GROUND UP SERIES

BLOCKCHAIN

using JavaScript python Java php

Syed Awase Khirni

RESEARCHER | ENTREPRENEUR | TECHNOLOGY COACH

@sak008 | sak@sycliq.com/sak@territorialprescience.com | +91. 9035433124

Syed Awase earned his PhD from University of Zurich in GIS, supported by EU V Framework Scholarship from SPIRIT Project (www.geo-spirit.org). He currently provides consulting services through his startup www.territorialprescience.com and www.sycliq.com . He empowers the ecosystem by sharing his technical skills worldwide, since 2008. He has envisioned, conceptualized, implemented and delivered many code play sessions(code camps) with various MNC to upskill their workforce.

Terms of Use

- You shall not circulate these slides without written permission from Territorial Prescience Research I Pvt ltd.
- If you use any material, graphics or code or notes from these slides, you shall seek written permission from TPRI and acknowledge the author Dr. Syed Awase Khirni
- If you have not received this material, post-training session, you shall destroy it immediately and not use it for unauthorized usage of the material. If any of the material, that has been shared is further used for any unauthorized training by the recipient, he shall be liable to be prosecuted for the damages. Any supporting material that has been provided by the author, shall not be used directly or indirectly without permission.
- If this material, has been shared to any organization prior to the training and the organization does not award the contract to TPRI, it should not use the training material internally. If by any chance, the organization is using this training material without written permission, the organization is liable to pay for the damages to TPRI and is subjected to legal action, jurisdiction being Bangalore.
- Without unauthorized usage, TPRI has right to claim damages ranging from USD 50000 to USD 10,0000 dollars as damages.
- Any organization, which does not intend to go ahead with training or does not agree with the terms and conditions, should destroy the material from its network immediately. The burden of proof lies on the client, with whom this material has been shared.
- Recovery of the damages and legal fees will be born by the client organization/candidate/party, which has violated the terms and conditions.
- Only candidates who have attended the training session in person from Dr. Syed Awase Khirni, TPRI are entitled to hold this training material. They cannot further circulate it, or use it or morph it, or change it to provide trainings.
- TPRI reserves all the rights to this material and code plays and right to modify them as and when it deems fit.
- If you agree with the terms and conditions, please go ahead with using the training material. Else please close and destroy the slide and inform TPRI immediately

Please read terms and conditions of use

Original Series

Slide Version Updates

Last Updated	Version	Release Date	Updated by	Code Plays Done @

Program Agenda

DAY 1	DAY 2	DAY 3	DAY 4

DAY 5	DAY 6	DAY 7	DAY 8

Operating features

Access on-boarding

Performance/scalability

cryptography

use cases

tokenization

privacy

security

Consensus algorithms

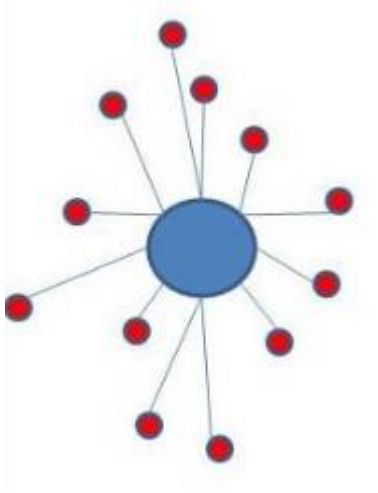
Risks

SYED AWASE KHIRNI

BLOCKCHAIN INTRODUCTION

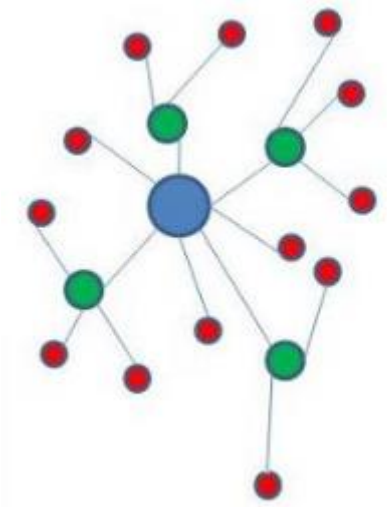
BITCOIN :2008 BITCOIN IMPLEMENTATION:2009

Centralized



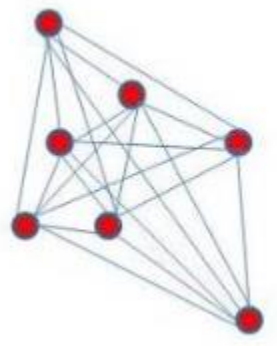
A single authority or server and all the other nodes act like clients or entities who accept message and enact accordingly.

Decentralized



There are multiple servers who receive messages from one central server. The individual nodes are connected to the secondary servers. All the servers can be of equal status in hierarchy with no central server as well.

Distributed



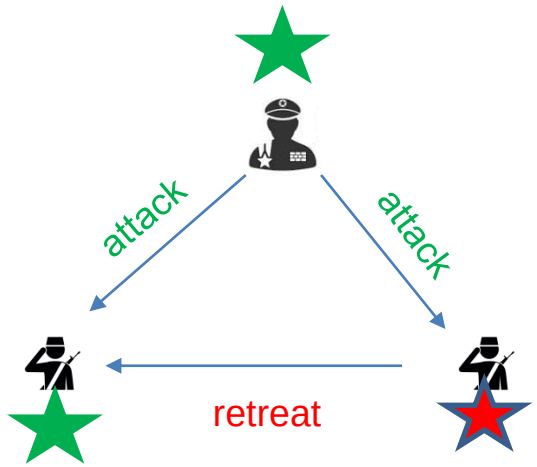
No central authority. Each node is connected to every other node and has the exact same authority. Each node may have varied processing power and memory.

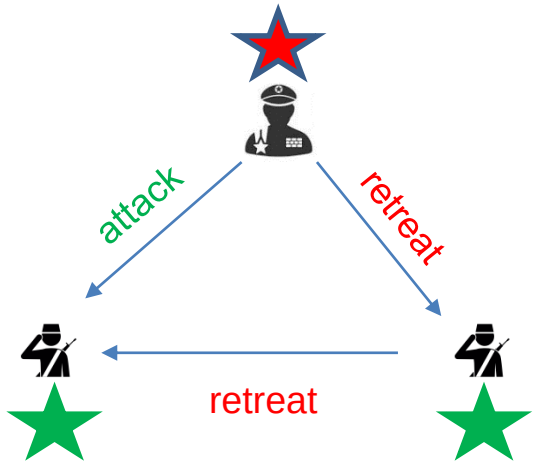
Distributed Systems

- A computing paradigm where by two or more **nodes** work with each other in a coordinated fashion to achieve a common outcome.

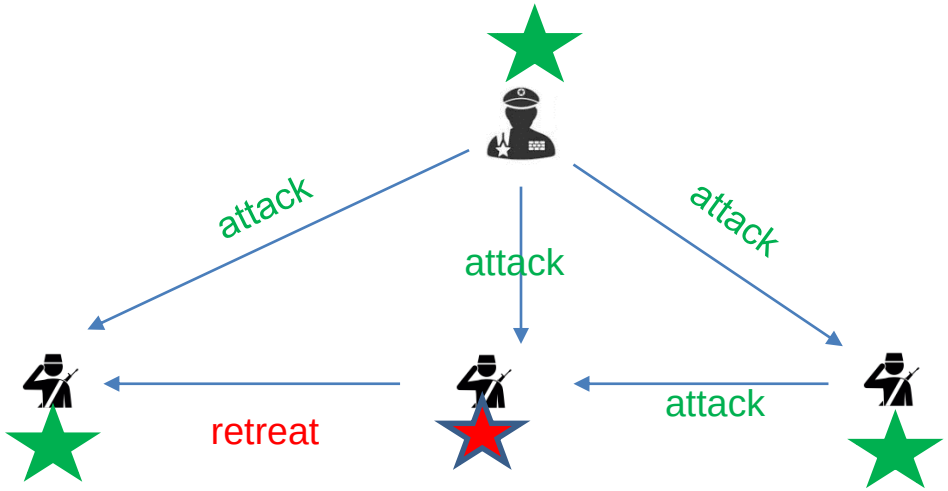
- A node is an individual player in a distributed system.
- All nodes are capable of sending and receiving messages to and from each other.
- A node can be characterized as honest, faulty, malicious
- A node has a memory and a processor to compute.

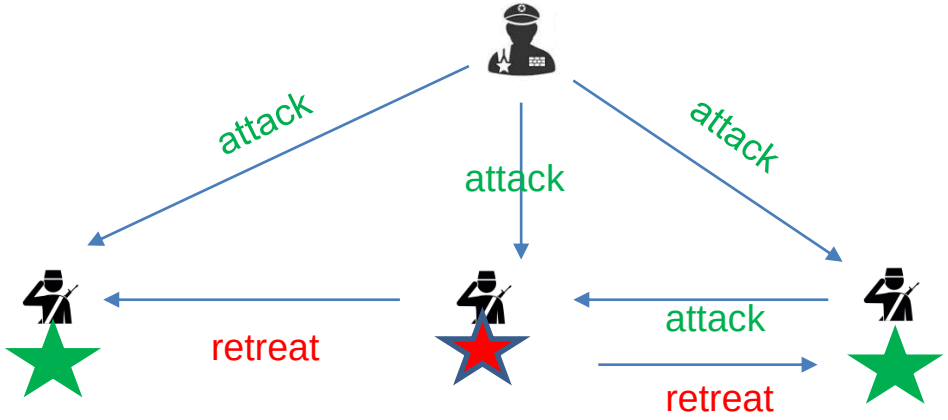
Byzantine/Roman Generals Problem





Traitor Lieutenant / **"Byzantine node"**





Practical Byzantine Fault Tolerance (PBFT)

- Castro and Liskov (1999)
- Consensus based signed content communication
- A consensus is reached after a certain number of messages are received containing the same signed content.
- Used to avoid inconsistent behavior of Byzantine nodes which can be intentionally malicious and can be detrimental to the operation of the network.
- A malicious node is called **"Byzantine node"**

challenges of distributed system design

Coordination	Communication	Fault Tolerance

ELECTRONIC CASH

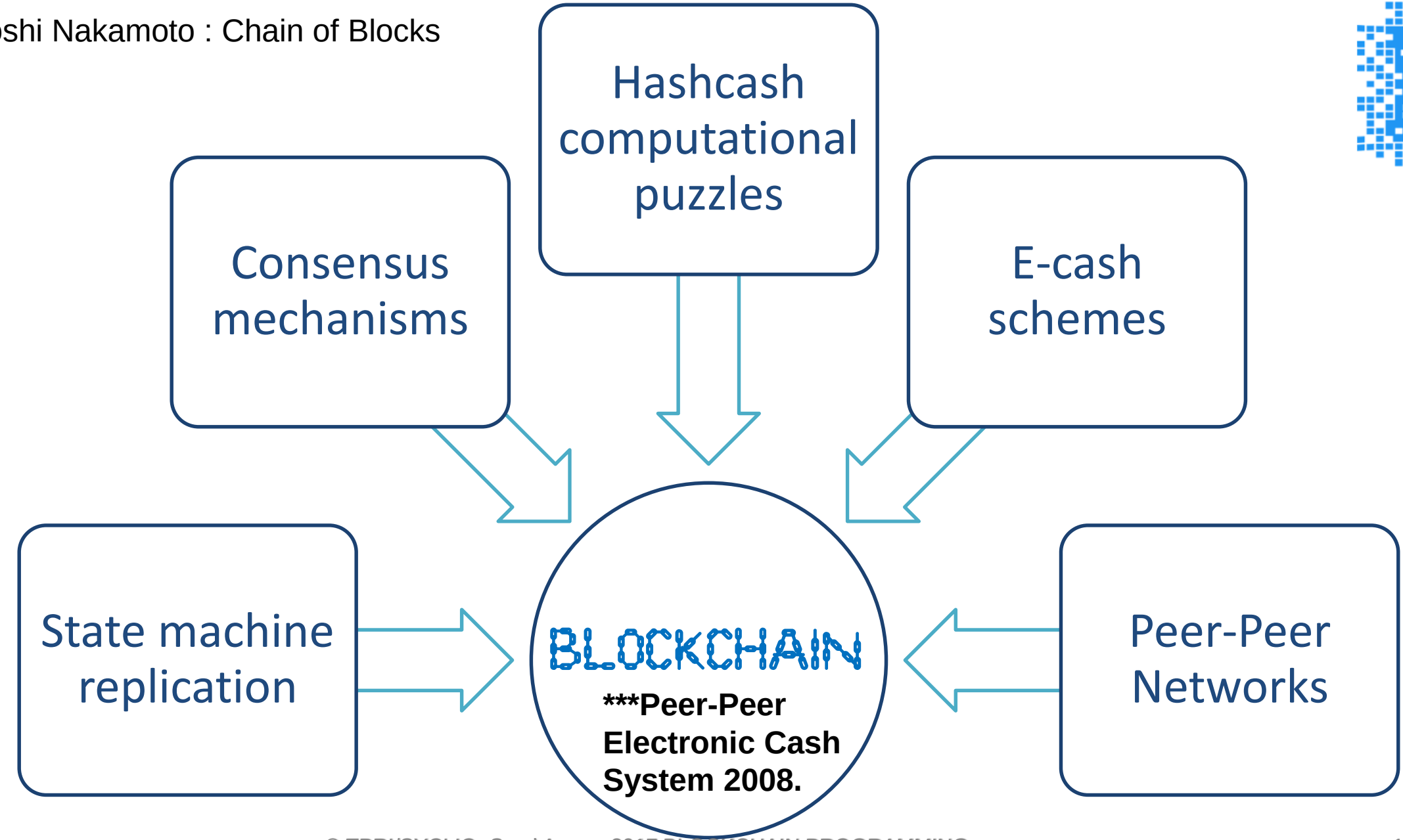
- Proposed by David Chaum in 1980's
- E-Cash systems demanded to address to fundamental challenges
  - Accountability
  - Annonymity

- Accountability: Available ledger records stating the amount of spendable cash by the rightful owner.
  - Double spend problem arises when the same money can be spent twice, As it is easy to make copies of digital data.
- Anonymity: to protect user's privacy, traceability of spending is possible in digital transactions.

ELECTRONIC CASH :David Chaum

- Blind Signatures: used to sign a document without actually seeing it.
- Secret Sharing: a concept that enables the detection of double spending, that is using the same e-cash token twice.

*** Satoshi Nakamoto : Chain of Blocks



BLOCKCHAIN

A peer-to-peer, **distributed ledger** that is cryptographically-secure, append-only, immutable (extremely hard to change) and updateable only via consensus or agreement among peers.

A consensus based cryptographically-secure public data base which store information immutably over a peer-to-peer network.

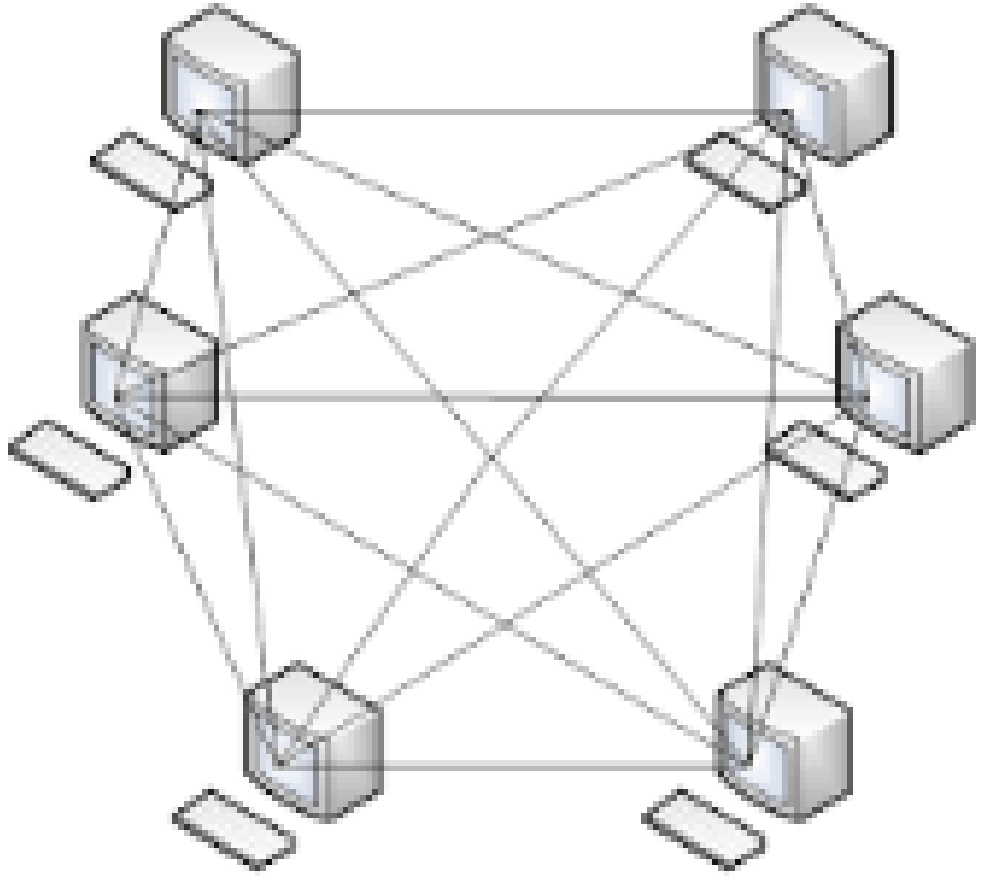
A platform where peers can exchange value/electronic cash using transactions without the need for a centrally trusted arbitrator.

BLOCKCHAIN ATTRIBUTES

- Peer-to-peer networks
- Cryptographically secured distributed ledger
- Append-only
- Updateable via consensus

PEER-TO-PEER

- A network topology where in all peers can communicate with each other and send and receive messages directly.
- In a P2P network, the user utilizes and provides the foundation of the network at the same time, although providing the resources is entirely voluntary.
- A peer makes a portion of computing resources such as disk storage, processing power or network bandwidth, directly available to other participants without the need for any central coordination by server or stable hosts.



DISTRIBUTED LEDGER

- A ledger that is spread across the network among all peers in the network and each peer holds a copy of the complete ledger in an encrypted format.
- A basic requirement is that each of these ledgers in the network has to be cryptographically-secure against any tampering or misuse.

- Some of the services include
  - Non-repudiation
  - Data integrity
  - Data origin
  - Authentication.
- 3 key attributes
  1. Recorded: time stamped information
  2. Transparent: any one can see the ledger of transactions
  3. Decentralized: the ledger exists on multiple computers, often referred to as nodes.

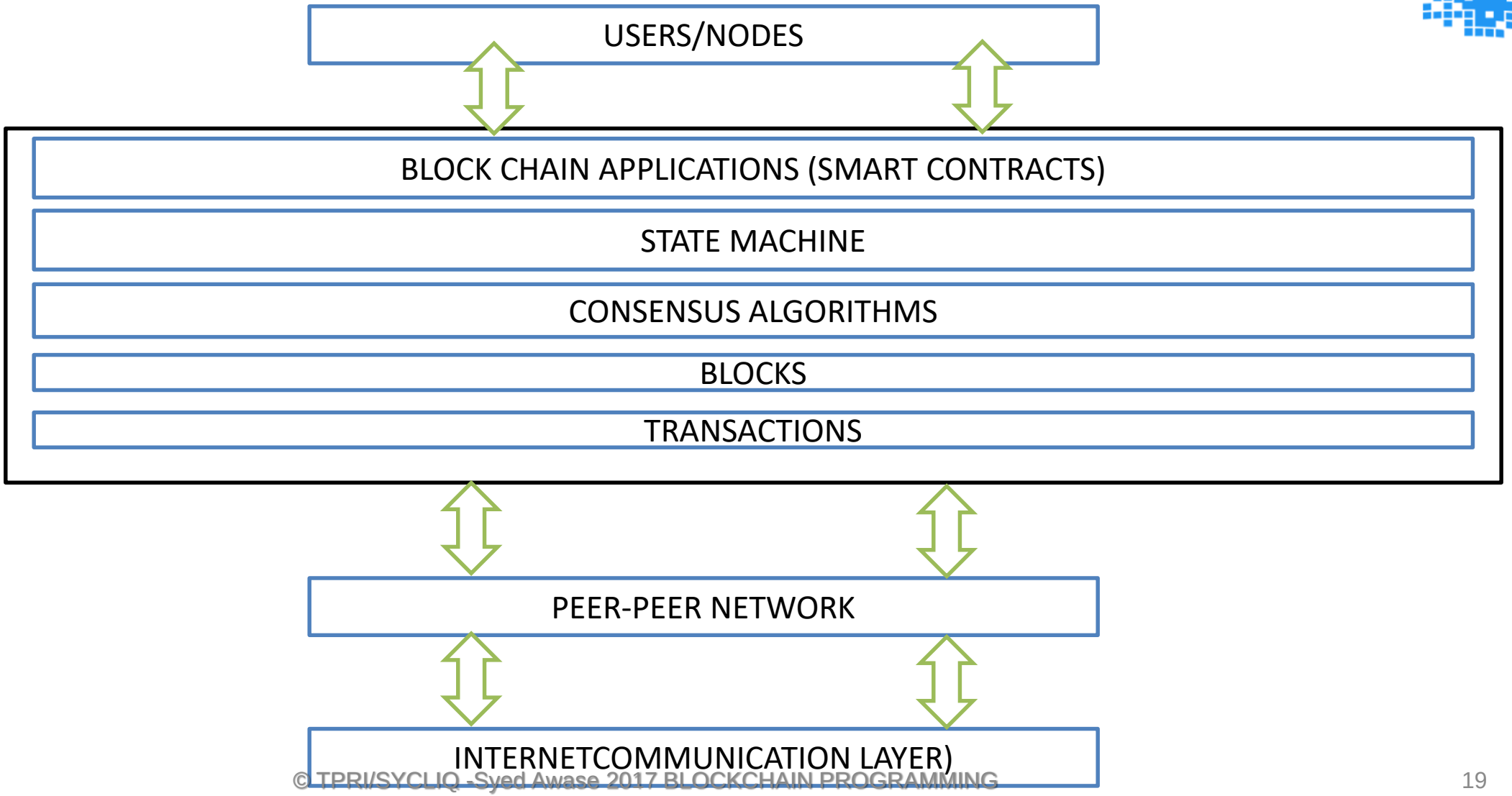
Append-Only

- Blockchain is append-only, which means that the data can only be added to the blockchain in time-ordered sequential order.
- This property implies that once data is added to the blockchain, it is almost impossible to change that data and can be considered practically immutable.
- Data can only change when collusion against the blockchain network succeeds in gaining more than 51% of the power.
- GDPR directives enforce legitimate reasons to change data in the blockchain once it has been added as per *right to be forgotten or right to erasure.*

Updateable via consensus

- It is the most critical attribute of a blockchain: updateable only via consensus, which powers decentralization, where no central authority is in control of updating the ledger.
- Updates made to the blockchain is validated against strict criteria defined by the blockchain protocol and added to the blockchain only after a consensus is reached by all peers participating in the network
- Consensus facilitation algorithms ensure that all parties are in agreement about the final state of the data on the blockchain network and resolutely agree upon it to be true.
- A method of authenticating and validating a value or transaction on a blockchain or a distributed ledger without the need to trust or rely on a central authority. Consensus mechanisms are central to the functioning of any blockchain or distributed ledger.

Blockchain Network



Definitions

- Block: a selection of transactions bundled together and organized logically. A block is made up of transactions and its size varies depending on the type and design of the blockchain in use.
  - A block also has a reference to the previous block
- Transaction: a record of an event. It corresponds to a “leaf” in the merkle tree.
- Genesis Block: the first block in the blockchain that is hardcoded at the time the blockchain was first started.
- Block Header: it is composed of pointer to previous block, the timestamp, nonce, merkle root and the block body that contains transactions.

Definitions

- Nonce: a number that is generated and used only once. A nonce is used extensively in many cryptographic operations to provide replay protection, authentication and encryption.
  - In blockchain, it's used in PoW consensus algorithms and for transaction replay protection.
- Log: synonymous for the merkle tree, a log is the hash tree constructed from the hashed records. Log is represented as a hash tree, the log has a specific property, new entries are always appended as a new leaf to the last leaf in the tree.

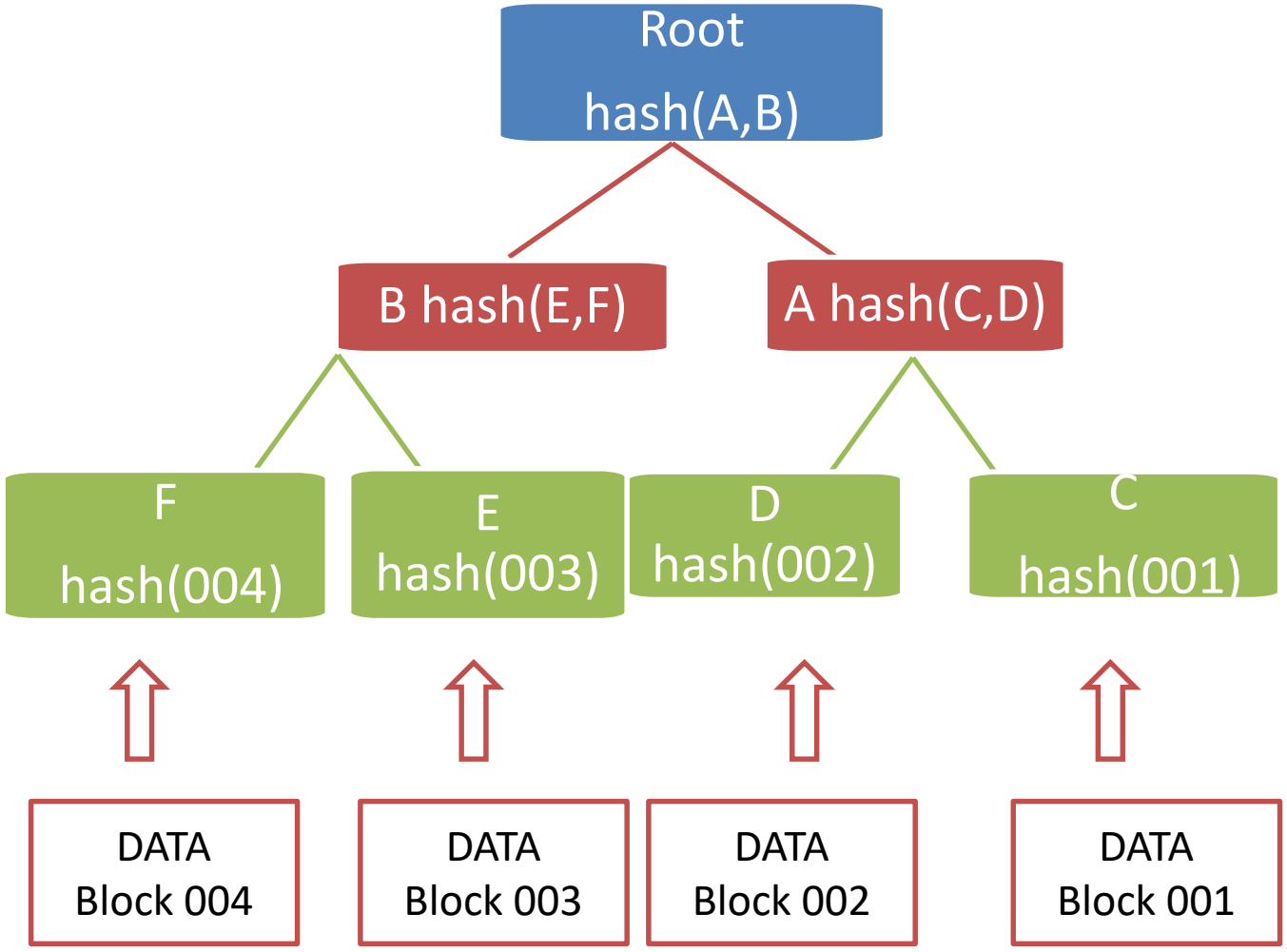
Merkle Root

- Merkle root: a hash of all the nodes of a merkle tree. Merkle trees are widely used to validate the large data structures securely and efficiently. Merkle tree are commonly used to allow efficient verification of transactions.
  - Merkle root in a block chain is present in the block header section of block, which is the hash of all transactions in a block.
- Merkle root is required to verify all transactions present in the merkle tree instead of verifying all transactions one by one.

