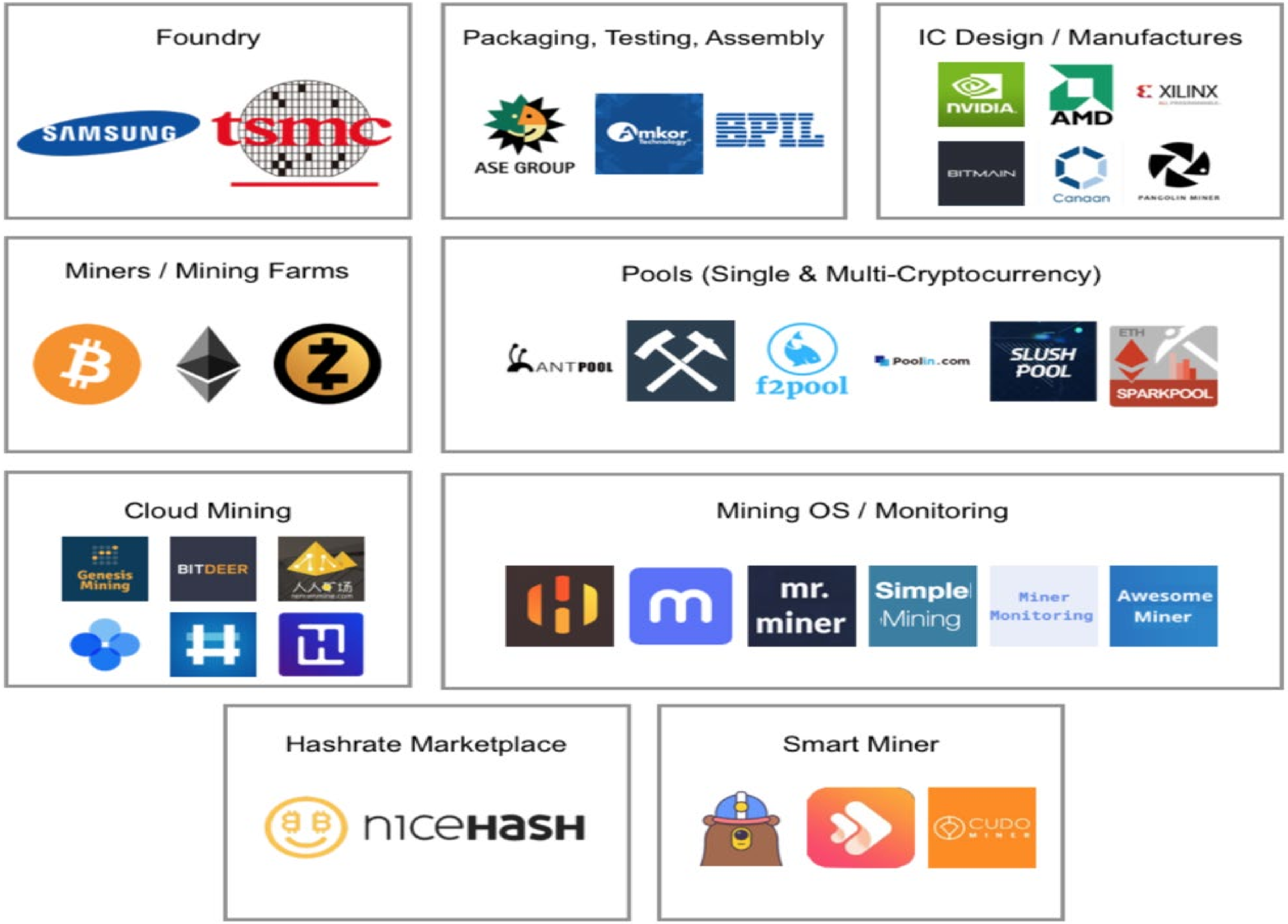
Blockchain characteristics comparison

Characteristics	Bitcoin	Ethereum	Hyperledger
Permission restrictions	Permissionless	Permissionless	Permissioned
Restricted public access to data	Public	Public or private	Private
Consensus	Proof-of-Work	Proof-of-Work	PBFT
Scalability	High node-scalability, Low performance-scalability	High node-scalability, Low performance-scalability	Low node-scalability, High performance-scalability
Centralized regulation (governance*)	Low, decentralized decision making by community/miners	Medium, core developer group, but EIP process	Low, open-governance model based on Linux model
Anonymity	Pseudonymity, no encryption of transaction data	Pseudonymity, no encryption of transaction data	Pseudonymity, encryption of transaction data
Native currency	Yes, bitcoin, high value	Yes, ether	No
Scripting	Limited possibility, stack-based scripting	High possibility, Turing-complete virtual machine, high-level language support (Solidity)	High possibility, Turing-complete scripting of chaincode, high-level Go-language

Characteristic	Ethereum	Hyperledger
Description of platform	– Generic blockchain platform	– Modular blockchain platform
Governance	– Ethereum developers	– Linux Foundation
Mode of operation	– Permissionless, public or private[4]	– Permissioned, private
Consensus	– Mining based on proof-of-work (PoW) – Ledger level	– Broad understanding of consensus that allows multiple approaches – Transaction level
Smart contracts	– Smart contract code (e.g., Solidity)	– Smart contract code (e.g., Go, Java)
Currency	– Ether – Tokens via smart contract	– None – Currency and tokens via chaincode

Characteristics	Ethereum	Hyperledger Fabric	R3 Corda
Programming Language	Solidity	Go, Java	Kotlin
Governance	Distributed among all participants	Linux foundation and organisation in the Chain	R3 and organisations involved.
Smart Contract	Not legally bounded	Not legally bounded	Legally bounded
Consensus Algorithm	PoW. Casper implementation PoS.	PBFT	Notary nodes can run several consensus algorithm
Scalability	Existing scalability issue	Not prevalent	Not prevalent
Privacy	Existing privacy issue	Not prevalent	Not prevalent
Currency	Ether	None Can be made using chaincode	None

Cryptocurrency Mining Ecosystem



Foundry

Packaging, Testing, Assembly

IC Design / Manufactures

Miners / Mining Farms

Pools (Single & Multi-Cryptocurrency)

Cloud Mining

Mining OS / Monitoring

Hashrate Marketplace

Smart Miner

Proof of Capital