Merkle Tree (1979)

- A hash-based data structure that is a generalization of the hash list.
- A hash list is an extension of the concept of hashing an item.
- It is used to protect any kind of data stored, handled and transferred in and between computers.
- They make sure that the data blocks received from other peers in a peer-peer network are received undamaged and unaltered
- A cryptographic hash function such as SHA-256 is used fro hashing.
- Hashlists download only the damaged blocks and reconstruct them.

Merkle Tree



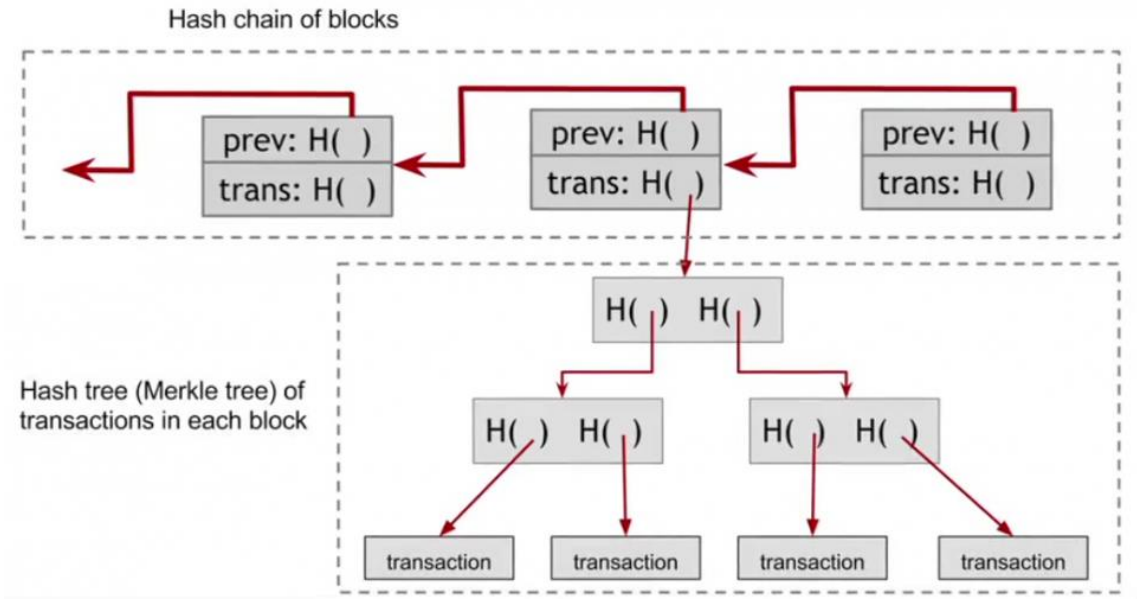
- Each leaf node is a hash of block of data.
- Each non-leaf node is a hash of its children
- They have a branching factor of 2, meaning that each node has up to 2 children.
- They are used in distributed systems for efficient data verification.
- Hashes are ways of encoding files that are much smaller than the actual file itself, widely used in peer-to-peer networks such as TOR, Bitcoin, and GIT

Merkle Tree Uses

- Who uses
  - Digital Currency
  - Global Supply Chain
  - Health Care Industry
  - Capital Markets
  - Git and Mercurial
- Why Use
  - Consistency verification
  - Data verification
  - Data Synchronization
  - Critical Proofing

Blockchain data structure

- Blockchain is the combination of two different hash-based data structures.
  - Has pointers between one block and the previous one : linkedlist
  - Inside every block the transactions are organized with a Merkle tree (hash tree).



Block Structure

- Block Header: it contains all metadata for the block.
  - It contains “nonce”
  - Reference to the block body, which is the hash of the “merkle tree root”
- Merkle Tree of Transactions: a long list of transactions organized inside this tree. Inside this tree there’s a special transaction information. It contains a value with corresponding information and reference to an output of previous transaction.

Application of Merkle Tree: Digital Currency

- Merkle trees and variations are used by
  - Bitcoin
  - Ethereum
  - Apache Cassandra
- They are used for
  - Consistency verification
  - Data verification
  - Data synchronization.

Consistency proof general algorithm

Blockchain Elements

- Address: Unique identifiers used in blockchain transaction to denote senders and recipients. An address is usually a public key or derived from a public key. Addresses are unique.
  - Bitcoin is a pseudonymous system, end users are usually not directly identifiable.
  - A good practice is for users to generate new address for each transaction in order to avoid linking transactions to the common owner.
- Transaction: A fundamental unit of blockchain. A transaction represents a transfer of value from one address to another.
- Block: A block is composed of multiple transactions and other elements such as the previous block hash, timestamp and nonce.

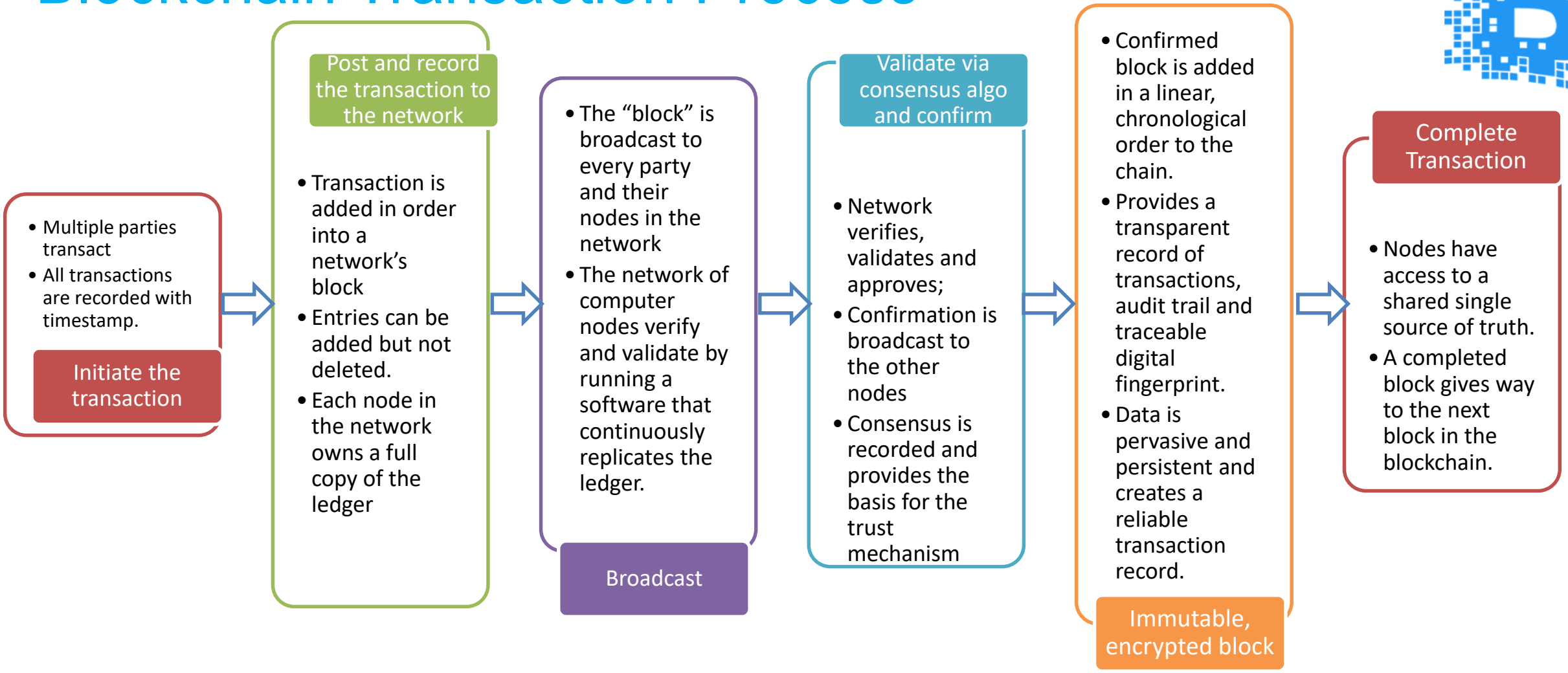
Blockchain Elements

- Peer-Peer:A network topology where in all peers can communicate with each other and send and receive messages directly.
- Virtual Machine: allows complete code to be run on blockchain as smart contracts.
  - Ethereum Virtual Machine
  - Chain Virtual Machine
- State machine: a state transition mechanism whereby a state is modified from its initial form to the next one and eventually to a final form by nodes on the blockchain network as a result of a transaction execution, validation and finalization process.

Blockchain Elements

- Smart Contract: programs that run on top of the blockchain and encapsulate the business logic to be executed when certain conditions are met.
  - Smart contracts feature is not available on all blockchain platforms, but is now becoming a very desirable feature due to flexibility and power that it provides to the blockchain applications.
- ICO: Initial Coin Offering – an unregulated means by which funds are raised for a new cryptocurrency venture.

Blockchain Transaction Process



Blockchain Benefits

- Decentralization: no need for a trusted third party or intermediary to validate transactions, instead a consensus mechanism is used to agree on the validity of transactions.
- Transparency and Trust: transparent querying of funds and disbursements.
- Immutability: Once the data is written to the blockchain, it is extremely difficult to change it back.
- High availability: high availability of peer-peer network with fault tolerance
- Highly secure: all transaction on a blockchain are cryptographically secured and thus provide network integrity.
- Cost saving: no third party expenses for clearing or approvals.

Blockchain Challenges

- Scalability
- Adaptability
- Regulatory
- Relatively immature technology
- Privacy

DEMO SESSION-1

IMPLEMENTING A BASIC BLOCKCHAIN IN JAVASCRIPT

Setting up and Installing Cycle.js

EXERCISE DEMO: 1.1

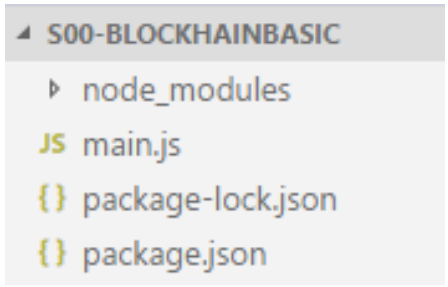
LEARNING OUTCOMES

- https://github.com/cyclejs-community/create-cycle-app

Step 1:

Create a basic blockchain application in JavaScript

- Create a folder s00-blockchainbasic
- Npm init –y (create package.json)
- Npm install --save crypto-js
- Touch main.js



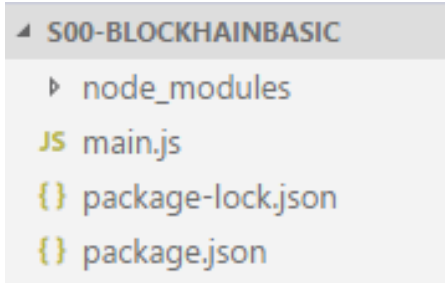

```
//importing the cryptographic library
const SHA256=require('crypto-js/sha256');

//create a block structure
class Block{

    constructor(index, timestamp,data,previousHash=''){
        this.index=index;
        this.timestamp=timestamp;
        this.data=data;
        this.previousHash=previousHash;
        this.hash='';
    }
//compute the hash -- method
    calculateHash(){
        return SHA256(this.index+this.previousHash+this.timestamp+ JSON.stringify
        (this.data)).toString();
    }
}
```

Create a blockchain class

- Create a blockchain class
  - Add createGenesisBlock()
  - Add getLatestBlock()
  - AddBlock()
  - ischainValid()




```
//blockchain class
class Blockchain{
    constructor(){
        this.chain=[this.createGenesisBlock()];
    }
    //genesis block - block zero - starting block -hardcoded
    createGenesisBlock(){
        return new Block(0,"01/01/2017","Genesis Block","0");
    }

    //fetching the latest block
    getLatestBlock(){
        return this.chain[this.chain.length-1];
    }
//adding a new block
    addBlock(newBlock){
        newBlock.previousHash=this.getLatestBlock().hash;
        newBlock.hash= newBlock.calculateHash();
        this.chain.push(newBlock);
    }
    //verifying whether a block is valid or not
    ischainValid(){
        for(let i=1;i=this.chain.length-1;i++){
            const currentBlock = this.chain[i];
            const previousBlock=this.chain[i-1];

            if(currentBlock.hash!==currentBlock.calculateHash()){
                return false;
            }
            if(currentBlock.previousHash!==previousBlock.hash){
                return false;
            }
            return true;
        }

    }
}
```

Step 3:

Create an Instance of Blockchain

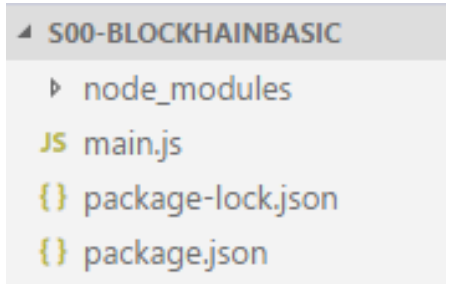
- Create an instance of the blockchain and add some transactions.
- Execute the application
  - Node main.js

```
let sycliqcoin = new Blockchain();
sycliqcoin.addBlock(new Block(1,"02/02/2017",{amount:100}));
sycliqcoin.addBlock(new Block(2,"03/03/2017",{amount:200}));


console.log(JSON.stringify(sycliqcoin, null, 4));
console.log("Is Blockchain Valid?:"+sycliqcoin.ischainValid());
```

Step 4:

Resulting blockchain output

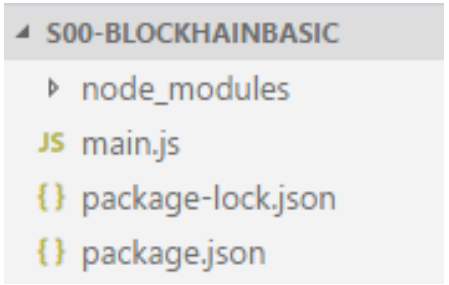



```
PS H:\awase-harddisk\CT\blockchain-js\codeplay-scenarios\s00-blockhainbasic> node main.js
{
    "chain": [
        {
            "index": 0,
            "timestamp": "01/01/2017",
            "data": "Genesis Block",
            "previousHash": "0",
            "hash": ""
        },
        {
            "index": 1,
            "timestamp": "02/02/2017",
            "data": {
                "amount": 100
            },
            "previousHash": "",
            "hash": "f55d5e1a5eeb2613962d9c18bad5343e251a8fae87754eb336e7016d8af12589"
        },
        {
            "index": 2,
            "timestamp": "03/03/2017",
            "data": {
                "amount": 200
            },
            "previousHash": "f55d5e1a5eeb2613962d9c18bad5343e251a8fae87754eb336e7016d8af12589",
            "hash": "05533aee2311eaeed8708fea05327340e4b6d738dea4047dcd5d1485bbfd744d"
        }
    ]
}
Is Blockchain Valid?:true
```

Step 5:

Altering Blockchain experiments

- Lets try to alter/modify the transaction or hash of the block.




```
console.log("------altering blockchain demo------------");
sycliqcoin.chain[1].data={amount:1266712};
//manipulating the hash
sycliqcoin.chain[1].hash=sycliqcoin.chain[1].calculateHash();
console.log("Is Blockchain Valid?:"+sycliqcoin.ischainValid());
console.log("---blocks once created cannot be tampered with/modified with------");
```

```
PS H:\awase-harddisk\CT\blockchain-js\codeplay-scenarios\s00-blockhainbasic> node main.js
{
    "chain": [
        {
            "index": 0,
            "timestamp": "01/01/2017",
            "data": "Genesis Block",
            "previousHash": "0",
            "hash": ""
        },
        {
            "index": 1,
            "timestamp": "02/02/2017",
            "data": {
                "amount": 100
            },
            "previousHash": "",
            "hash": "f55d5e1a5eeb2613962d9c18bad5343e251a8fae87754eb336e7016d8af12589"
        },
        {
            "index": 2,
            "timestamp": "03/03/2017",
            "data": {
                "amount": 200
            },
            "previousHash": "f55d5e1a5eeb2613962d9c18bad5343e251a8fae87754eb336e7016d8af12589",
            "hash": "05533aee2311eaeed8708fea05327340e4b6d738dea4047dcd5d1485bbfd744d"
        }
    ]
}
Is Blockchain Valid?:true
------altering blockchain demo------------
Is Blockchain Valid?:false
---blocks once created cannot be tampered with/modified with------
```

SYED AWASE KHIRNI

BLOCKCHAIN FEATURES

Distributed Consensus

- Distributed Consensus is the primary underpinning of a blockchain. This mechanism allows a blockchain to present a single version of the truth, which is agreed upon by all parties without the requirement of a central authority.

Transaction Verification

- Any transactions posted from the nodes on the blockchain are verified based on a predetermined set of rules. Only valid transactions are selected for inclusion in a block.

Platform for Smart Contracts

- A blockchain is a platform on which programs can run to execute business logic on behalf of the users.
  - Not all blockchains have a mechanism to execute smart contracts.
  - This is available on new blockchain platforms such as Ethereum and Multichain.
- Smart Contracts: Autonomous and Automated programs that reside on the blockchain network and encapsulate the business logic and code needed to execute a required function when certain conditions are met.

Transferring value between peers

- It enables the transfer of value between its users via tokens. Tokens can be thought of as carrier of value.

Generation of Cryptocurrency

- Optional feature, depends on the type of blockchain in use. A blockchain can create cryptocurrency as an incentive to its miners who validate the transactions and spend resource to secure the blockchain.

Smart Property

- Digital or Physical assets are linked to a blockchain in such a secure and precise manner that it cannot be claimed by anyone else.
- Owners are in full control of the asset and it cannot be double-spent or double owned.
- Blockchain can provide DRM functionality in such a way that it can be enforced fully.

5 Pillars of Information Assurance (IA)

- Availability
- Integrity
- Confidentiality
- Authentication
- Non-repudiation

Security

- Blockchain is based on proven cryptographic technology that ensures the integrity and availability of data.
- Non-repudiation and authentication are also provided by blockchain, as all actions are secured using private keys and digital signatures.
- Non-repudiation: it is the assurance that someone cannot deny the validity of something. It is a way to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.

Immutability

- Once records are added to the blockchain, they are immutable. There is the remote possibility of rolling back changes, but this is to be avoided at all costs as doing so would consume an exorbitant amount of computing resources.

Uniqueness

- This blockchain feature ensures that every transaction is unique and has not already been spent (double-spend problem). This is relevant for cryptocurrencies, where detection and avoidance of double spending are a vital requirement.

Types of Blockchain

- Distributed Ledgers
- Distributed Ledger Technology (DLT)
- Blockchain
- Ledgers

Distributed Ledgers

- A distributed ledger is a broad term describing shared databases, all blockchain technically fall under the umbrella of shared databases or distributed ledgers.
- All distributed ledgers are not necessarily a block chain
- A distributed ledger is distributed among its participants and spread across multiple sites or organizations.
- This type of ledger can be either private or public.
- **The records are stored contiguously.**
- A distributed ledger does not necessarily consist of blocks of transactions to keep the ledger growing.
- A blockchain is special type of shared databased that is comprised of blocks of transactions.
- R3 Corda is a distributed ledger which is developed to record and manage agreements and is especially focused on financial service industry.

Distributed Ledger Technology (DLT)

- DLT and blockchain are used interchangeably.
- DLT are permissioned blockchains that are shared and used between known participants.
- DLTs usually serve as a shared database, with all participants known and verified.
- They do not have cryptocurrency or do not require mining to secure the ledger

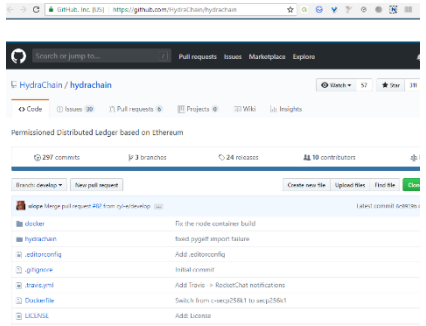
Public Blockchains

- They are not owned by anyone, but open to public and anyone can participate as a node in the decision-making process.
- Users may or may not be rewarded for their participation.
- All users of these permissionless or unpermissioned ledgers maintain a copy of the ledger on their local nodes and used distributed consensus mechanism to decide the eventual state of the ledger.
- Bitcoin and Ethereum are both considered public blockchains.
- Anyone can read/write without explicit authorization, but involves complex rules for security.
- It involves a complex consensus algorithm
- Computationally expensive to mine and add a block.
- Computational power is distributed globally.

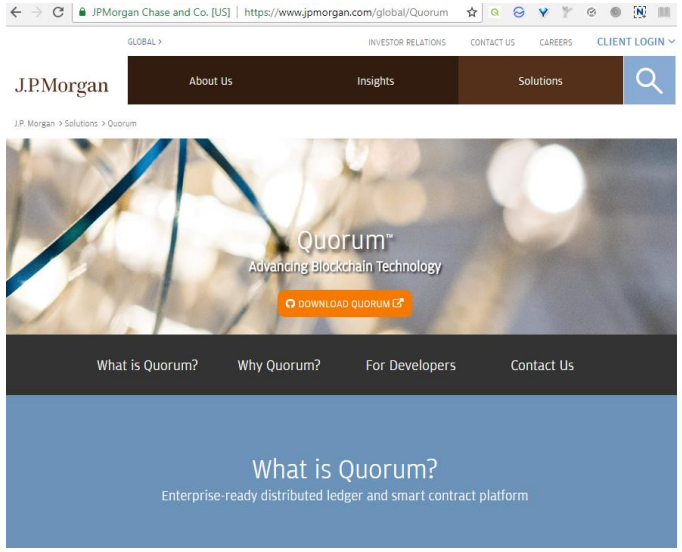
Private Blockchains

- They are just private, they are only open to a consortium or group of individuals or organization who have decide to share the ledger among themselves.
- Hydrachain and Quorum are two private blockchains.

https://github.com/HydraChain/hydrachain



https://www.jpmorgan.com/global/Quorum



Semiprivate blockchains

- Hybrid model where part of the blockchain is private and part of the it is public.

Sidechains/Pegged Sidechains

- It is a concept where by coins can be moved from one blockchain to another and moved back again.
- Typical uses include the creation of new altcoins (alternative cryptocurrencies) where by coins are burnt as a proof of an adequate stake.
- Burning the coins means that the coins are sent to an address which is unspendable and this process makes the burnt coins irrecoverable.
- Usually used to bootstrap a new currency or introduce scarcity which results in increased value of the coin.

Sidechains/Pegged Sidechains.

- Proof of Burn (PoB) is an alternative method for distributed consensus to PoW and Proof of Stake. It is also called one-way pegged side chain.
- Two-way pegged sidechain allows the movement of coins from the main chain to the sidechain and back to the main chain when required.
- Rootstock is an example of a sidechain, which enables smart contract development for Bitcoin using this paradigm.

Permissioned Ledger

- A blockchain where participants of the network are already known and trusted.
- No necessity for distributed consensus mechanism, instead an agreement protocol is used to maintain a shared version of the truth about the state of the records on the blockchain.
- All verifiers are preselected by a central authority and no necessity for mining mechanism.
- It's access control is regulated.
- Bitcoin can be a permissioned ledger if an access control layer is introduced on the top of it that verifies the identity of a user and then allows access to the blockchain.

Shared ledger

- A generic term that is used to describe any application or database that is shared by the public of a consortium.
- All blockchains fall into the category of a shared ledger.

Tokenized blockchains

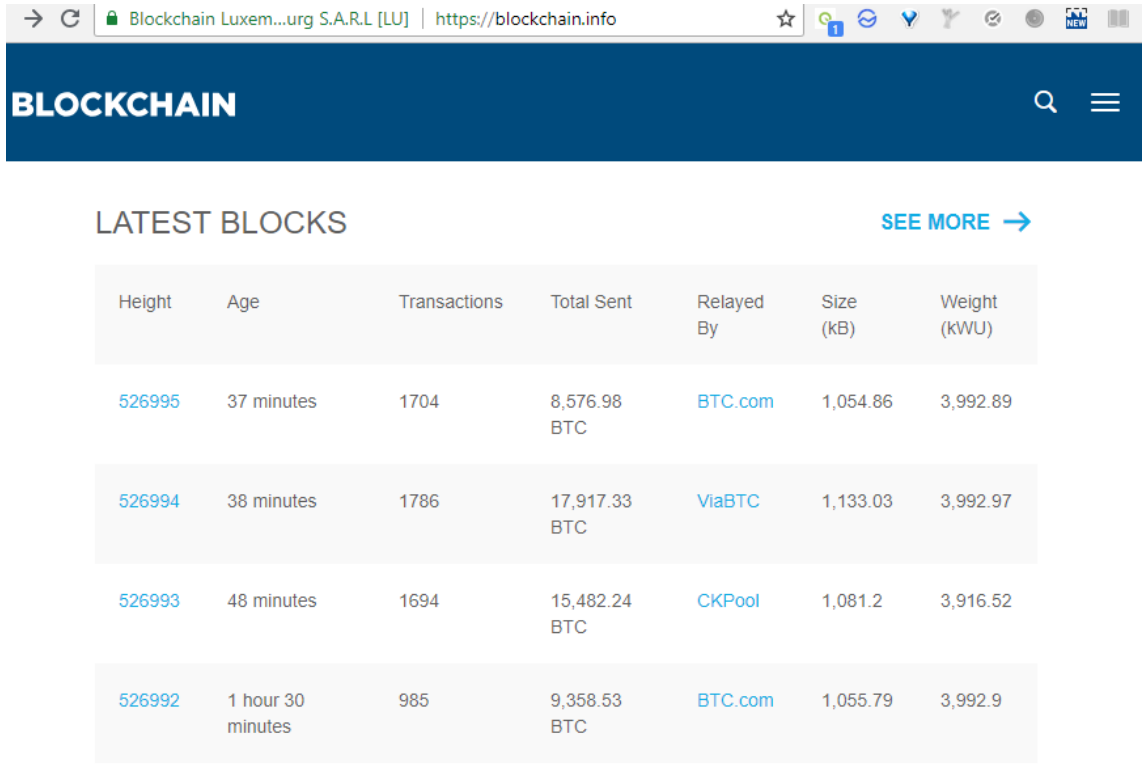
- They are standard blockchains that generate cryptocurrency as a result of a consensus process via mining or initial distribution.
- Bitcoin and Ethereum are prime examples of this type of blockchain.

Tokenless blockchains

- They are designed in such a way that they do not have the basic unit for the transfer of value.
- Used in situations where there is no need to transfer value between nodes and only the sharing of the data among various trusted parties is required.
- Similar to private blockchains, the only difference being that use of tokens is not required.
- Similar to a shared distributed ledger used for storing data.
- It provides
  - Immutability
  - Security
  - Consensus based updates.

Blockchain Info

https://blockchain.info/



Height	Age	Transactions	Total Sent	Relayed By	Size (kB)	Weight (kWU)
526995	37 minutes	1704	8,576.98 BTC	BTC.com	1,054.86	3,992.89
526994	38 minutes	1786	17,917.33 BTC	ViaBTC	1,133.03	3,992.97
526993	48 minutes	1694	15,482.24 BTC	CKPool	1,081.2	3,916.52
526992	1 hour 30 minutes	985	9,358.53 BTC	BTC.com	1,055.79	3,992.9

Block #416731

Summary	
Number Of Transactions	3077
Output Total	29,086.43197851 BTC
Estimated Transaction Volume	6,294.44741544 BTC
Transaction Fees	0.97211332 BTC
Height	416731 (Main Chain)
Timestamp	2016-06-17 16:41:18
Received Time	2016-06-17 16:41:18
Relayed By	BTCC Pool
Difficulty	196,061,423,939.65
Bits	403020704
Size	997.223 kB
Weight	3988.64 kWU
Version	0x20000001
Nonce	4279594355
Block Reward	25 BTC

Hashes	
Hash	00000000000000000043d73959c1fd990d2a783b38c79e801ce9daab567446671
Previous Block	00000000000000000055d59425a4f802b41bc1cbf3437329525ab5d5e6b4687a6
Next Block(s)	00000000000000000001571bcd52f918ce67af43d406f0fef83b965aee59b23137
Merkle Root	613782e97b57a9805610bfab1f2fb6391edc8c55994f0a3de836b3d0950c6192

SCALABILITY

GOVERNANCE

PERFORMANCE

DATA CAPACITY

FAILURE REDUNDANCY

CONSISTENCY

SECURITY

SYED AWASE KHIRNI

CONSENSUS PROTOCOLS OR PLATFORMS

Consensus

- It is the backbone of a blockchain and as a result provides decentralization of control through an optional process known as mining.
- Choice of consensus algorithm is also governed by the type of blockchain in use.

- A process of agreement between distrusting nodes on the final state of data. To achieve consensus, different algorithms are used.
  - Drawing consensus among multiple nodes to agree on a single value.
  - It is the process of attaining agreement on common state or value among multiple nodes despite the failure of some nodes is known as distributed consensus.

Consensus mechanism

- A set of steps that are taken by most or all nodes in a blockchain to agree on a proposed state or value.
- It is a means of agreeing to a single version of the truth by all peers on the blockchain network.

- Various requirements that must be met to provide desired result in a consensus mechanism
  - Agreement – all honest nodes decide on the same value
  - Termination- all honest nodes terminate execution of the consensus process and eventually reach a decision.
  - Validity- the value agreed upon by all honest nodes must be the same as the initial value proposed by atleast one honest node.
  - Fault tolerant: consensus algorithm should be able to run in the presence of faulty or malicious nodes (Byzantine nodes)
  - Integrity: a requirement that no node can make the decision more than once in a single consensus cycle.

BASIC PARAMETERS THAT DEFINE CONSENSUS

1. Decentralized governance: a single central authority cannot provide transaction finality
2. Quorum structure: Nodes exchange messages in predefined ways, which may include stages or tiers.
3. Authentication: this process provides means to verify the participants identities.
4. Integrity: it enforces the validation of the transaction integrity.
5. Non-repudiation: this provides means to verify that the supposed sender really sent the message.
6. Privacy: it helps ensure that only the intended recipient can read the message
7. Fault Tolerance: the network operates efficiently and quickly, even if some nodes or servers fail or are slow.
8. Performance: it considers throughput, liveness, scalability and latency.

Types of Consensus Mechanisms

- All consensus mechanisms are developed to deal with faults in a distributed system and to allow distributed systems to reach a final state of agreement.
- Two general categories
  - Traditional byzantine fault tolerance (BFT) based :
  - Leader election-based consensus mechanism

Traditional Byzantine Fault Tolerance (BFT)

- They are not compute intensive operations, they rely on simple scheme of nodes that are publisher-signed messages. Eventually, when a certain number of messages are received, then an agreement is reached.
- It is more traditional approach based on rounds of votes. This class of consensus is also known as consortium or permissioned type of consensus mechanism.
- Performs well when there are limited number of nodes, but they do not scale well.

Leader election-based consensus

- It requires nodes to compete in a leader-election lottery and the node that wins proposes a final value.
- Proof-based, leader-election based or the Nakamoto consensus whereby a leader is elected at random using an algorithm and proposes a final value.
- Fully decentralized or permissionless type of consensus mechanism.
- They scale very well but perform very slowly and are based on proof of work.

Other Proposals for consensus

- PBFT
- Hybrid BFT
- Block DAG
- Tezos
- Stellar
- GHOST

Consensus based algorithms

- Proof of Work (PoW)
- Proof of Stake (PoS)
- Delegated Proof of Stake (DPoS)
- Proof of Elapsed Time (PoET)
- Proof of Deposit (PoD)
- Proof of Importance (PoI)
- Federated Consensus or Federated Byzantine Consensus.
- Reputated-based mechanisms
- Proof of Activity (PoA)
- Proof of Capacity (PoC)
- Proof of Storage (PoS)

Consensus mechanisms

- Stellar (Ripple Fork)
- Ripple (evolving into inter-ledger protocol)

Federated Consensus

- Open Chain
- Proof of Elapsed Time by Intel (Sawtooth Lake Project)

Proprietary Distributed Ledger

- PBFT
- Derived PBFT
- RBFT –Redundant Byzantine Fault Tolerance e.g. Everynym
- SBFT – Simplified Byzantine Fault Tolerance e.g. chain

Practical Byzantine Fault Tolerance and Derivatives

- Distributed concurrence
- Corda (R3CEV)

N2N

- Bitcoin
- Colored Coins
- Proprietary Metacoins
- DAG (Directed Acyclic Graphs)
- Factom
- Coinprism

Proof of Work

- Casper
- Ethereum (moving to POS)

Proof of stake

- Graphene
- Steem
- Bit hares

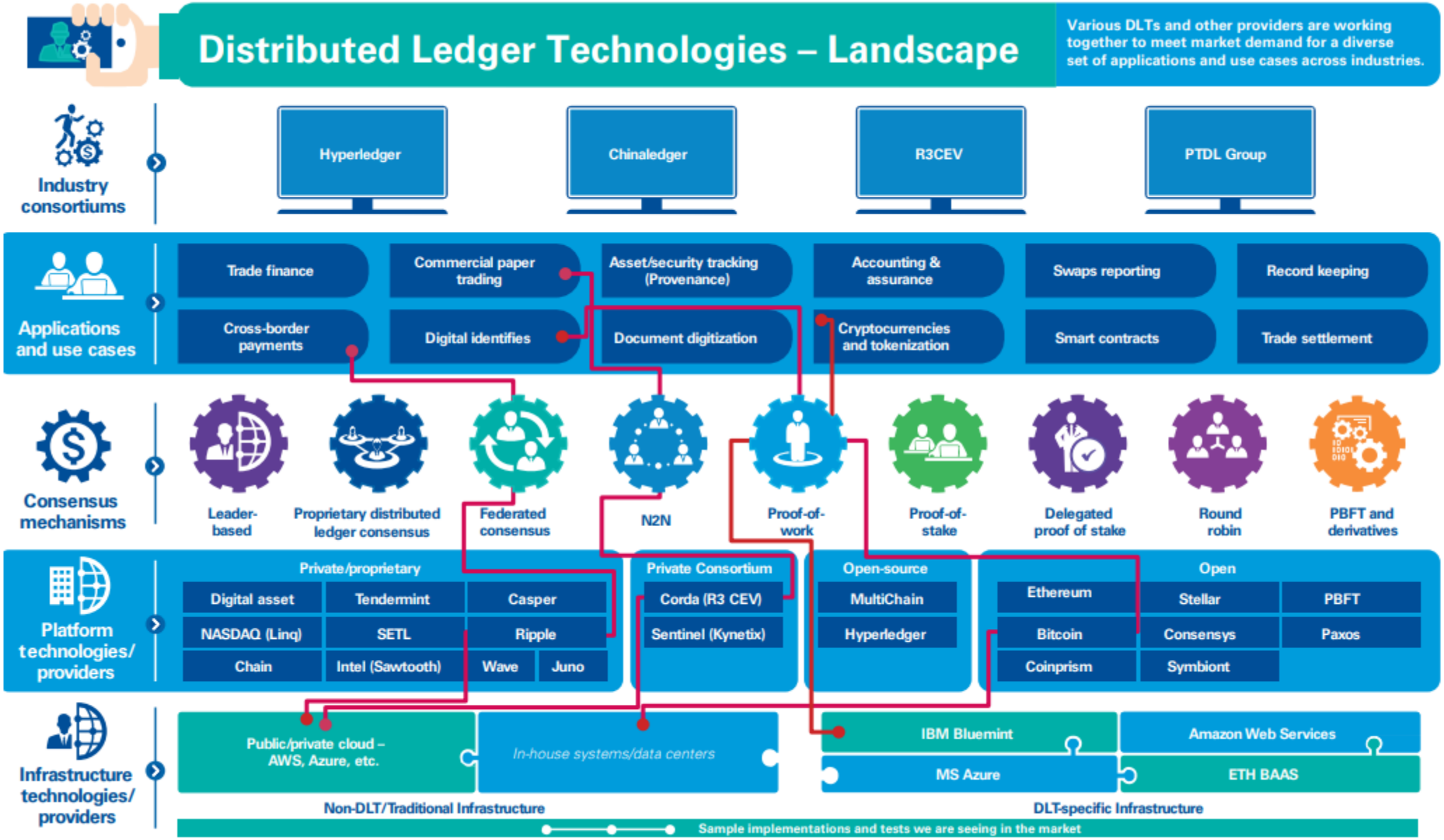
Delegated Proof of Stake

- BigChainDB
- RAFT
- PAXOS
- JUNO
- Tangaroa
- Mencius

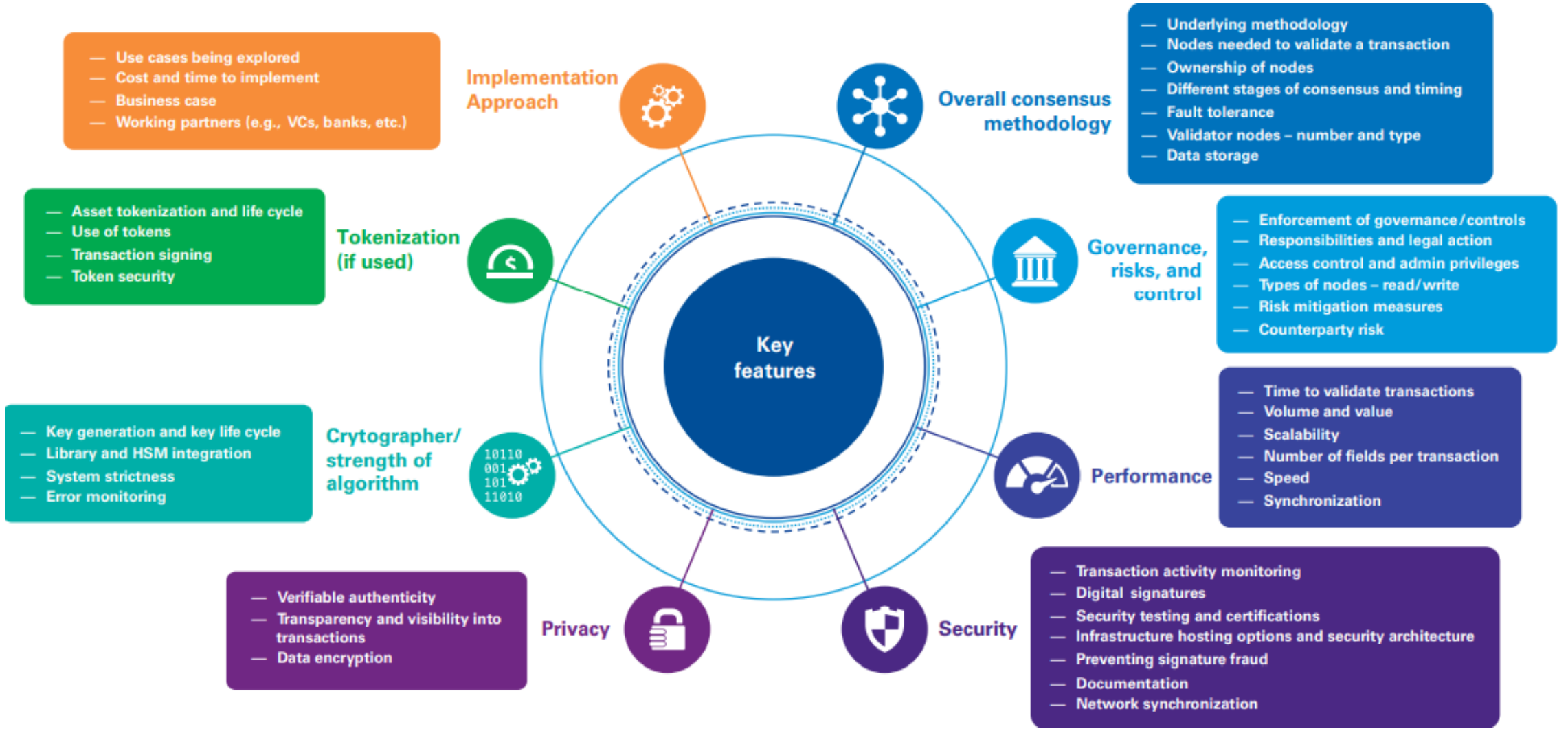
Leader-based consensus (including PAXOS/RAFT-based derivatives)

- Multichain
- Tendermint

Round Robin



Source: Illustrative distributed ledger technologies fig4. Pg 10, https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmg-blockchain-consensus-mechanism.pdf

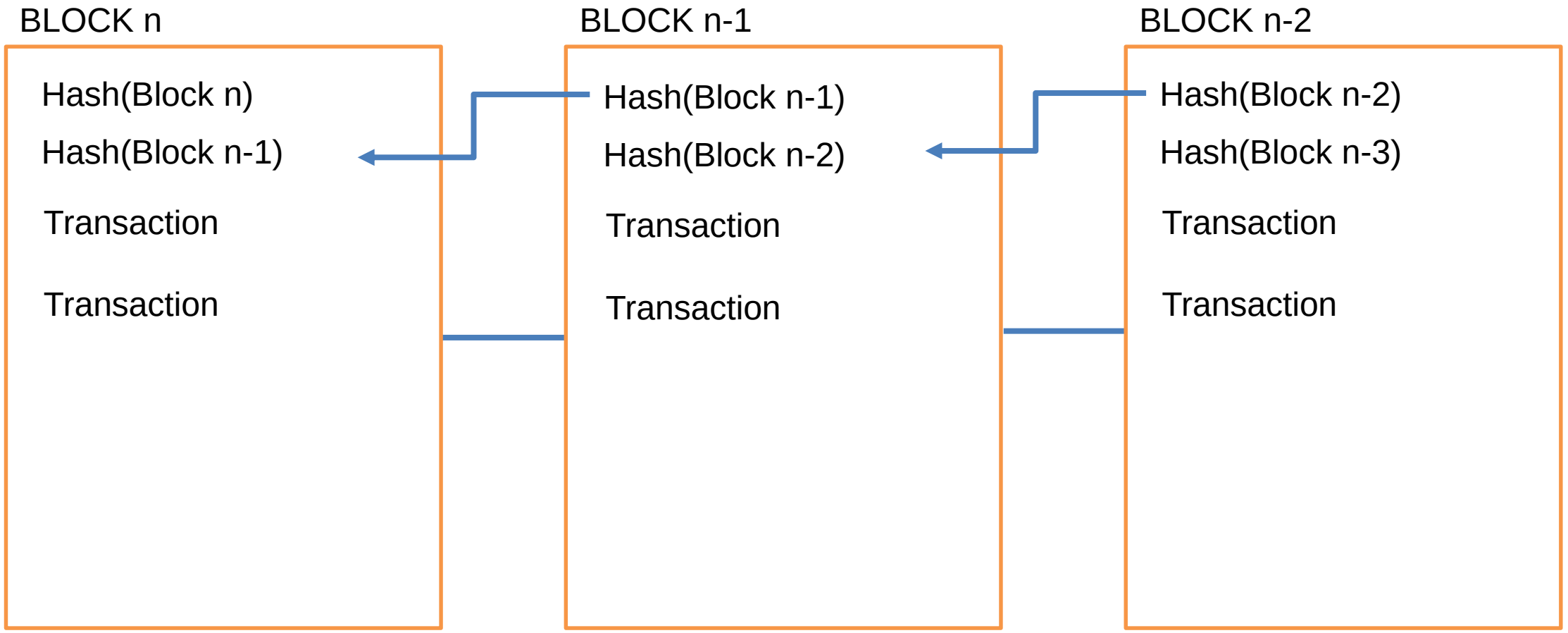


Source: Distributed consensus evaluation framework fig 6, pg.12 https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmg-blockchain-consensus-mechanism.pdf

Proof of Work (PoW)

- It relies on proof that adequate computational resources have been spent before proposing a value for acceptance by the network.
- It is the original consensus algorithm in a blockchain network.
- Used to confirm transactions and produce new blocks to the chain.
- Miners compete against each other to complete transactions on the network and get rewarded.

- In a network users send each other digital tokens.
- A decentralized ledger gathers all the transactions into blocks.
- Implemented in Bitcoin, Litecoin.
- Ehtereum also uses PoW.



PoW Benefits

- Defense from DoS attacks- PoW imposes some limits on actions in the network. They need a lot of effort to be executed. Efficient attack requires a lot of computation power and a lot of time to do the calculations. Costs of attack are high.
- Mining Possibilities: What matters is to have large computational power to solve the puzzles and form new blocks. Thus, the holders of huge amounts of money are not in charge of making decisions for the entire network.

Hash function

- A hash function is any function that can be used to map data of arbitrary size to the data of fixed size.

Proof of Stake (PoS)

- It works on the idea that a node or a user has an adequate stake in the system, the user has invested enough in the system so that any malicious attempt by the user would outweight the benefits of performing such an attack on the network.
- First introduced in PeerCoin and used in Ethereum blockchain version called serenity.
- Coinage: derived from amount of time and number of coins that have not been spent.
- The chances of proposing and signing the next block increases with the coin age.

Delegated Proof of Stake (DPoS)

- An improvisation on the standard PoS, whereby each node that has a stake in the system can delegate the validation of a transaction to other nodes by voting.
- It is used in the BitShares Blockchain.

Proof of Elapsed Time (PoET)

- Introduced by intel in 2016.
- PoET uses a Trusted Execution Environment TEE to provide randomness and safety in the leader election process via a guaranteed wait time.
- It required the intel software guard extensions (SGX) processor to provide the security guarantee.

Proof of Deposit (PoD)

- The nodes that wish to participate in the network have to make a security deposit before they can mine and propose blocks.
- This is used in the tendermint blockchain.

Proof of Importance (PoI)

- The level of trust and importance is computed based on the percentage share of stake a user has in the system and the frequency of usage and movement of tokens by the user inorder to establish a level of trust and importance.
- It is widely used in NEM coin blockchain (https://nem.io)

Federated consensus or federated byzantine consensus

- Widely used in the stellar consensus protocol.
- Nodes in this protocol retain a group of publicly trust peers and propogate only those transactions that have been validated by the majority of trusted nodes.

Reputation based mechanisms

- A leader is elected by the reputation it has built over time on the network. It is based on the votes of other members.

Proof of Activity (PoA)

- It is combination of PoS and PoW which ensures that a stakeholder is selected in a pseudorandom but uniform fashion.
- They are both combine together to achieve consensus and good level of security.

Proof of Capacity (PoC)

- It uses hard disk space as a resource to mine the blocks. This is different from PoW, where CPU resources are used.
- Hard disk space is utilized for mining and as such is also known as hard drive mining
- Introduced by Burstcoin cryptocurrency.

Proof of Storage(PoS)

- It allows for the outsourcing of storage capacity.
- This scheme is based on the concept that a particular piece of data is probably stored by a node which serves as a means to participate in the consensus mechanism.
- Several variation of this scheme are proposed
  - Proof of replication
  - Proof of Data Possession
  - Proof of space
  - Proof of Space-time

CAP THEOREM/ BREWER's THEOREM

- The theory states that any distributed system cannot have consistency, availability and partition tolerance simultaneously
  - Consistency is a property which ensures that all nodes in a distributed system have a single, current and identical copy of the data.
- Availability means that the nodes in the system are up, accessible for use and are accepting incoming requests and responding with data without any failures as and when required.
- Partition tolerance ensures that if a group of nodes is unable to communicate with other nodes due to network failures, the distributed system continues to operate correctly

Fault Types

- Fail-stop fault: it occurs when a node merely has crashed. Fail-stop faults are easier ones to deal with of the two fault types.
- Byzantine faults: it is one where the faulty node exhibits malicious or inconsistent behavior arbitrarily. This type is difficult to handle, since it can create confusion due to misleading information.

Decentralization using blockchain

- Blockchain by design, are decentralized, where any node can compete to become the decision-making authority, by a consensus mechanism.
- Control is distributed among many nodes, enabling users to agree on something via a consensus algorithm without the need for a central, trusted third party, intermediary or service provider.

- Two methods of decentralization
  - Disintermediation
  - Competition (contest-driven decentralization)
- Decentralization
  - What is being decentralized?
  - What level of decentralization is required?
  - What blockchain is used?
  - What security mechanism is used?

Disintermediation

- It involves direct communication between nodes without using any third party intermediaries such as banks or some authority to validate or authenticate.
- Widely used in bitcoin blockchain.

Contest-driven decentralization

- Different service providers compete with each other in order to be selected fro the provision of services by the system.
- In blockchain technology, a system can be envisioned in which smart contracts can choose an external data provider from a large number of providers based on their reputation, previous score, reviews and quality of service.

Blockchain vs Traditional database

Blockchain

- Anonymous and secure transactions.
- Consensus based mechanisms for transferring value based assets.
- Bitcoin and Ethereum are best examples of blockchain.
- Ethereum has become more prominent choice because of the flexibility it allows for programming any business logic into the blockchain by using smart contracts.

Traditional Database

- High data throughput required.
- Centrally controlled updates.
- Trusted nodes and secured through permission based mechanisms

Data Storage in BlockChains

- Data can be stored directly in a blockchain and with this fact it achieves decentralization, but it cannot store large amounts of data by design.
- Distributed Hash Tables (DHTs) are widely used initially in peer-to-peer file sharing software such as BitTorrent.
- The only challenge is there is no incentive for users to keep the files indefinitely.

- There are two primary requirements
  - High availability
  - Link stability
- Two new concepts were proposed
  - InterPlanetary File System (IPFS) by Juan Benet with a vision to provide decentralized WWW by replacing the HTTP Protocol
  - Merkle Directed Acyclic Graph (DAG) to provide storage and searching functionality.

FileCoin Protocol

- An incentive mechanism for storing data, which pays incentives to nodes that store data using the Bitswap mechanism.
- Bitswap mechanism lets nodes keep a simple ledger of bytes sent or bytes received in one-to-one relationship.
- A git based control mechanism is used in IPFS to provide structure and control over the versioning of the data.
- Other Alternatives for data storage
  - Ethereum Swarm
  - Storj
  - MaidSage.

Ethereum Data Storage:Ethereum Swarm

- It has its own decentralized and distributed ecosystem that uses Swarm for storage and the Whisper protocol for communication.

MaidSafe

- It aims to provide a decentralized world wide web
- It provides a Secure Access for Everyone (SAFE) network that is made up of unused computing resources, such as storage, processing power and the data connections of its users.
- It uses Safecoin as a token to incentivize its contributors.
- The files on the network are divided into small chunks of data, which are encrypted and distributed randomly throughout the network.
- This data can only be retrieved by its respective owner.
- Duplicate files are automatically rejected on the network, which helps reduce the need for additional computing resources needed to manage the load.

Communication Layer in Blockchain

- The communication layer in blockchain is considered to be decentralized.
- An example of peer-to-peer fashion communication with each other without an internet connection is “firechat”.

Smart Contracts

- A smart contract is a decentralized program.
- Smart contracts do not necessarily need a blockchain to run, however, due to the security benefits that blockchain technology provides, blockchain has become a standard decentralized execution platform for smart contracts.
- A smart contract usually contains some business logic and a limited amount of data.
- The business logic is executed if specific criteria are met.
- Actors or participants in the blockchain use these smart contract or they run autonomously on behalf of the network participants.

Decentralized organizations.

- Dos are software programs that run on a blockchain and are based on the idea of actual organizations with people and protocols.
- Once a DO is added to the blockchain in the form of a smart contract or a set of smart contract, it becomes decentralized and parties interact with each other on the code defined within the DO software

Decentralized Autonomous Organizations

- They run a top on a blockchain and embedded within it are governance and business logic rules.
- DAO and DO are fundamentally the same thing.
- DAOs are autonomous, which means that they are fully automated and contain artificially-intelligent logic.
- First introduced by Ethereum blockchain.
- In DAO, the code is considered the governing entity rather than people or paper contracts.
- A human curator maintains this code and acts a proposal evaluator for the community. DAOs are capable of hiring external contractors if enough input is received from the token holders

Decentralized Autonomous Corporations

- They are similar to DAO's in concept, though considered to be a smaller subset of them.
- DAO's are considered to be non-profit, which DAC's are considered to earn profit via shares offered to the participants and to whom they can pay dividends.
- DACs can run a business automatically without human intervention based on the logic programmed into them.

Decentralized Autonomous Societies.

- DAS is a concept where an entire society can function on a blockchain with the help of multiple, complex smart contracts and a combination of DAOs

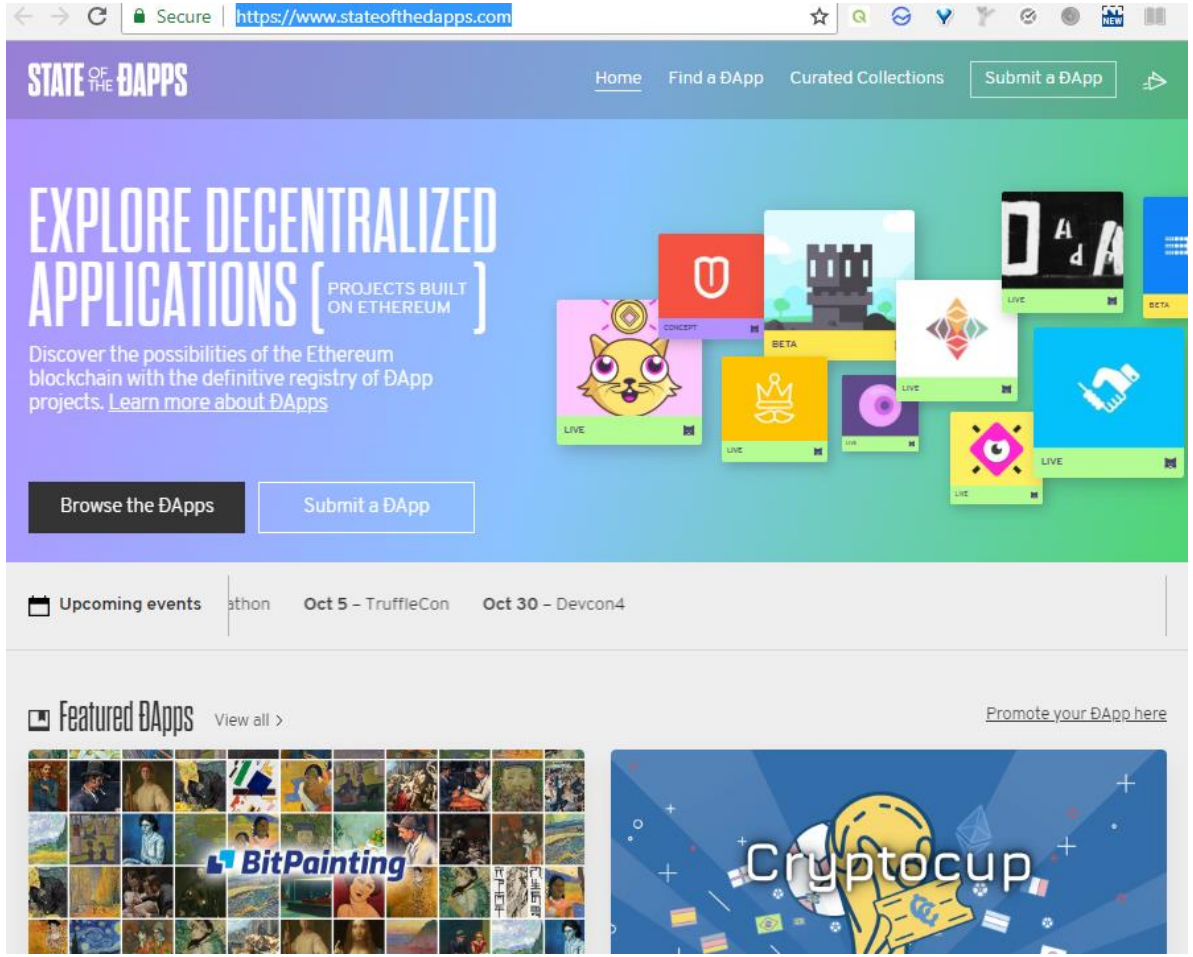
Decentralized Applications Requirements

- Criteria for an application to be considered decentralized:
  - Should be fully open sourced and autonomous and no single entity should in control of a majority of its tokens. All changes to the application must be consensus-driven based on the feedback given by the community.
  - The tokens must be generated by the DApp according to the standard cryptographic algorithm,
  - Data and records of operations of the applications must be cryptographically secured and stored on a public, decentralized blockchain to avoid any central points of failure.
  - A cryptographic token must be used by the application to provide access and rewards to those who contribute value to the applications.

Decentralized Applications Examples

- KYC-Chain provides the facility to manage Know Your Customer (KYC) data securely and conveniently based on smart contracts.
- OpenBazaar: a decentralized peer-to-peer network that enables commercial activities directly between sellers and buyers instead of relying on central party.
- Lazooz: a decentralized equalivalent of Uber, allows peer-to-peer ride sharing and users to be incentivized by proof of movement, and they can earn zooz coins.

https://www.stateofthedapps.com/



Decentralization Platforms

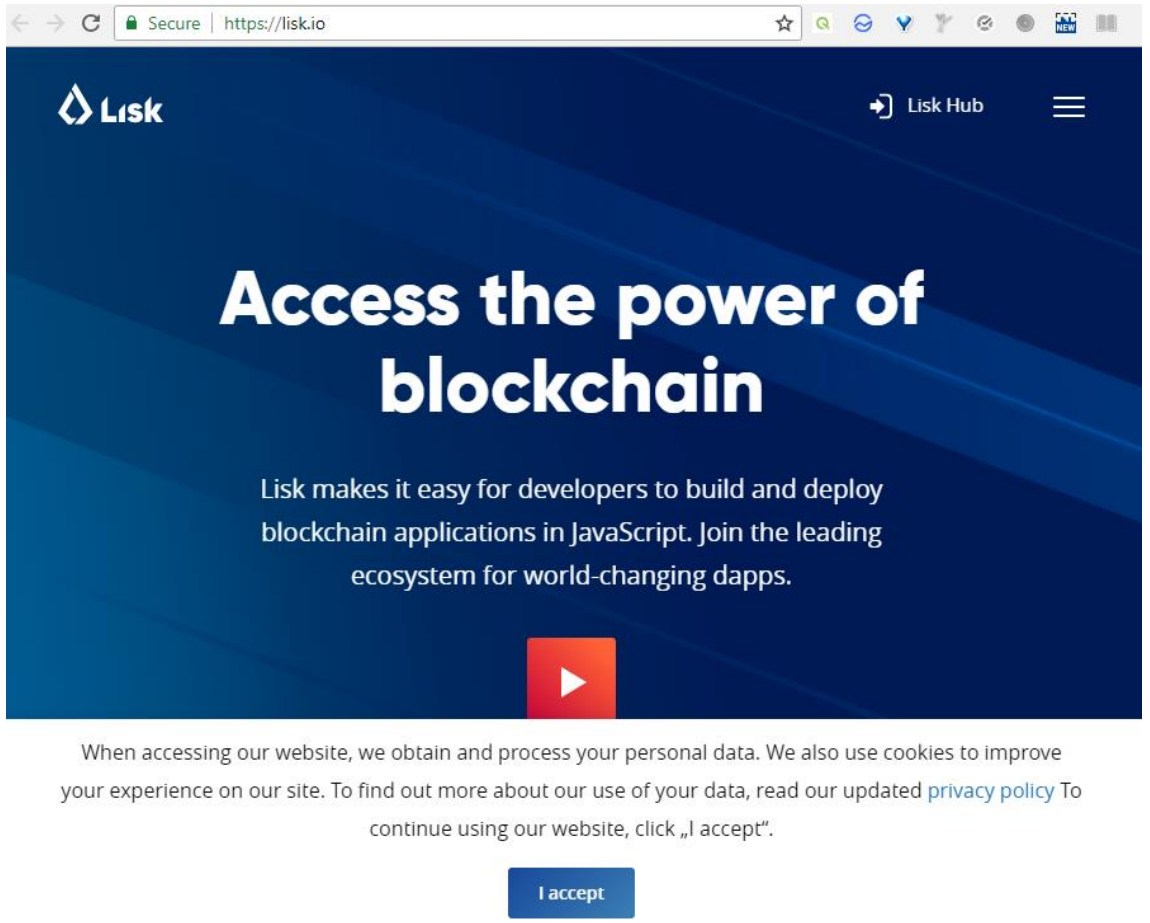
- Bitcoin
- Ethereum
- Hyperledger Fabric
- Quorum

- Ethereum: first blockchain to introduce turing-complete language and the concept of a virtual machine. Turing complete language called Solidity, endless possibilities have opened for the development of decentralized applications. (Vitalik Buterin). Currency tokens on Ethereum are called Ethers.

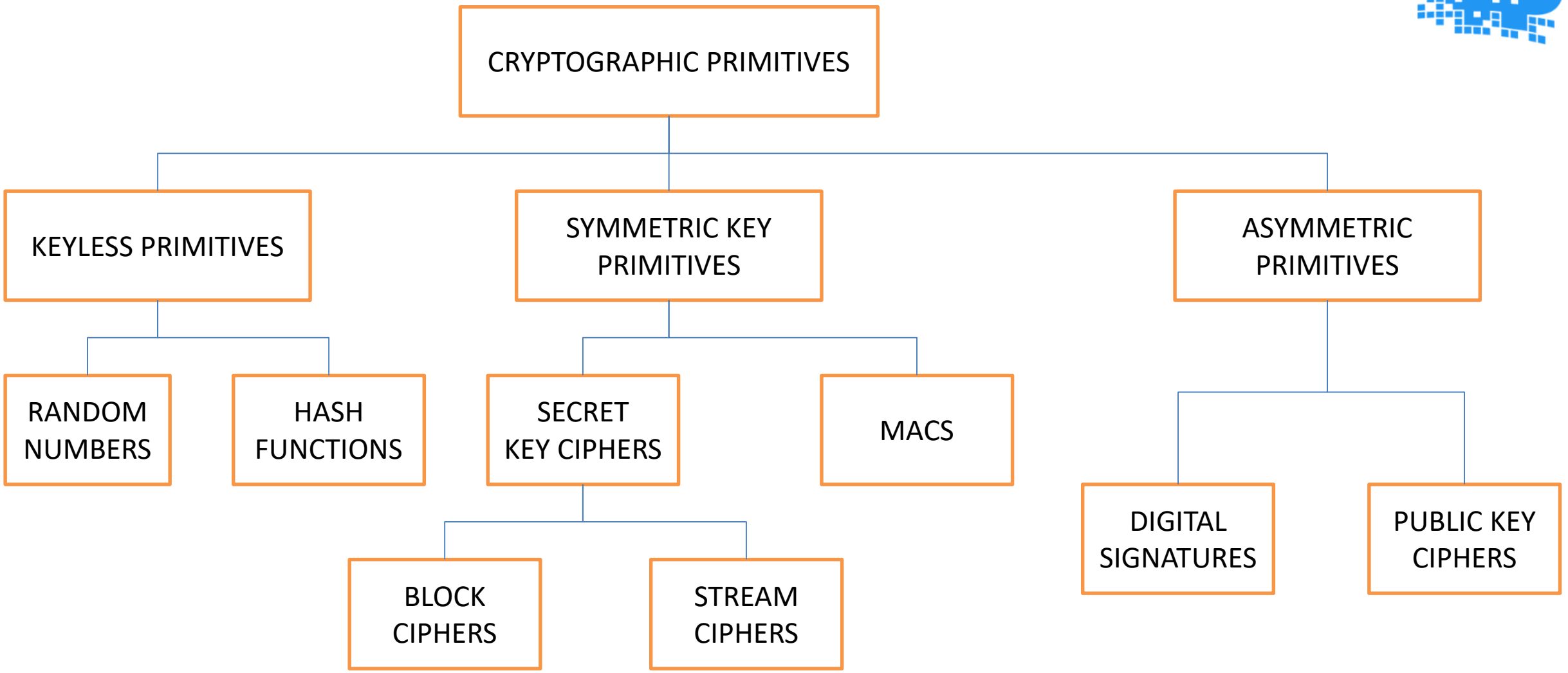
Lisk

- Lisk is blockchain application development and cryptocurrency platform. It allows developers to use JavaScript to build decentralized applications and host them in their respective sidechains.
- It uses Delegated Proof of Stake (DPOS) mechanism for consensus where by 101 nodes can be elected to secure the network and propose blocks
- It uses the node.js and JavaScript backend, where the frontend allows the use of standard technologies such as CSS3, HTML5 and JavaScript
- Lisk uses LSK coin as currency on the blockchain. Another derivative of Lis is Rise, which is a Lisk based decentralized application and digital currency platform.

https://lisk.io/



Cryptographic Primitives



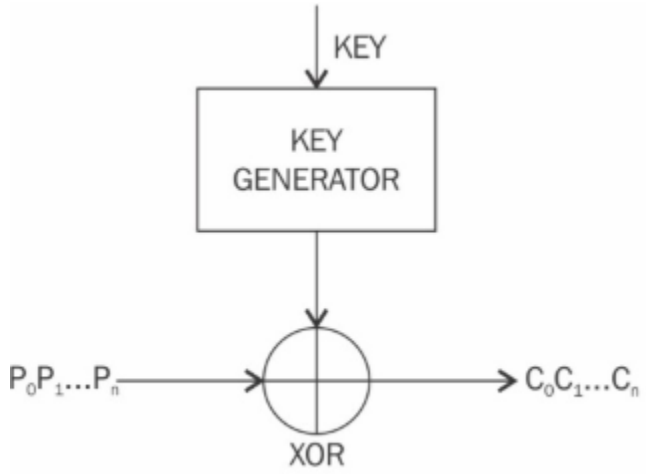
SYMMETRIC CRYPTOGRAPHY

- A type of cryptography where the key that is used to encrypt the data is the same one that is used for decrypting the data.
- It is also known as shared key cryptography.
- The key must be established or agreed upon before the data exchange occurs between the communicating parties, also called as secret key cryptography

- Two types of symmetric ciphers:
  - Stream ciphers
    - RC4
    - A5
  - Block ciphers
    - Data Encryption Standard (DES)
    - Advanced Encryption Standard (AES)

STREAM CIPHERS

- An encryption algorithm that is applied on a bit-by-bit basis (one bit at a time) to plaintext using a keystream.
  - There are two types
    - Synchronous stream ciphers are those where the keystream is dependent only on the key.
    - Asynchronous stream ciphers have a keystream that is also dependent on the encrypted data.

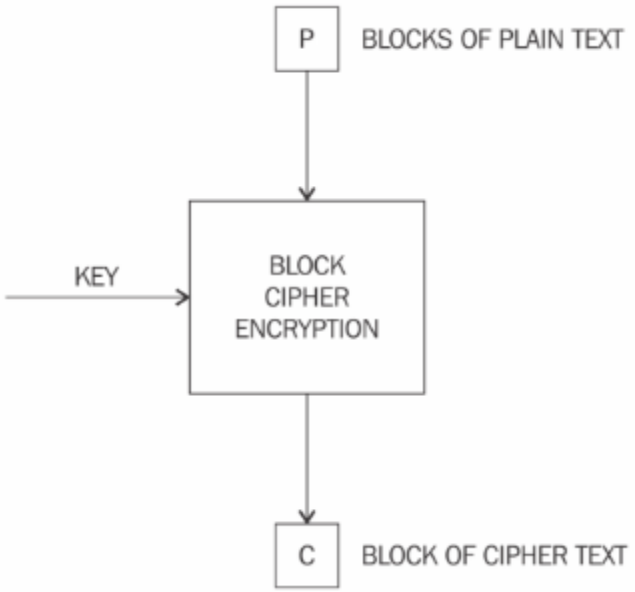


Block ciphers

- Encryption algorithms that break up the text to be encrypted (plaintext) into blocks of a fixed length and apply the encryption block-by-block.
- Block ciphers are generally built using a design strategy known as a Feistel cipher.
  - AES(Rjindael)- Substitution-Permutation Network (SPN)
- Fieitel ciphers are based on Feistel networks, which is a structure developed by Horst Feistel.
- This structure is based on the idea of combining multiple rounds of repeated operations to achieve desirable cryptographic properties known as confusion and diffusion.

Block Ciphers

- Feistel networks operate by dividing data into two blocks (left and right) and processing these blocks via keyed round functions in iterations to provide sufficient pseudorandom permutation.
- Various modes of operation for block ciphers are **Electronic Code Book** (**ECB**), **Cipher Block Chaining** (**CBC**), **Output Feedback** (**OFB**) mode, and **Counter** (**CTR**) mode.
- These modes are used to specify the way in which an encryption function is applied to the plaintext. Some of these modes of block cipher encryption are introduced here.

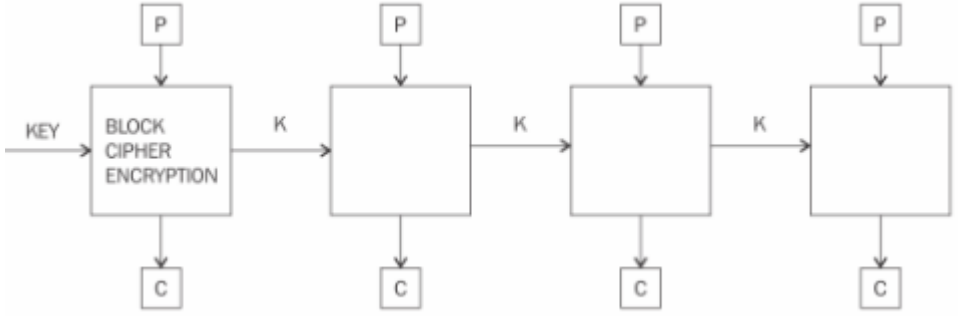


Block encryption mode

- The plaintext is divided into blocks of fixed length depending on the type of cipher used.
- Then the encryption function is applied to each block.

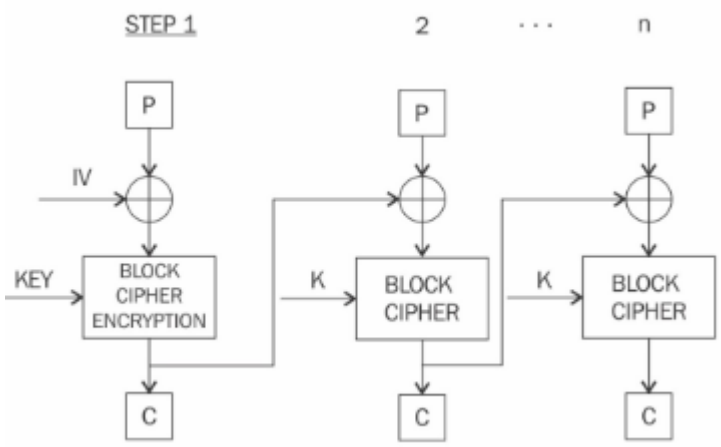
Electronic code book

- A mode of operation in which the encrypted data is produced as a result of applying the encryption algorithm one-by-one to each block of plain text.



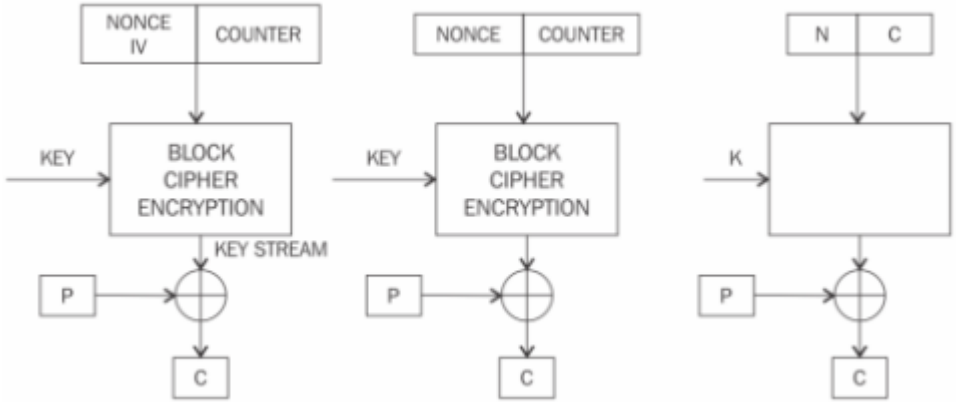
CIPHER BLOCK CHAINING

- Each block of plaintext is XOR'd with the previously-encrypted block. It uses initialization vector to encrypt the first block.



COUNTER MODE

- Effectively uses a block cipher as a stream cipher.
- A unique nonce is supplied that is concatenated with the counter value to produce keystream.



KEYSTREAM GENERATION MODE

- The encryption function generates a keystream that is then XOR'd with the plaintext stream to achieve encryption.

MESSAGE AUTHENTICATION MODE

- MAC results from an encryption function. The MAC is a cryptographic checksum that provides an integrity service.

CRYPTOGRAPHIC HASH MODE

- Hash functions are primarily used to compress a message to a fixed-length digest. In cryptographic hash mode, block ciphers are used as a compression function to produce a hash of plaintext.

DATA ENCRYPTION STANDARD

- Data Encryption Standard DES was introduced by the U.S.National Institute of Standards and Technology NIST as a standard algorithm for encryption (1980).
- Due to security vulnerability there was an upgraded version 3DES which uses 168-bit key instead of 56-bit key encryptions.

ADVANCED ENCRYPTION STANDARD

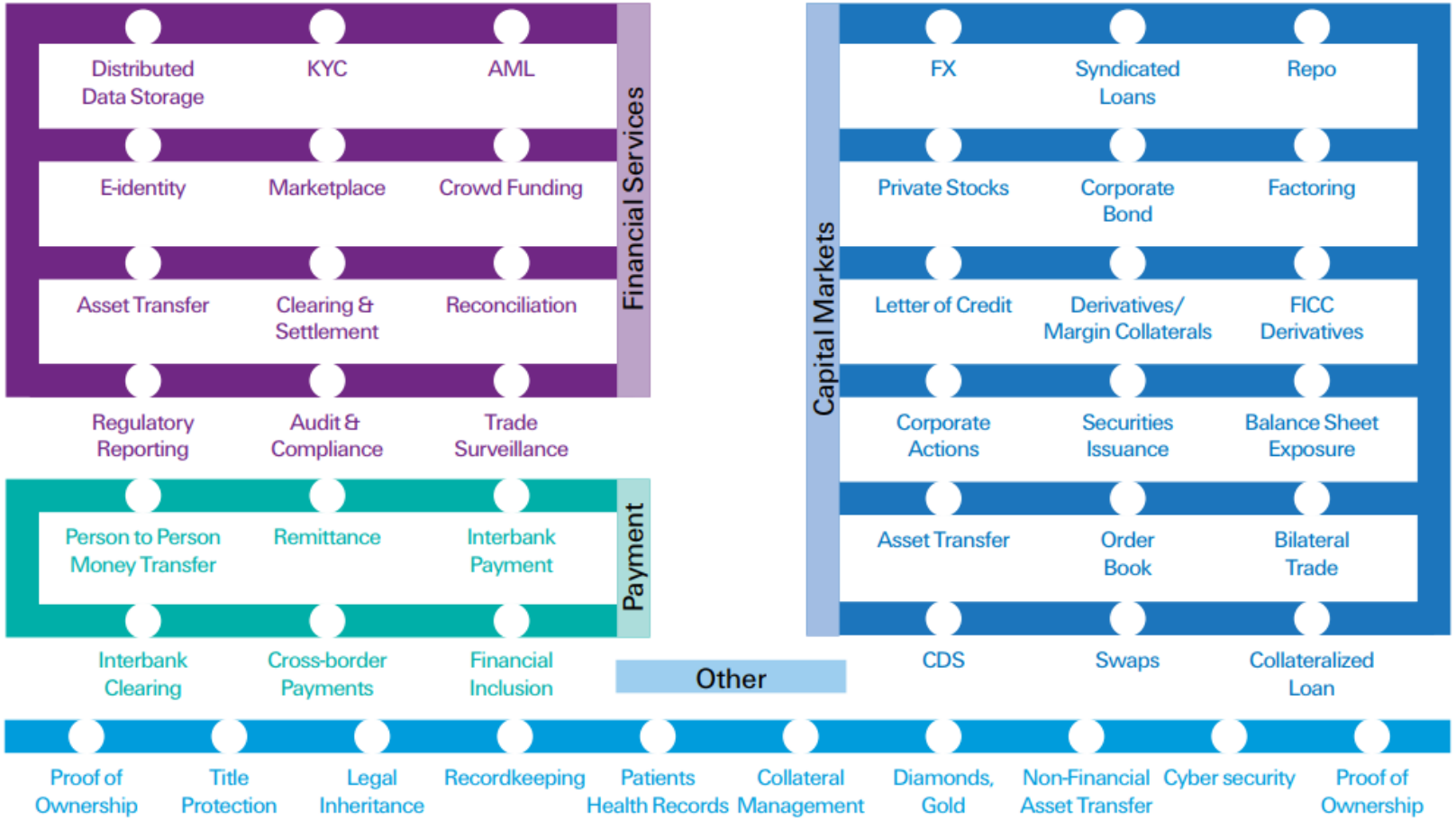
- Proposed by Joan Daemen and Vincent Rijmen in 2001, based on NIST DES with modifications, Advanced Encryption Standard block sizes vary from 128-bit, 192-bit and 256-bit. Most financial institutions use 256-bit AES encryptions.
- During AES algorithm processing, a 4x4 array of bytes known as the state is modified using multiple rounds.
- Full encryption requires 10 to 14 rounds, depending on the size of the key.

Key size	Number of rounds required
128 bit	10 rounds
192-bit	12 rounds
256-bit	14 rounds

SYED AWASE KHIRNI

EXISTING BLOCKCHAIN PLATFORMS

VARIOUS USE CASES

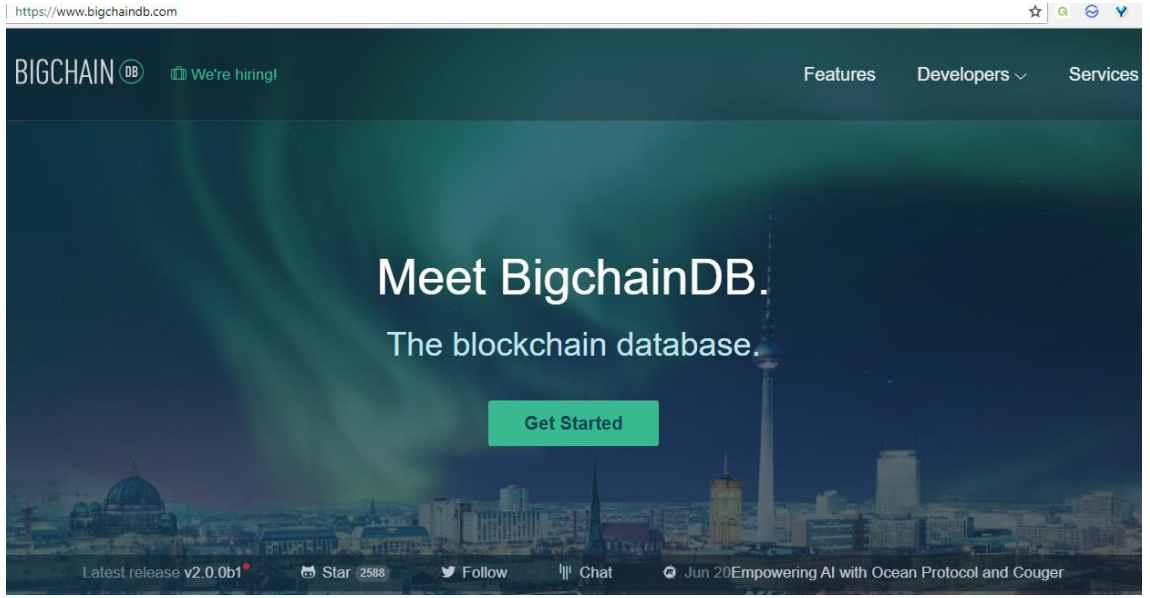


Source: Use cases currently being tested and implemented fig7 pg.15 https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmg-blockchain-consensus-mechanism.pdf

BIGCHAIN

- An open source system that starts with a big data distributed database and then adds blockchain characteristics – decentralized control, immutability and the transfer of digital assets.
  - Each write is recorded on the blockchain database without the need for merkle trees or sidechains.
  - Support for custom assets, transactions, permissions, and transparency.
  - Federation consensus model
  - Supports public and private networks.
  - Has no native currency
  - Set permissions at transaction level.

https://www.bigchaindb.com/

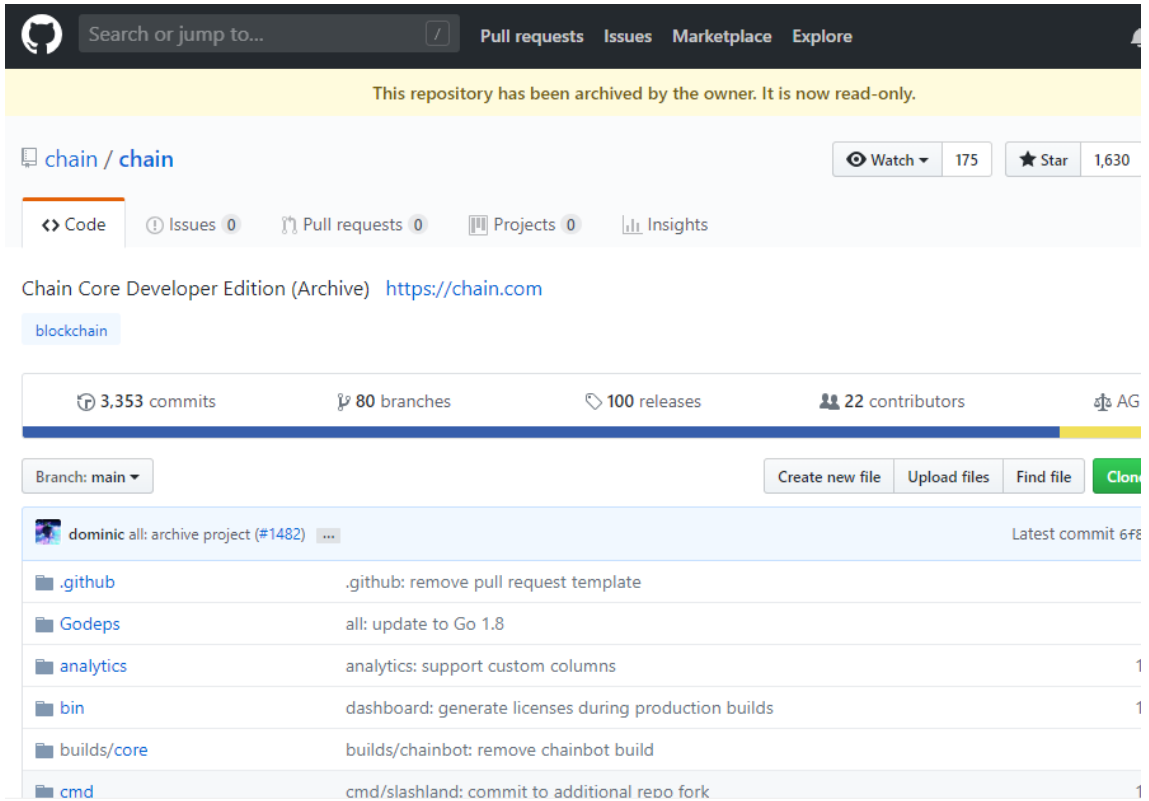


It complements decentralized processing platforms and file systems such as Ethereum and IPFS

CHAIN CORE

- Chain core is software designed to operate and connect to highly scalable permissioned blockchain networks conforming to the chain protocol.
- Each network maintains a cryptographically-secured transaction log known as blockchain, which allows participants to define, issue and transfer digital assets on a multi-asset shared ledger.

https://github.com/chain/chain



ChainCore Features

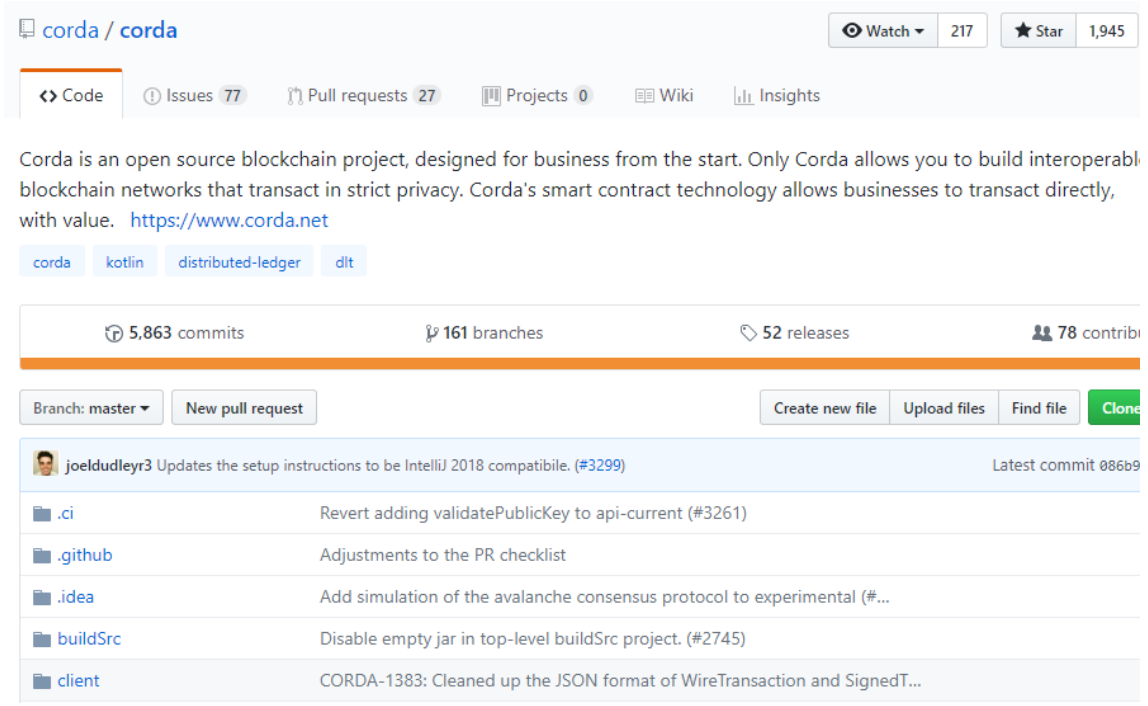
- Native digital assets-currencies, securities etc.
- Role based permissions for operating, accessing and participating in a network
- Support for multi-signature accounts
- Federated consensus
- Support for smart contracts
- Transaction privacy

Corda

. https://www.corda.net/

https://github.com/corda/corda

- An opensource blockchain project designed for business from the start.
- Allows us to build interoperable blockchain networks that transact in strict privacy.
- Smart contract technology allows business to transact directly with value
- Distributed ledger platform with pluggable consensus.



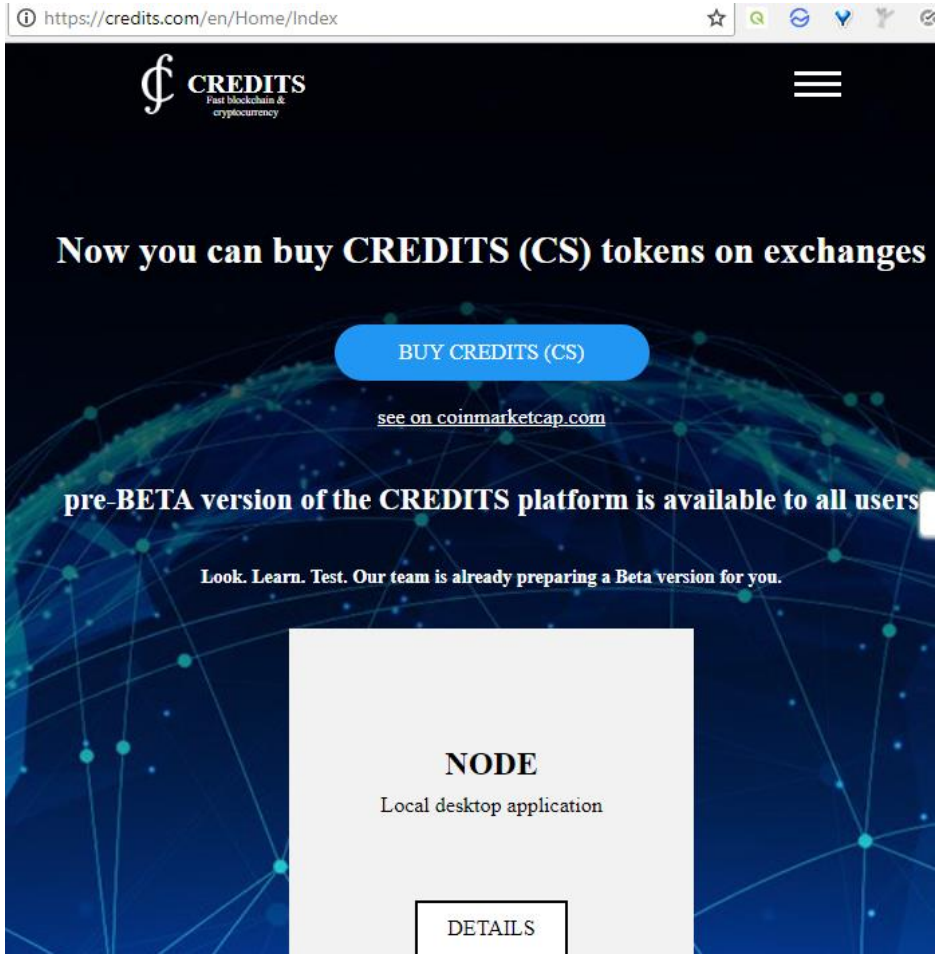
Corda : Features

- No global broadcasting of data across the network
- Pluggable consensus.
- Querying with SQL, join to external databases, bulk imports.

Credits

https://credits.com/en/Home/Index

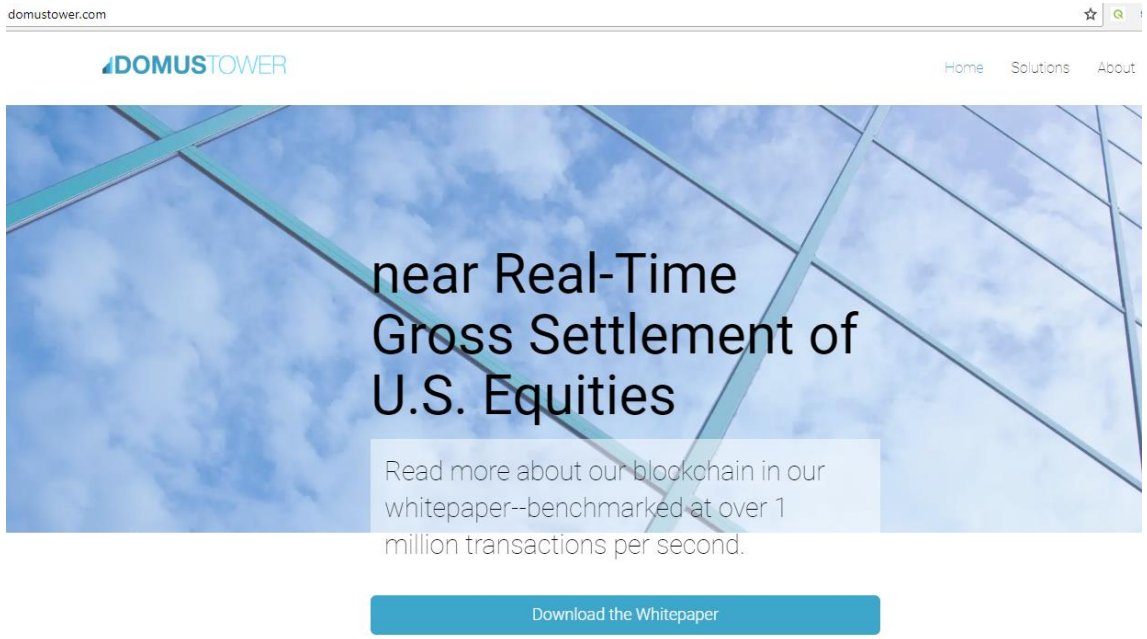
- A development framework for building permissioned distributed ledgers.
- Credits uses a variant of Proof of Stake (a leaderless two-phase commit algorithm with variable voting power)



Domus Tower

http://domustower.com/

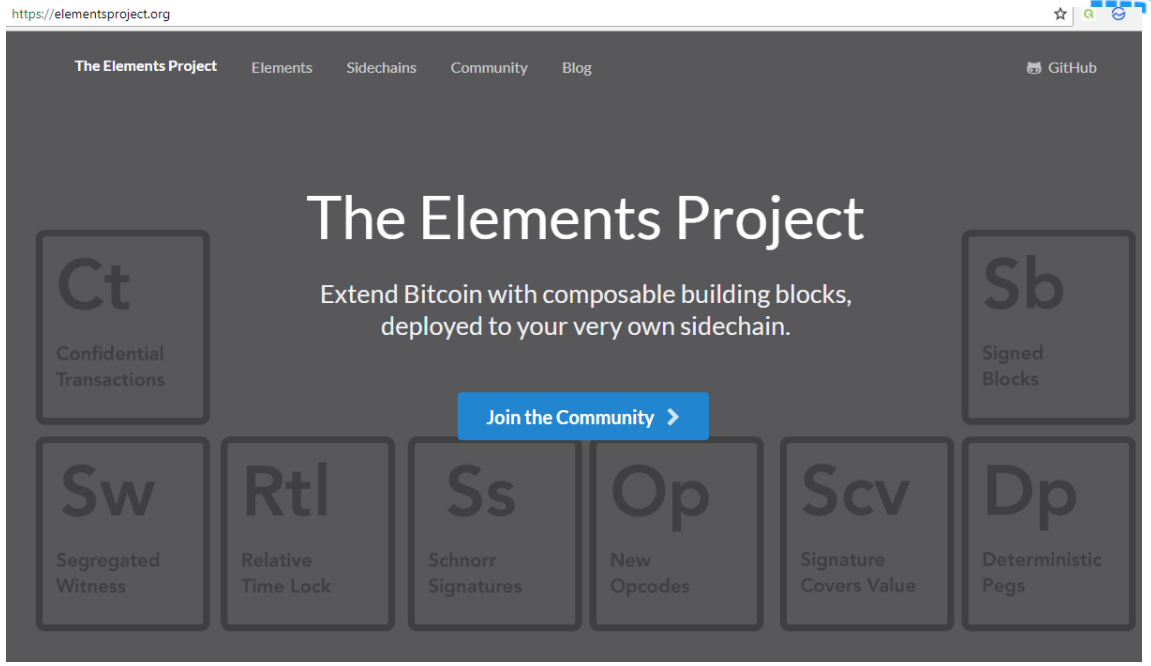
- Designed for regulated environments, benchmarked at ingesting over 1 million transaction per second.
- Creation of linked blockchains where the assets of an account on one blockchain must match the liabilities on the account of another blockchain
- Capability of recording a high rate of transactions in a scalable manner
- Recording of double-entry balance sheet that tracks credits and debits.
- Centralized consensus model



The Elements Project

https://elementsproject.org/

- An open source protocol level technology, developers can use elements to extend the functionality of bitcoin
- Confidential assets- issue multiple assets who's identifiers and amounts are blinded yet auditable.
- Confidential transactions – keep the amounts transferred visible only to participants in the transaction and to designated entities.
- Deterministic pegs- allow cross chain transactions to be constructed in a decentralized fashion.



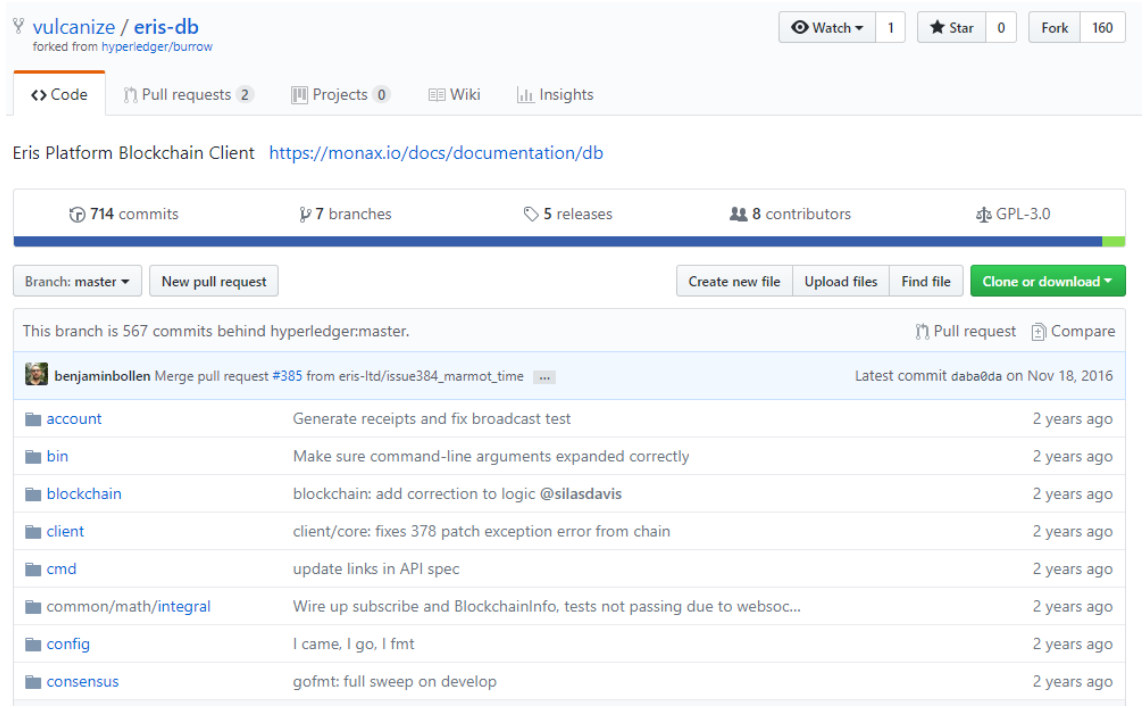
The Elements Project: Features

- Signed Blocks: allows blocks can be cryptographically signed thereby allowing the creator of the block to verify their identity in the future.
- Segregated Witness: Bitcoin transactions contain two things:
  - a. information about the effect on the ledger
  - B.data proving that the transaction is authorized.
- Relative locktime which allows a transaction to be time-locked.
- Additional opcodes

Eris:db

https://monax.io/platform/db/

- an open source, protocol-level technology for extending the functionality of Bitcoin
- Multiple interfaces
- Ethereum Virtual Machine
- Permissioned Systems
- Byzantine fault-tolerant tindermint consensus engine, which is a deposit based proof of stake protocol.



Ethereum

- a decentralized platform that runs smart contracts on a custom built blockchain.
- Ethereum wallet – facilitates holding crypto-assets as well as writing deploying and using smart contracts.
- Creation of crypto-currencies
- Creation of democratic autonomous organizations (DAOs)
- Command line tools built in Go, C++, Python, Java

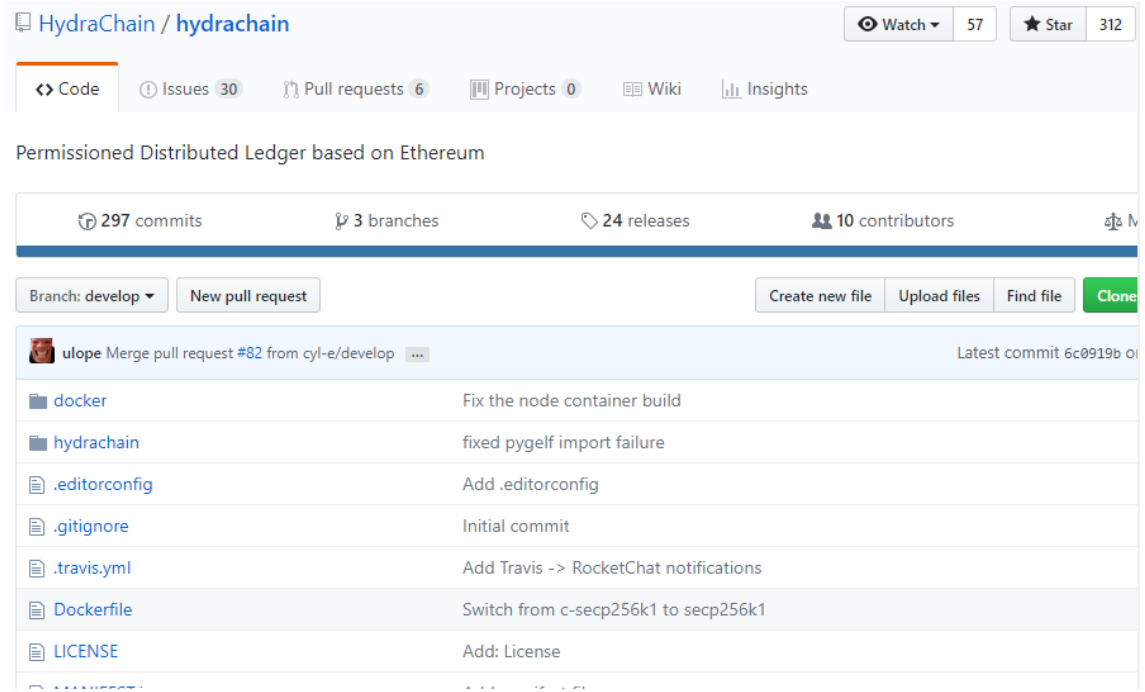
https://www.ethereum.org/

Etash- a proof of work algorithm

Hydrachain

https://github.com/HydraChain/hydrachain

- An ethereum extension for creating permissioned distributed ledgers for private and consortium chains.
- Full compatibility with ethereum protocol
- Account validators
- Instatnt finalty of block and no forks or reverts
- Support for sub-second block times
- New blocks are only created in the presence of pending transactions.



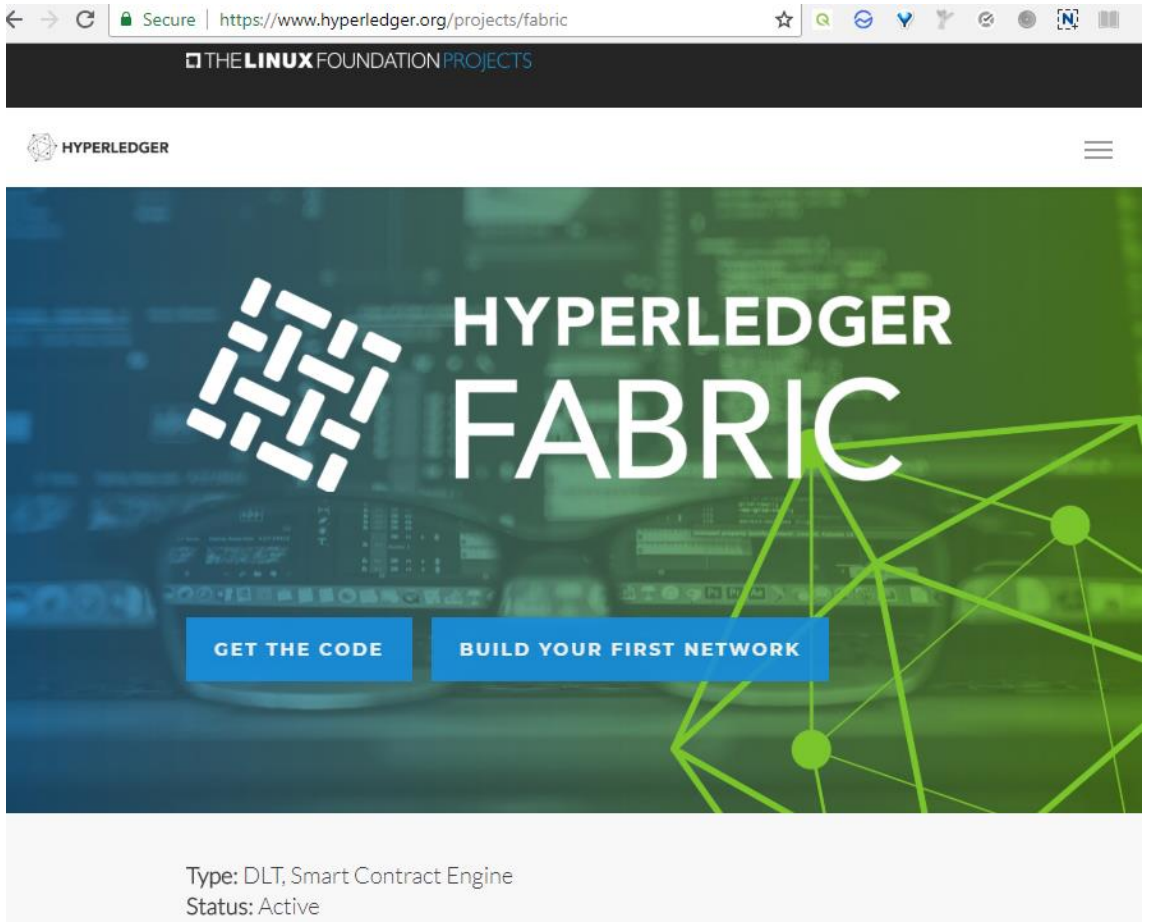
Hydrachain

https://github.com/HydraChain/hydrachain

- Infrastructure for developing smart contracts in python
- Customizability of transaction fees, gas limits, genesis allocation, block time
- Open source.

Hyperledger Fabric

https://www.hyperledger.org/projects/fabric



- It supports the use of one or more networks each managing different assets, agreements and transactions between different sets of member nodes.
- Query and update ledger using key-based lookups, range queries and composite key queries.
- Read-only history queries
- Transactions contain signatures of every endorsing peer and are submitted to ordering service.

Hyperledger Fabric

https://github.com/hyperledger/fabric

- Peers validate transactions against endorsement policies and enforce the policies.
- A channel's ledger contains a configuration block defining policies, access control lists, and other pertinent information.
- Channel's allow crypto materials to be derived from different certificate authorities.
- Consensus is ultimately achieved when the order and result of a block's transaction have met the explicit policy criteria checks.

Hyperledger Iroha

https://github.com/hyperledger/iroha

- A simple and modularized distributed ledger system with emphasis on mobile application development.
- Sumeragi, a Byzantine Fault Tolerant Consensus algorithm heavily inspire by the B-Chain Algorithm.

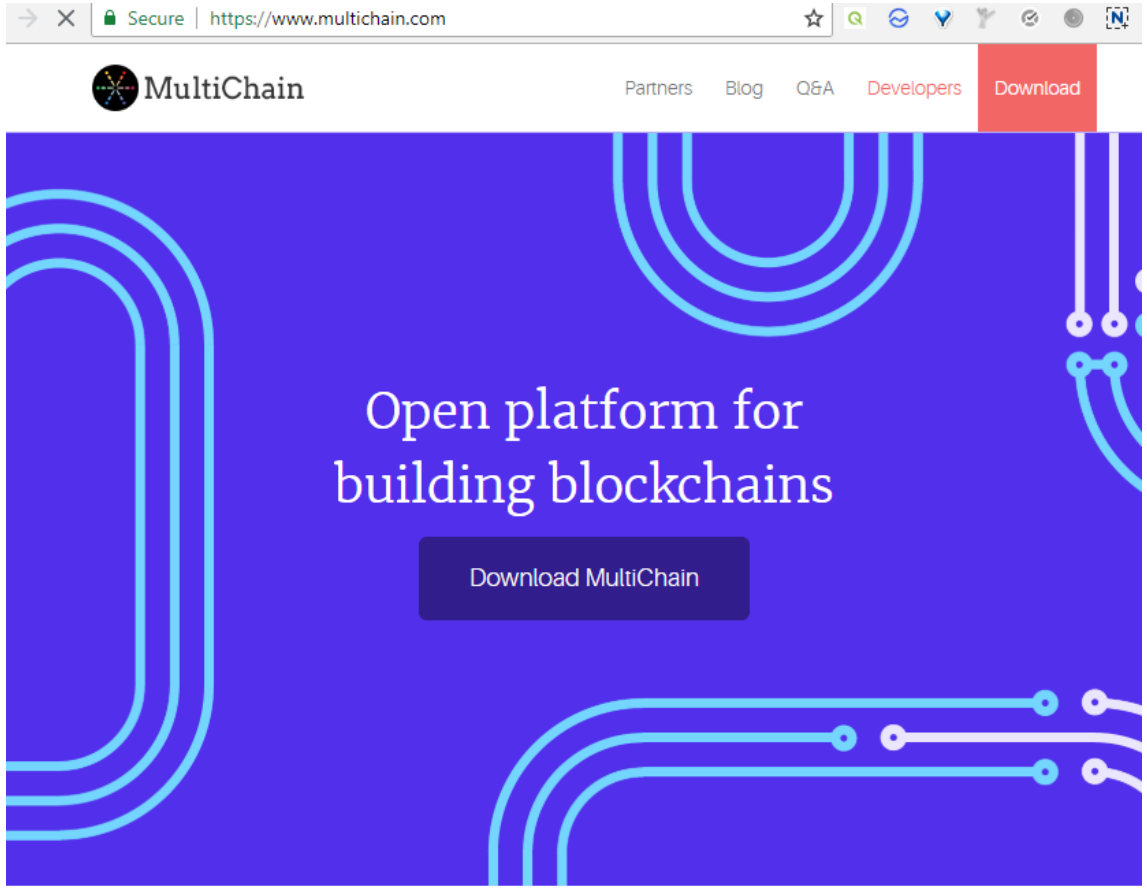
Hyperledge Sawtooth Lake

- A modular blockchain suite that supports both permissioned and permissionless deployments.
- Transaction business logic in hyperledger sawtooth lake is decoupled from the consensus layer
- Consensus mechanism is based on proof of elapsed time.

Multichain

https://www.multichain.com/

- An open-source blockchain platform, based on bitcoin's blockchain for multi-asset financial transactions.
- Native multi-currency support
- Atomic two or multi-way exchanges of assets between participants.
- Permission management.
- Rapid deployment
- Multiple networks can simultaneously be on a single server



Multichain

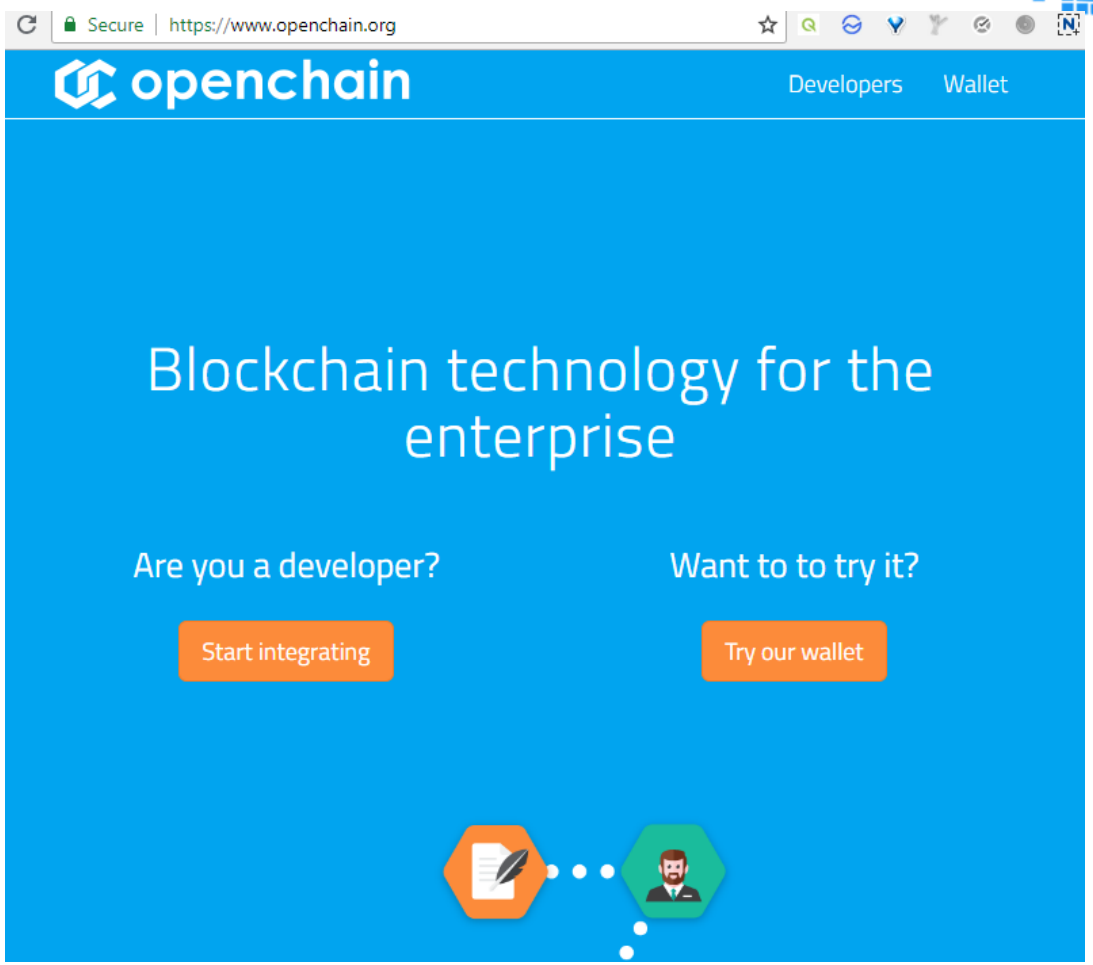
- Per network custom parameter – permitted transaction types, confirmation times, minimum quantities, transaction rate and size limits.
- Data streams.
- Distributed consensus between identified block validators. Similar to Practical Byzantine Fault Tolerance with one validator per block, working in a round-robin type of fashion

OpenChain

- An open source distributed ledger system for issuing and managing digital assets.
- Tokens on open chain can be pegged to bitcoin , making it a side chain.
- Smart contract modules
- Unified API
- Assign aliases to the users instead of using base-58 addresses.
- Multiple levels of control
- Hierarchical account system allowing to set permissions at any level.
- Ability to have multiple open chain instances replicating from each other.

Partitioned Consensus

https://www.openchain.org/



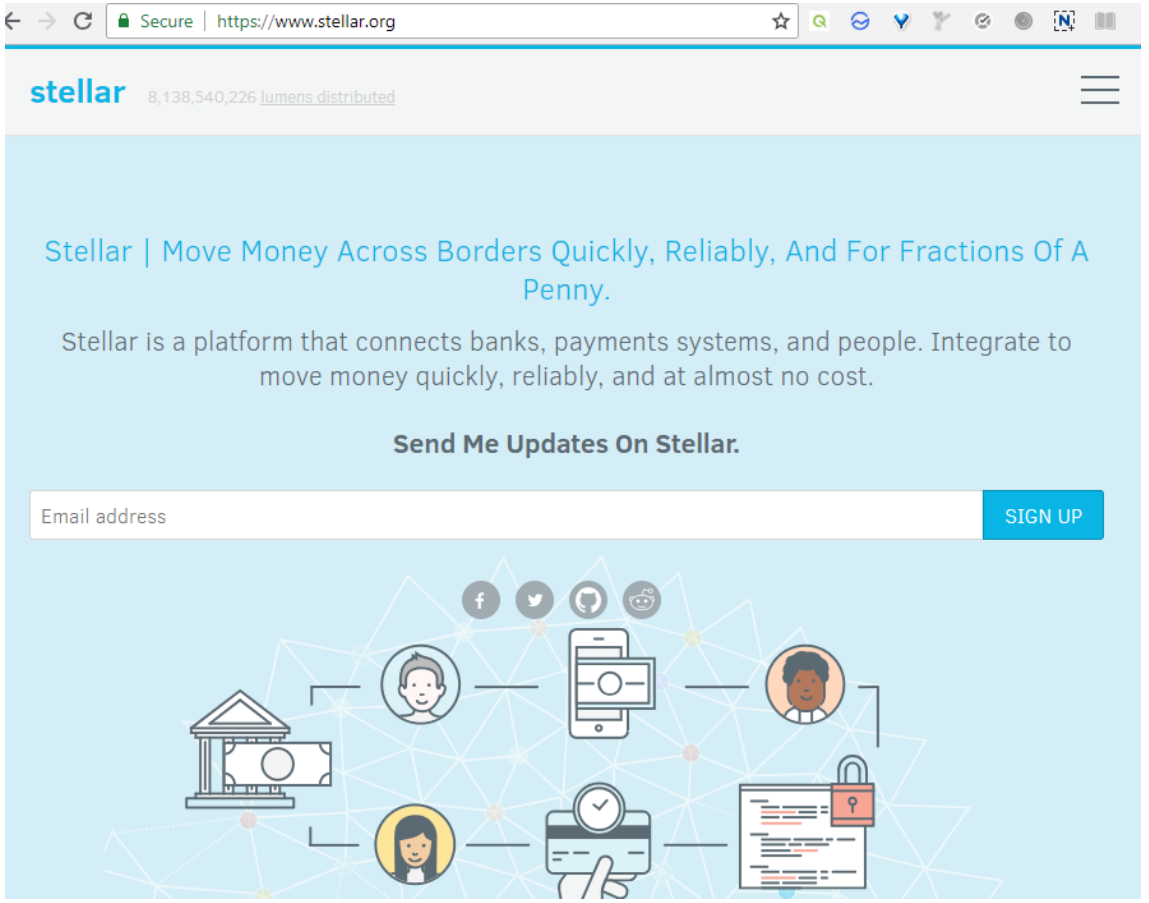
Quorum

- An open source distributed ledger and smart contract platform based on ethereum
- Easy GUI for working with networks, smart contracts and APIs
- Idela for applications requiring high speed and high throughput processing of private transactions.
- Consensus model based on majority voting. Raft based consensus model for faster blocktimes, transaction finality and on-demand block creation.

Stellar

https://www.stellar.org/

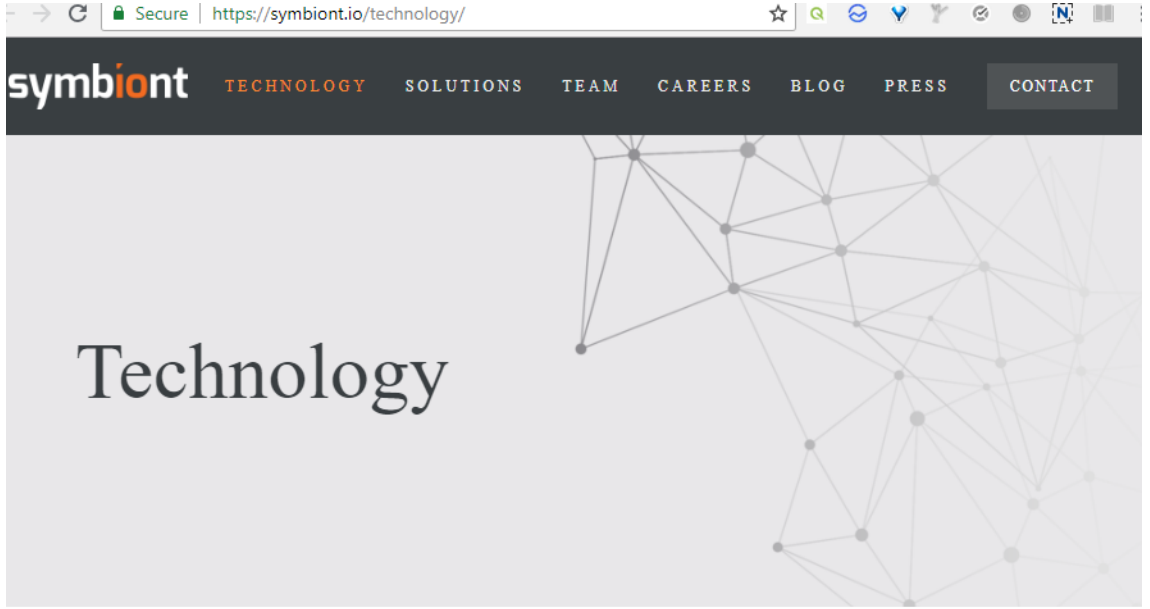
- An open source distributed payments infrastructure that provides RESTful HTTP API servers which connect to Stellar Core, the backbone of the Stellar network
- Stellar consensus protocol



Symbiont

https://symbiont.io/technology/

- Symbiont Assembly is a blockchain platform for building networks in which multiple, independent entities may share data and logic in real time.
- It is a decentralized database that replicates and executes application logic in the form of smart contracts.
- This platform may be used to create financial instruments, such as loans and securities in a digital form from their inception.



Symbiont Assembly is a blockchain platform for building networks in which multiple, independent entities may share data and logic in real time. It is a decentralized database that replicates and executes application logic in the form of smart contracts. This platform may be used to create financial instruments–such as loans and securities–in a digital form from their inception. Assembly was purpose-built to meet the standards of institutional

Symbiont

- Capability to handle thousand of transactions per second
- Assembly API –RESTful standard JSON over HTTP.

www.matrixfan.site

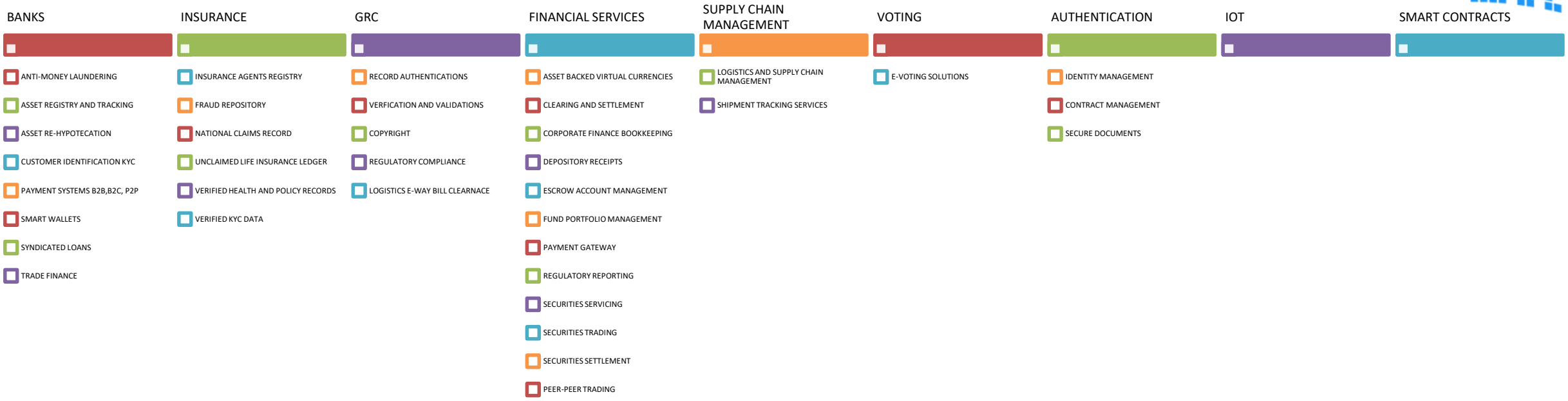
SYED AWASE KHIRNI

BUILDING A BLOCKCHAIN SOLUTION

BUILDING BLOCKCHAIN SOLUTION

1. IDENTIFY THE APPLICATION SECTOR
2. MOST SUITABLE CONSENSUS ALGORITHM/APPROACH
3. IDENTIFY THE SUITABLE PLATFORM
4. DESIGN THE NODE TYPE
5. DESIGN THE BLOCKCHAIN INSTANCE
6. DESIGN THE APPLICATION PROGRAMMING INTERFACE
7. DESIGN THE ADMINSTRATIVE DASHBOARDS AND GUI

1.IDENTIFY THE APPLICATION SECTOR



2.MOST SUITABLE CONSENSUS APPROACH

1. PROOF OF WORK
2. PROOF OF STAKE
3. BYZANTINE FAULT TOLERANT
4. DEPOSIT BASED CONSENSUS
5. FEDERATED BYZANTINE AGREEMENT
6. PROOF OF ELAPSED TIME
7. DERIVED PBFT
8. REDUNDANT BYZANTINE FAULT TOLERANCE
9. SIMPLIFIED BYZANTINE FAULT TOLERANCE
10. FEDERATED CONSENSUS
11. ROUND ROBIN
12. DELEGATED PROOF OF STAKE

3.IDENTIFY SUITABLE PLATFORM

BIGCHAINDB

CHAINCORE

CORDA

CREDITS

ELEMENTS

ERIS:DB

ETHEREUM

HYDRACHAIN

HYPERLEDGER (CELLO/FABRIC/IROHA /SAWTOOTHLAKE)

MULTICHAIN

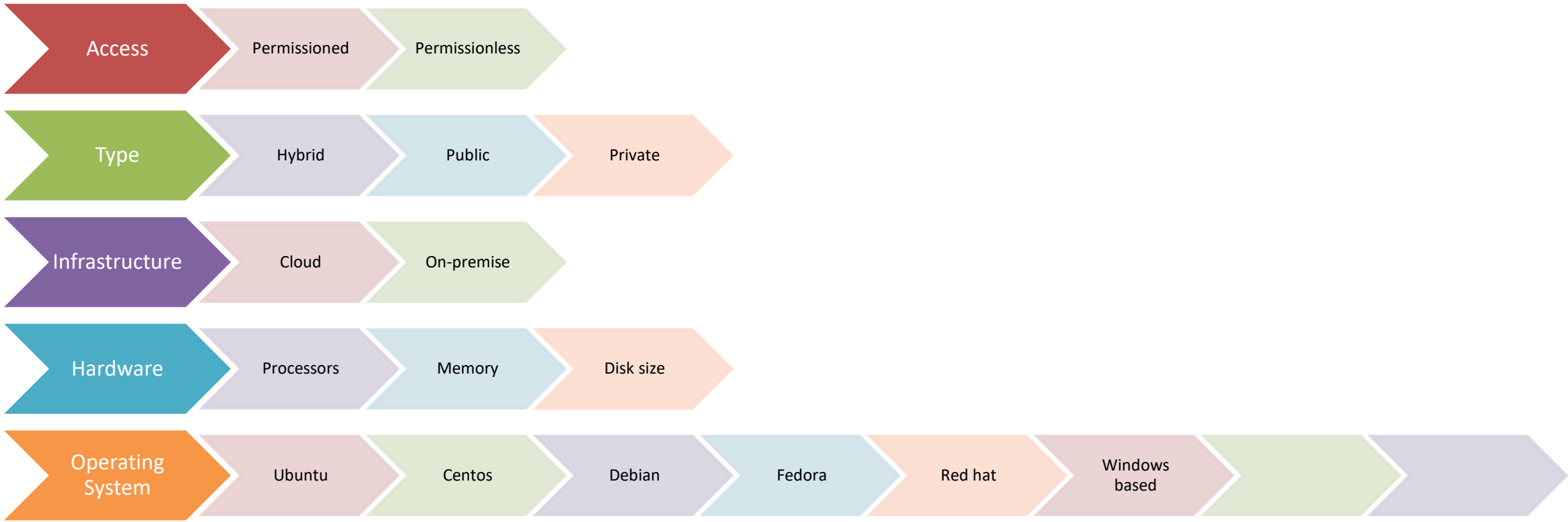
OPENCHAIN

QUORUM

STELLAR

SYMBIONT

4.DESIGN THE NODE TYPE



5.DESIGN THE BLOCKCHAIN INSTANCE

- Permissions based
- Asset re-issuance
- Atomic exchanges
- Key management
- Multi-signatures
- Parameters
- Native assets
- Address formats
- Key formats
- Block signatures
- Hand-shaking

6.DESIGN THE APPLICATION PROGRAMMING INTERFACE

- Address based
- Audit based
- Data authentication system
- Data storage system
- Data streams based
- Digital signature
- Secure messaging
- Smart assets
- Smart contracts
- Transactions based

7.DESIGN THE ADMINSTRATIVE DASHBOARD

- HTML5 CSS +
  - PHP
  - C#
  - Java
  - Javascript
  - Python
  - Golang
  - Solidity
  - Express
  - AngularJS
  - Apache
  - FTP Server
  - Databases
    - MySQL
    - MongoDB

SYED AWASE KHIRNI

BLOCKCHAIN APPLICATIONS

BLOCKCHAIN CONSORTIA FOR R&D

ENTERPRISE ETHEREUM ALLIANCE(EEA)

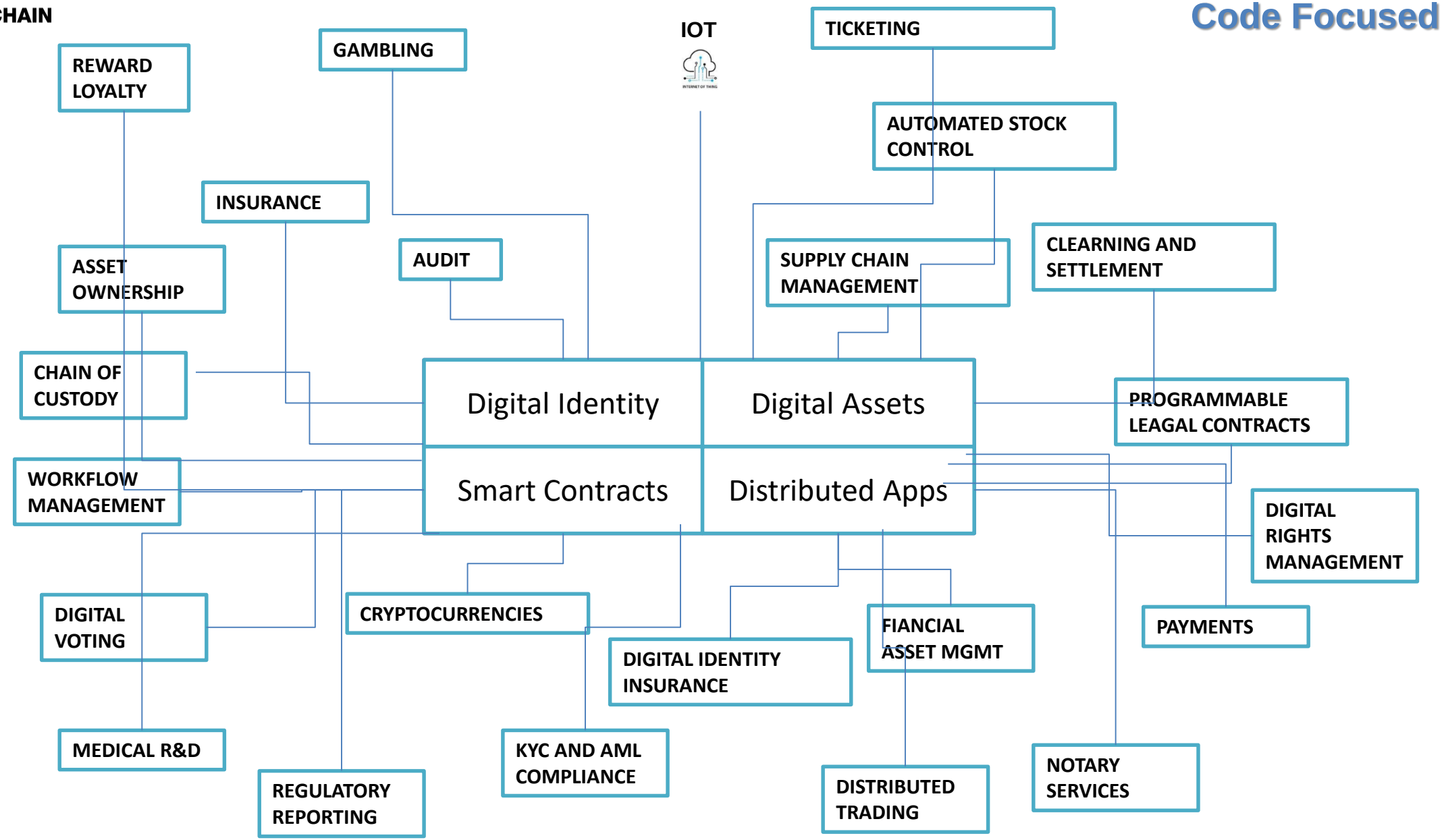
NOTES

HYPERLEDGER

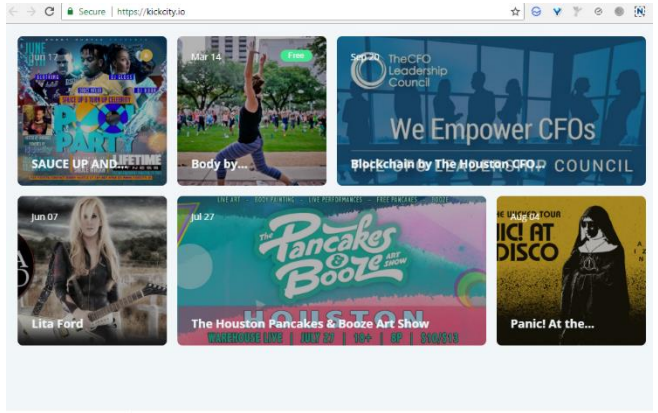
NOTES

R3

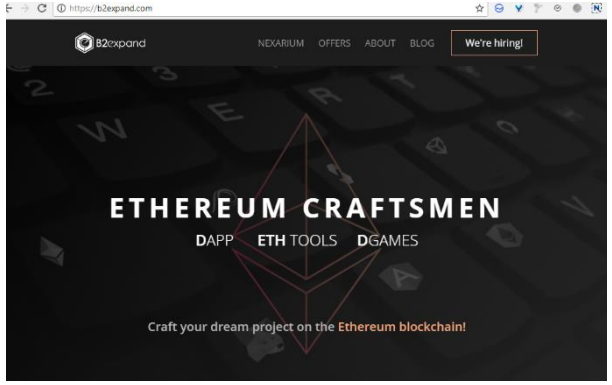
NOTES



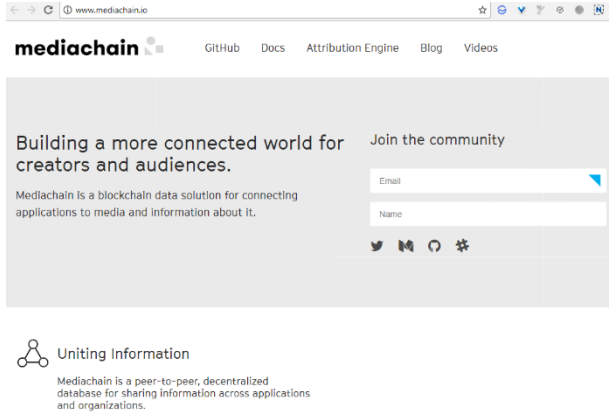
ENTERTAINMENT



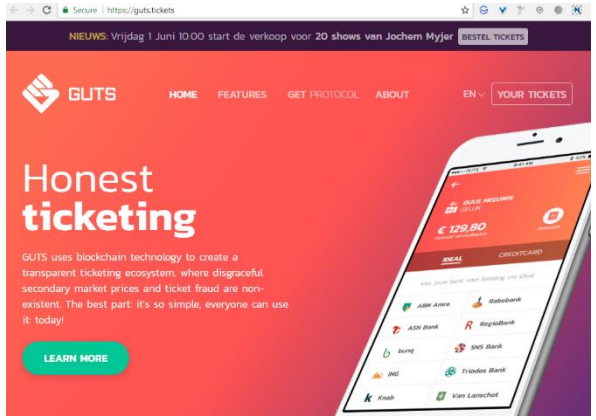
WWW.KICKCITY.IO



https://b2expand.com/

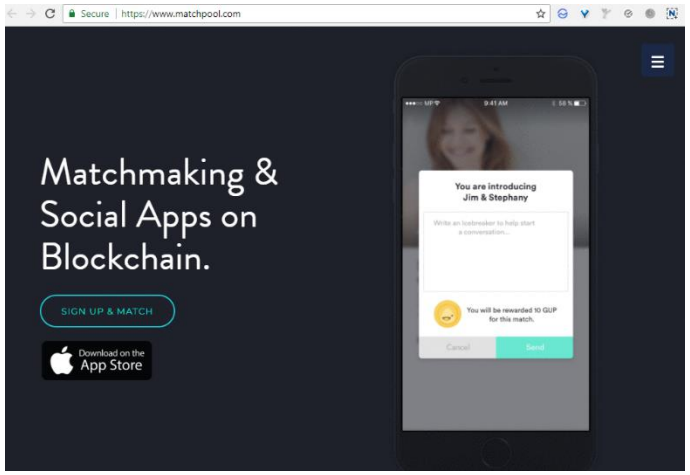


http://www.mediachain.io/

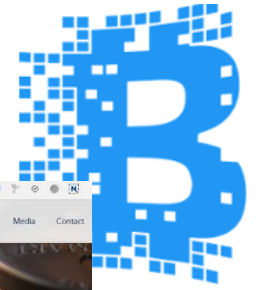


https://guts.tickets/

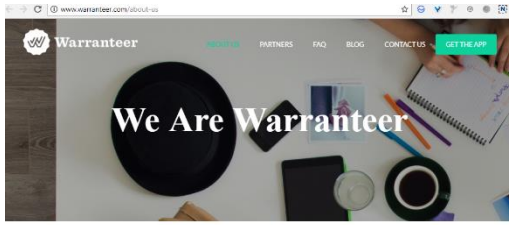
SOCIAL ENGAGEMENT



https://www.matchpool.com/

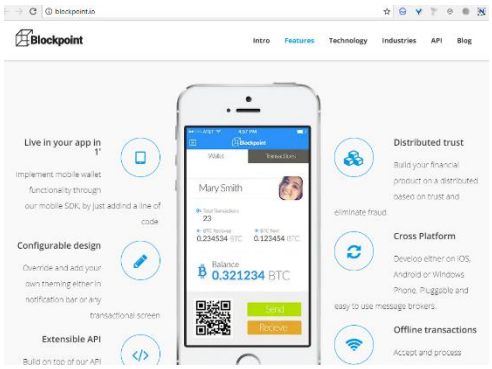


RETAIL

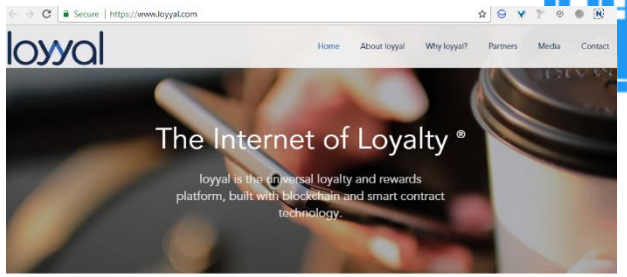


Setting the eWarranty standard

http://www.warranteer.com/about-us

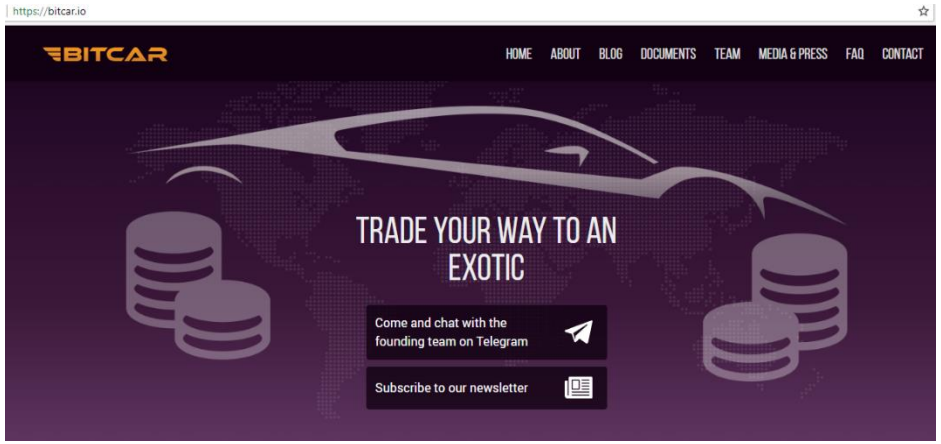


http://blockpoint.io/



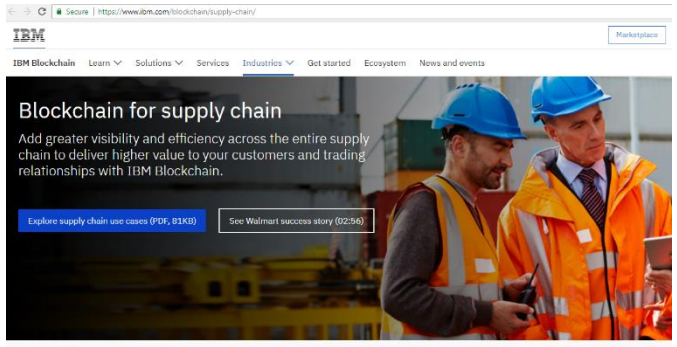
https://www.loyyal.com/

EXOTIC CARS



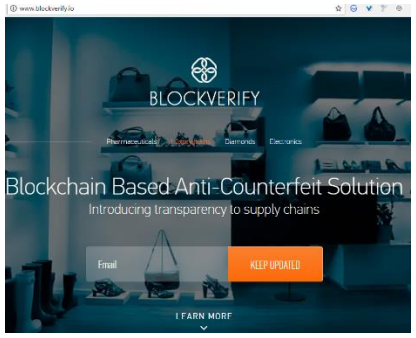
https://bitcar.io/

SUPPLY CHAIN AND LOGISTICS

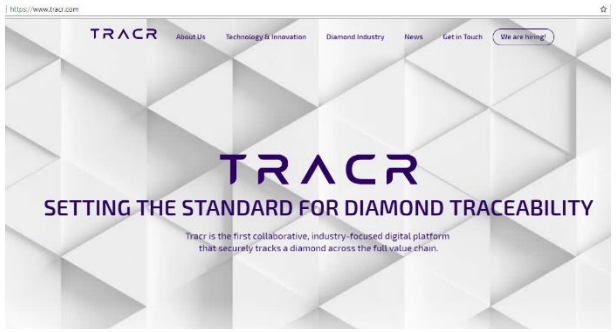


https://www.ibm.com/blockchain/supply-chain/

https://origintrail.io/



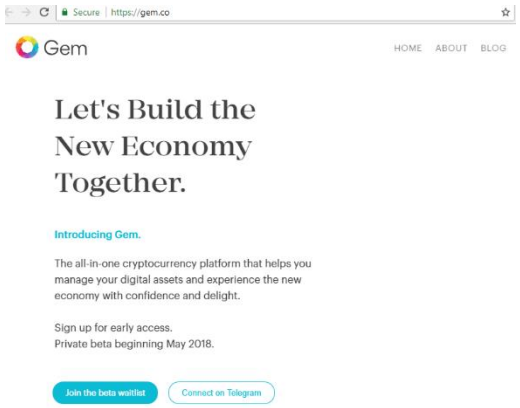
http://www.blockverify.io/



https://www.tracr.com/

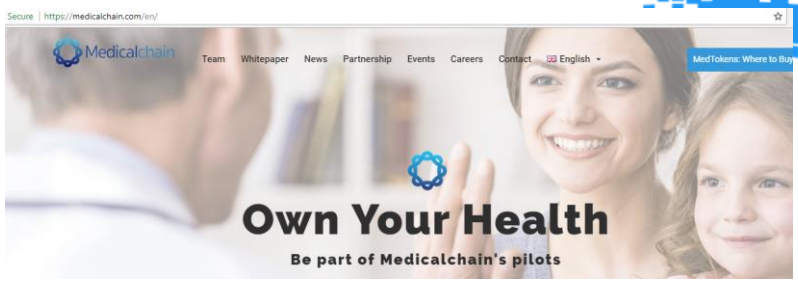
INSURANCE

HEALTHCARE

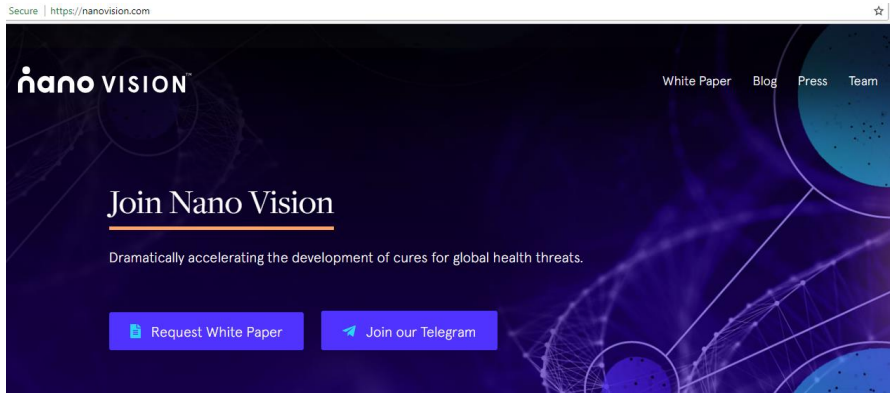


https://gem.co/

https://www.simplyvitalhealth.com/

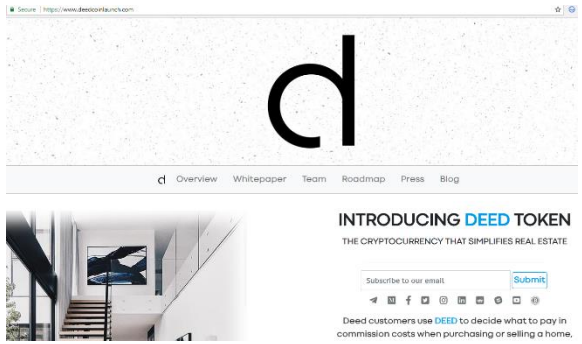


https://medicalchain.com/en/

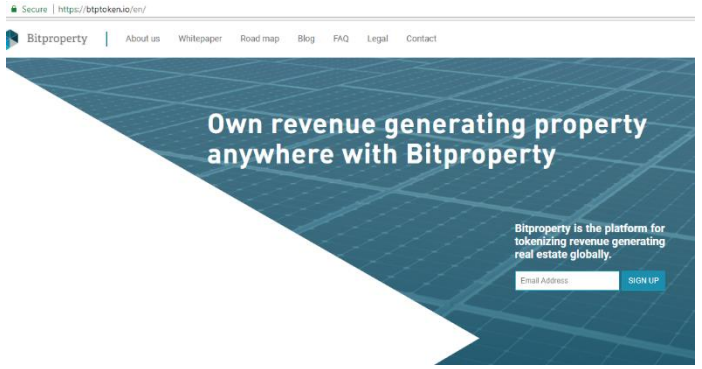


https://nanovision.com/

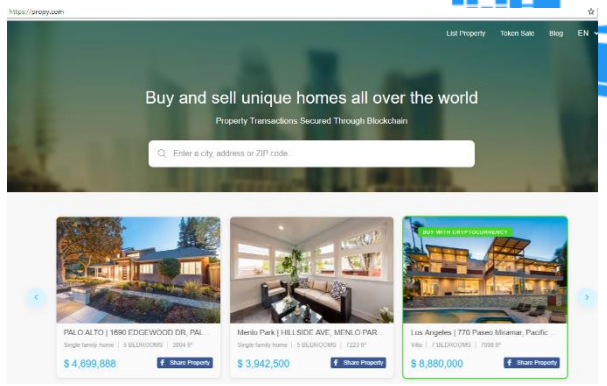
REAL ESTATE



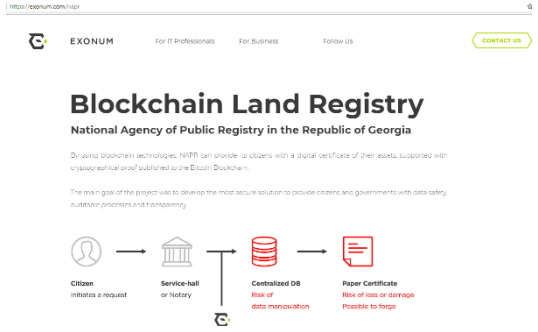
https://www.deedcoinlaunch.com/



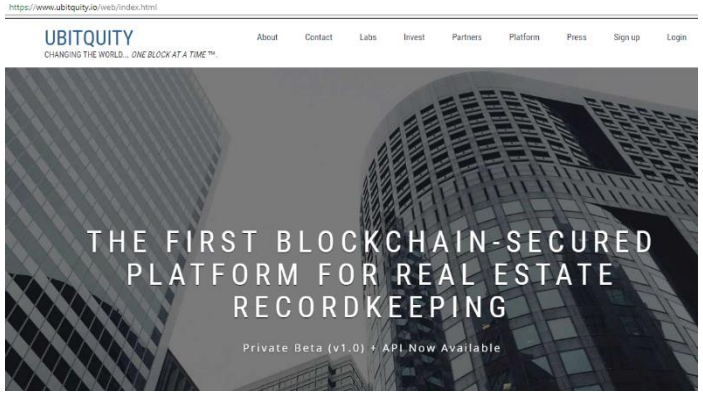
https://btptoken.io/en/



https://propy.com/

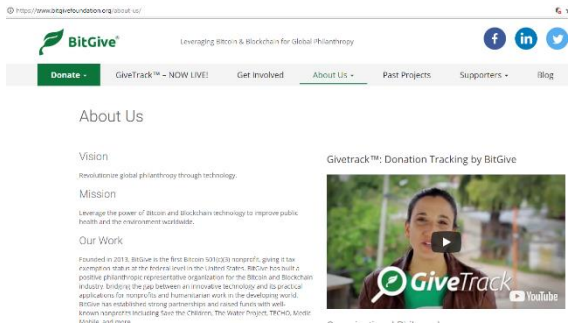


https://exonum.com/napr

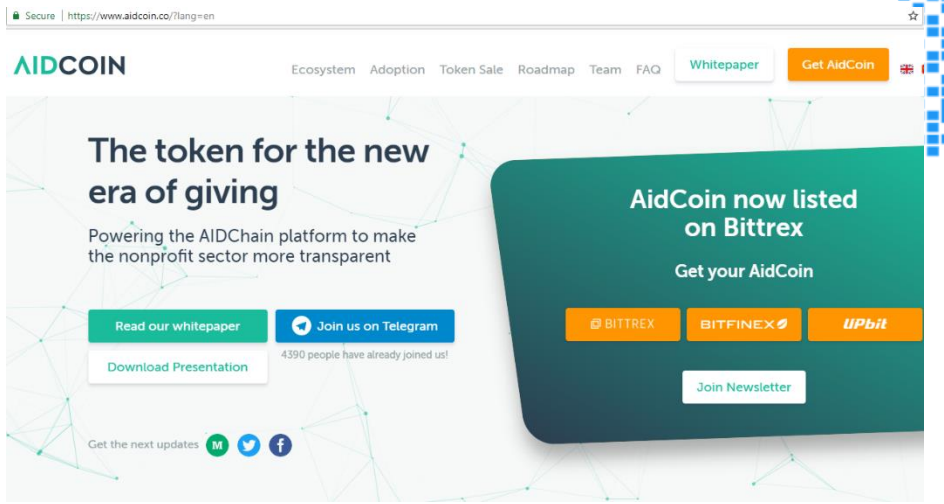


https://www.ubitquity.io/web/index.html

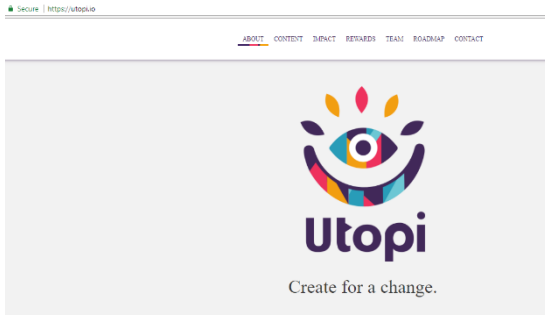
CHARITY



https://www.bitgivefoundation.org/about-us/

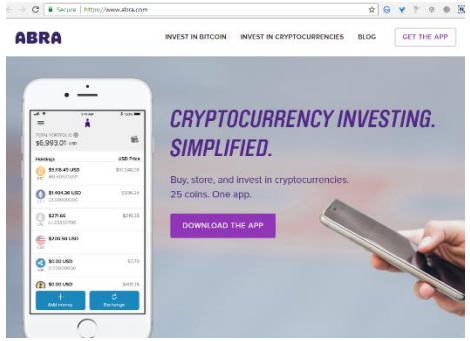


https://www.aidcoin.co/?lang=en

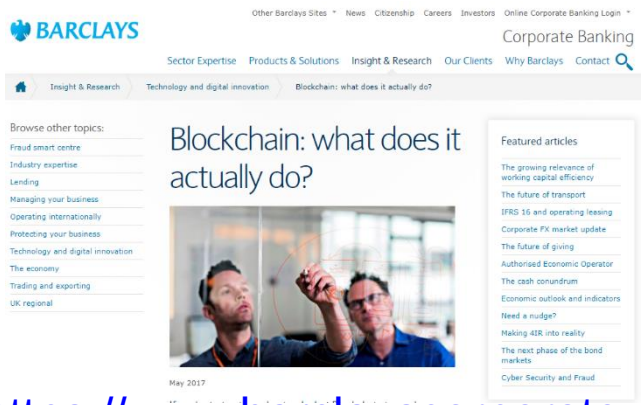


https://utopi.io/

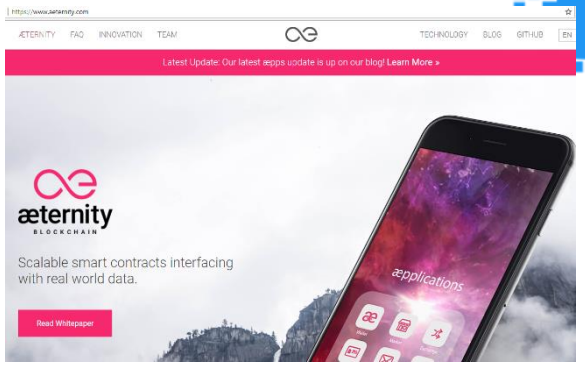
FINANCIAL SERVICES



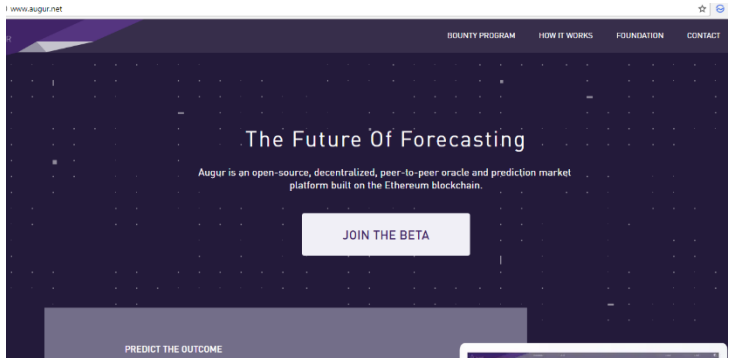
https://www.abra.com/



https://www.barclayscorporate.com

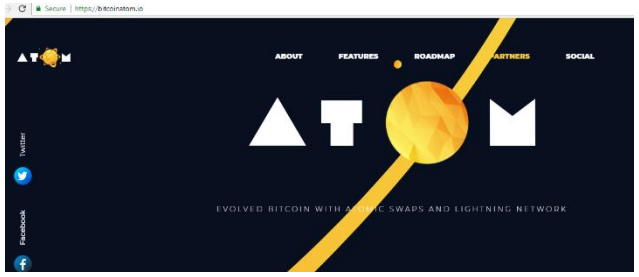


https://www.aeternity.com/



http://www.augur.net/

https://securrency.com/



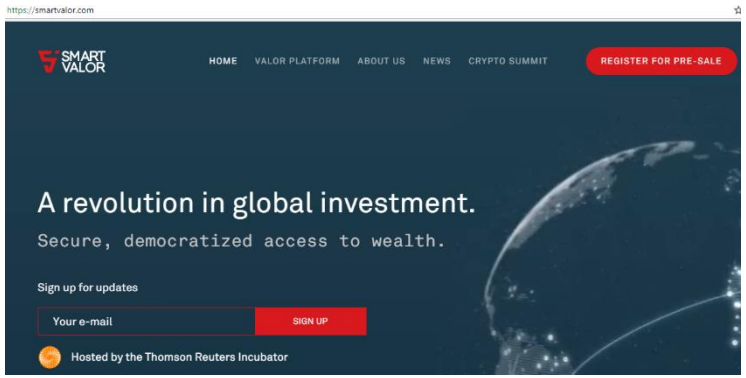
https://bitcoinatom.io/

Please read terms and conditions of use

Original Series

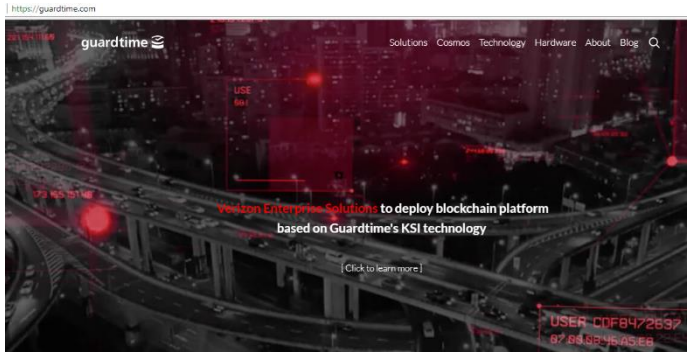
https://ripple.com/

https://www.circle.com/en-gb/

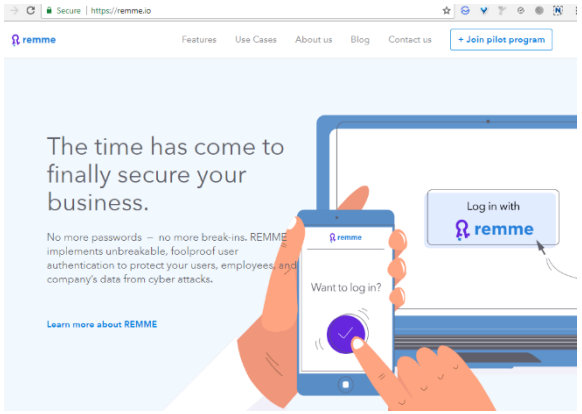


https://smartvalor.com/

CYBERSECURITY

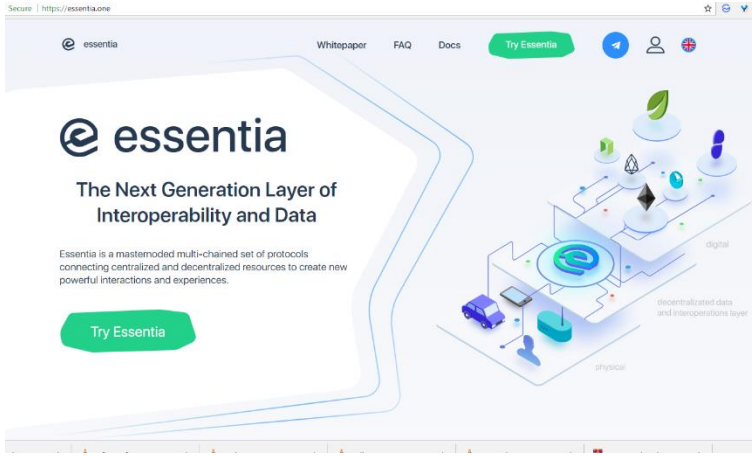


https://guardtime.com/



https://remme.io/

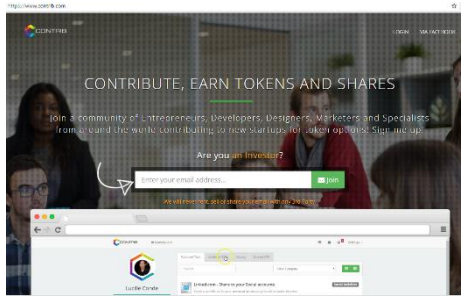
SECURITY



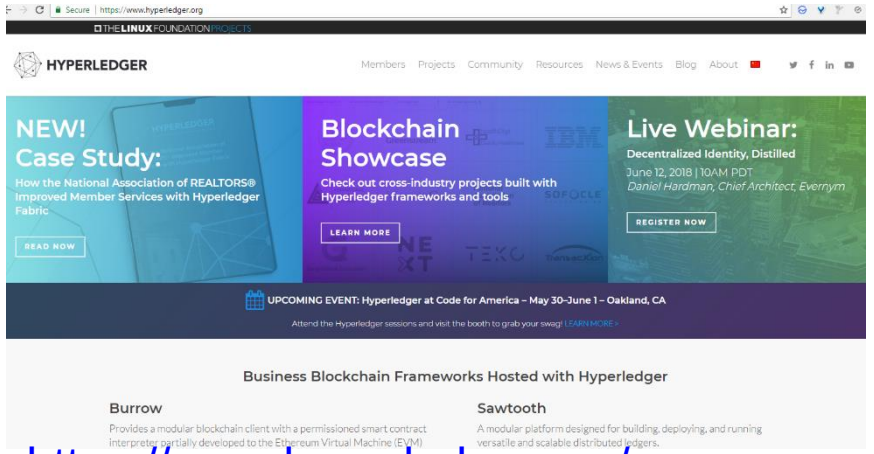
https://essentia.one/

Mining

- Mining is the process of recording the block transactions into the blockchain of that currency.
- Miners use computing power to identify a nonce (number used only once) through which they can create the digest/signature of the next block in the blockchain, which is less than the recently added block.
- Each miner is awarded a virtual coin like bitcoin, ethereum etc.. for the computation.



https://www.contrib.com/



https://www.hyperledger.org/

https://www.nyiax.com/

SECTION:

SUMMARY

BLOCKCHAIN

A public distributed ledger of information collected through a peer-to-peer network, that tracks digital assets. The way it records this information gives blockchain its groundbreaking potential. Instead of using a third party mediator, it relies on the peers/nodes to build a consensus based algorithm to validate the accuracy of the information, in a transparent way.

A new way of documenting data on the internet.

Used to develop applications such as social networks, messengers, games, exchanges, storage platforms, voting systems, prediction markets, online shops.

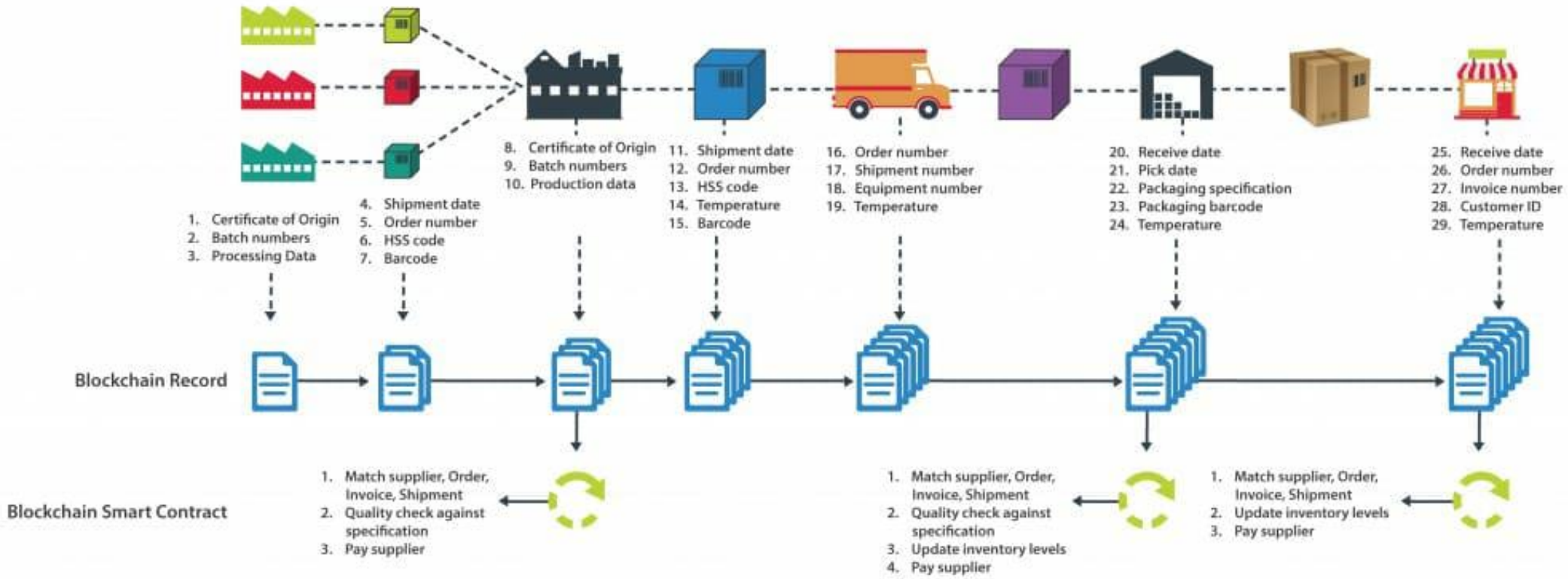
BLOCKCHAIN

- The information recorded on a blockchain can take on any form
  - Transfer of money,
  - Ownership
  - Transaction
  - Identity
  - Agreement between two parties

- A block chain has multiple copies of the same data stored in different nodes, which means there is no single point of failure or malicious transaction.

SYED AWASE KHIRNI

LOGISTICS BLOCKCHAIN



Source:

SYED AWASE KHIRNI

FINANCIAL SERVICES IN BLOCKCHAIN

USECASE SCENARIOS

- DISTRIBUTED DATA STORAGE
- KNOW YOUR CUSTOMER
- AML
- E-IDENTITY
- MARKETPLACE
- CROWD FUNDING
- ASSET TRANSFER
- CLEARING AND SETTLEMENT
- RECONCILIATION
- REGULATORY REPORTING
- AUDIT AND COMPLIANCE
- TRADE SURVEILLANCE

SYED AWASE KHIRNI

PAYMENT IN BLOCKCHAIN

USECASE SCENARIOS

- PERSON TO PERSON MONEY TRANSFER
- REMITTANCE
- INTERBANK PAYMENT
- INTERBANK CLEARING
- CROSS-BORDER PAYMENTS
- FINANCIAL INCLUSION

SYED AWASE KHIRNI

CAPITAL MARKETS IN BLOCKCHAIN

USECASE SCENARIOS

- FOREIGN CURRENCY EXCHANGE
- SYNDICATED LOANS
- REPO
- PRIVATE STOCKS
- CORPORATE BOND
- FACTORING
- LETTER OF CREDIT
- DERIVATIES/MARGIN COLLATERALS
- FICC DERIVATIVES
- CORPORATE ACTIONS
- SECURITIES ISSUANCE
- BALANCE SHEET EXPOSURE
- ASSET TRANSFER
- ORDER BOOK
- BILATERAL TRADE
- CDS
- SWAPS
- COLLATERALIZED LOAN

SYED AWASE KHIRNI

HEALTCH CARE IN BLOCKCHAIN

USECASE SCENARIOS

- PATIENT HEALTH RECORDS
- RECORD KEEPING

SYED AWASE KHIRNI

LEGAL IN BLOCKCHAIN

USECASE SCENARIOS

- PROOF OF OWNERSHIP
- TITLE PROTECTION
- LEGAL INHERITANCE
- RECORD KEEPING
- COLLATERAL MANAGEMENT
- DIGITAL RIGHTS MANAGEMENT
- ASSET TRANSFER / NON-FINANCIAL ASSET TRANSFER

SYED AWASE KHIRNI

SECURITY IN BLOCKCHAIN

USECASE SCENARIOS

- CYBER SECURITY

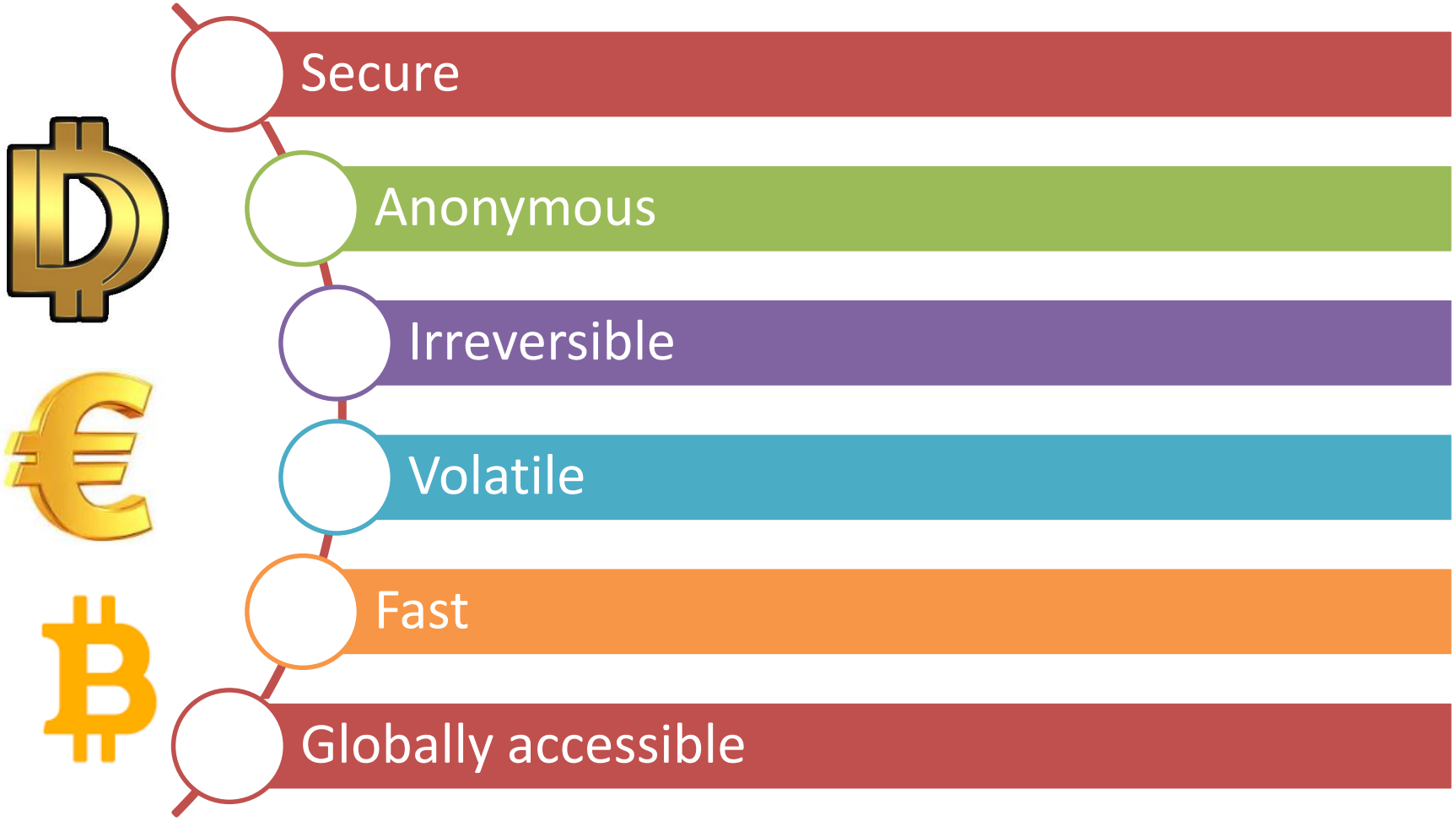
SYED AWASE KHIRNI

CRYPTOCURRENCY USING BLOCKCHAINS

Money

- A basic accounting systems that illustrates
  - Who owns what
  - who has what
  - Who owes what to whom.
- Cryptocurrency: a tradeable digital asset or digital form of money built on block chain technology that only exists online. They rely on cryptography to verify and secure transactions.
- Money has taken many forms through its evolution to this day
  - Coins
  - Paper money
  - Bank promissory notes
  - Debit/Credit Cards –Plastic Money
  - Digital Cash
  - Bitcoin/Ethereum Ethers

Cryptocurrency characteristics



SYED AWASE KHIRNI

TOKEN CLASSIFICATION FRAMEWORK

-THOMAS EULER (18 JAN 2018), *Prof. Dr. Andranik Tumasjan Dr. Oliver Krause, Dr. Karl-Michael Henneking and Daniel Pichler.*

Evaluation Criteria

- Various existing token types
- Classification and analysis of tokens in various dimensions
- Foster better understanding of crypto tokens

Classifying Tokens in Five Dimensions

PURPOSE

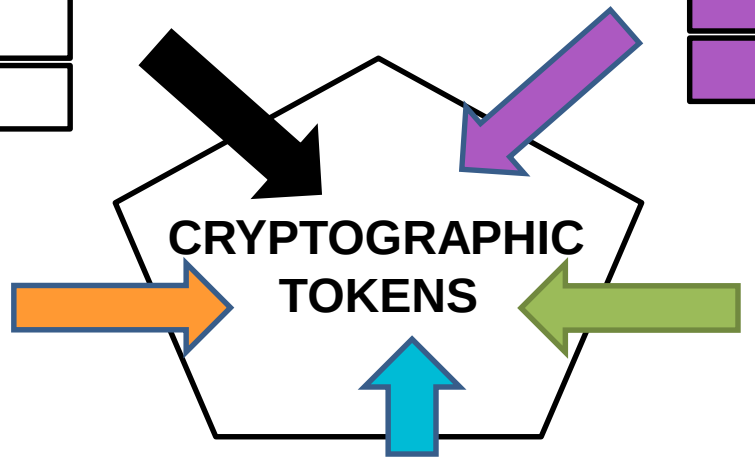
What is the token's main purpose?

- cryptocurrencies
- Network Tokens
- Investment Tokens

TECHNICAL LAYER

On Which system layer is the token implemented?

- Blockchain-Native Tokens
- Non-Native Protocol Tokens
- (d) App Tokens



UNDERLYING VALUE

Where does the token derive its value from ?

- Asset-backed Tokens
- Network Value Tokens
- Share-like Tokens

What utility does the token provide?

- Usage Tokens
- Work Tokens
- Hybrid Tokens

LEGAL STATUS

What is the token's legal status?

- Utility Tokens
- Security Tokens
- Cryptocurrencies

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Purpose

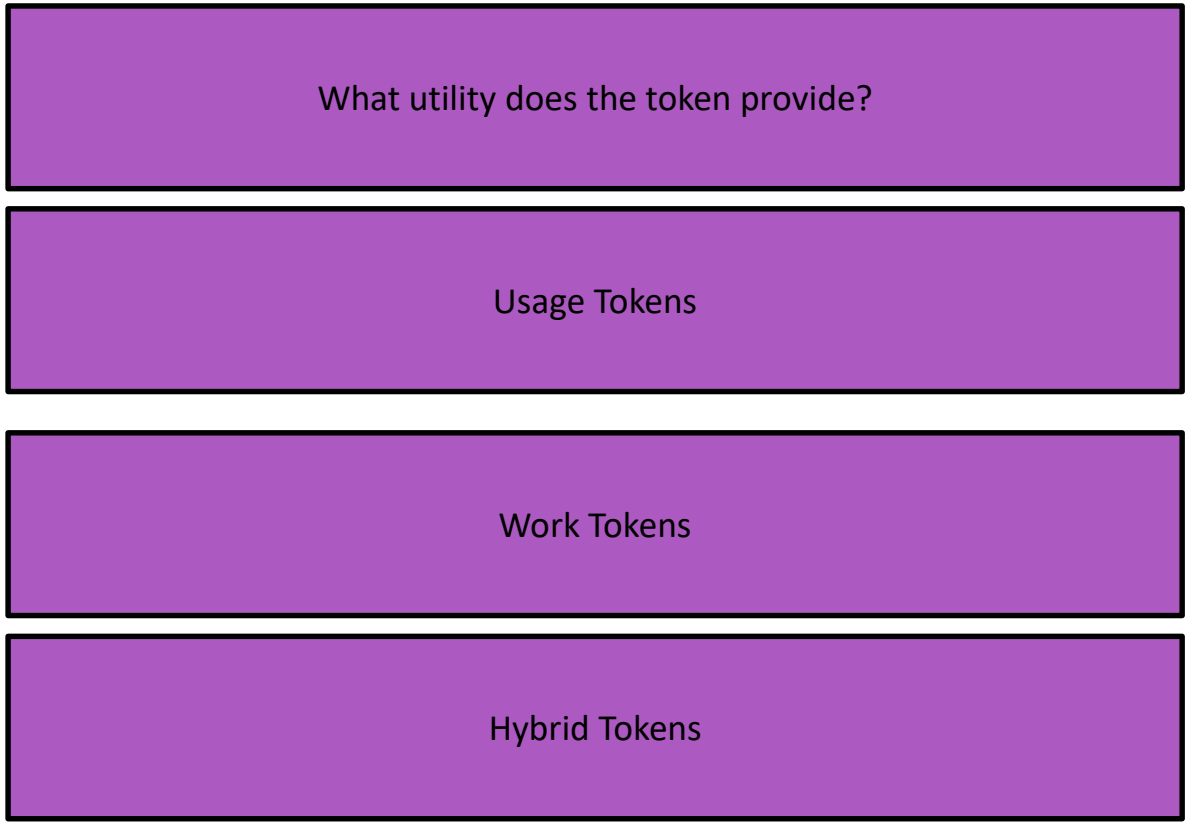
- This dimension illustrates why the various users define it as cryptocurrency, as a token can be meant to enable a specific network and catalyze its growth (network tokens) or it can represent an investment in an entity or an asset.

PURPOSE
What is the token's main purpose?
cryptocurrencies
Network Tokens
Investment Tokens

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Utility

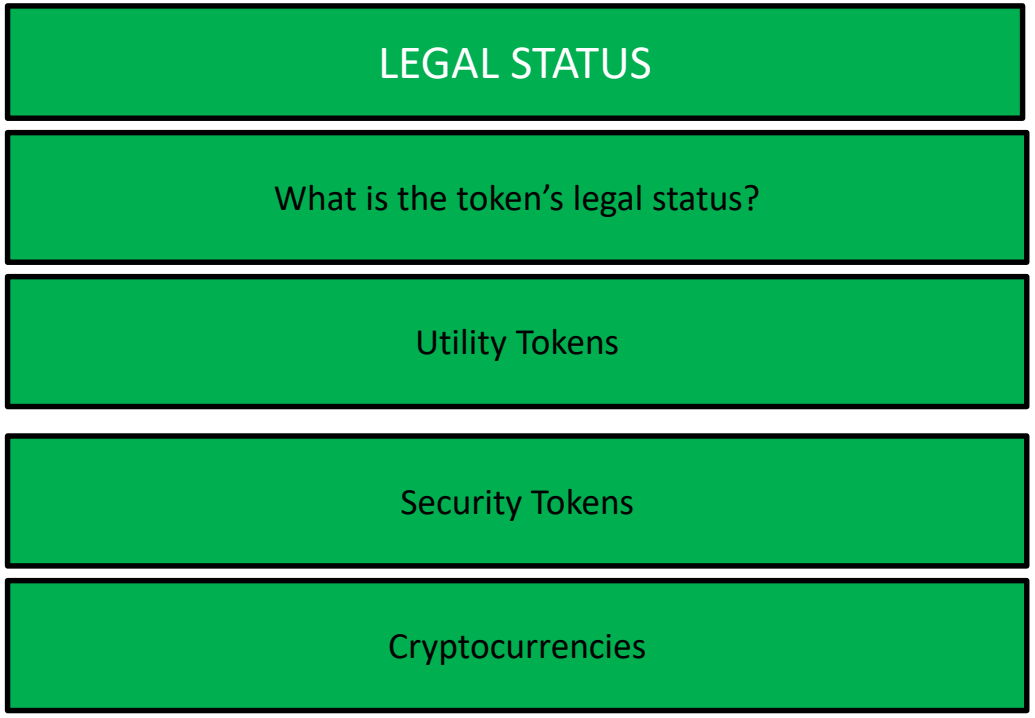
- There are two major ways to provide utility
  - By giving access to network or service features (usage tokens)
  - By allowing token holders to actively contribute work to the system (work token)
  - Some tokens do both (hybrid tokens) and some tokens don't provide any utility at all.



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Legal Status

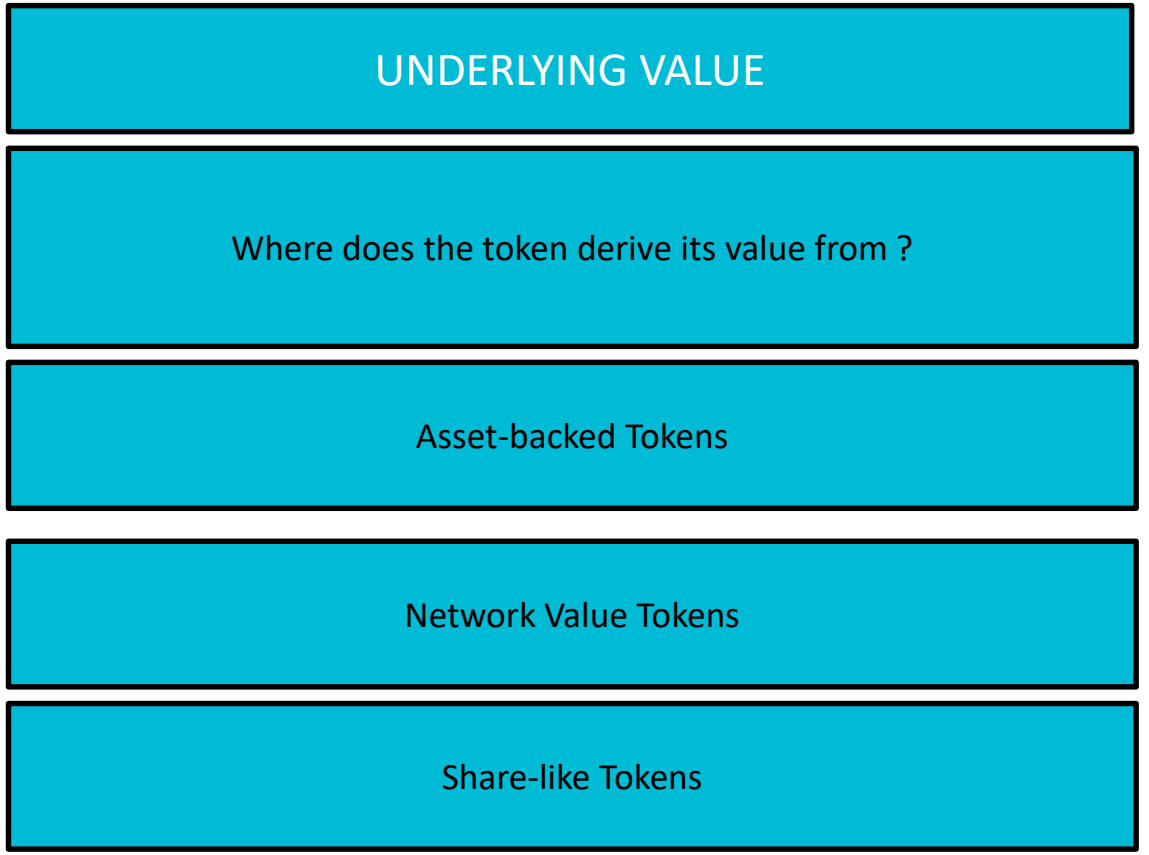
- Defines the legal validity based on jurisdictional compliance



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Underlying Value

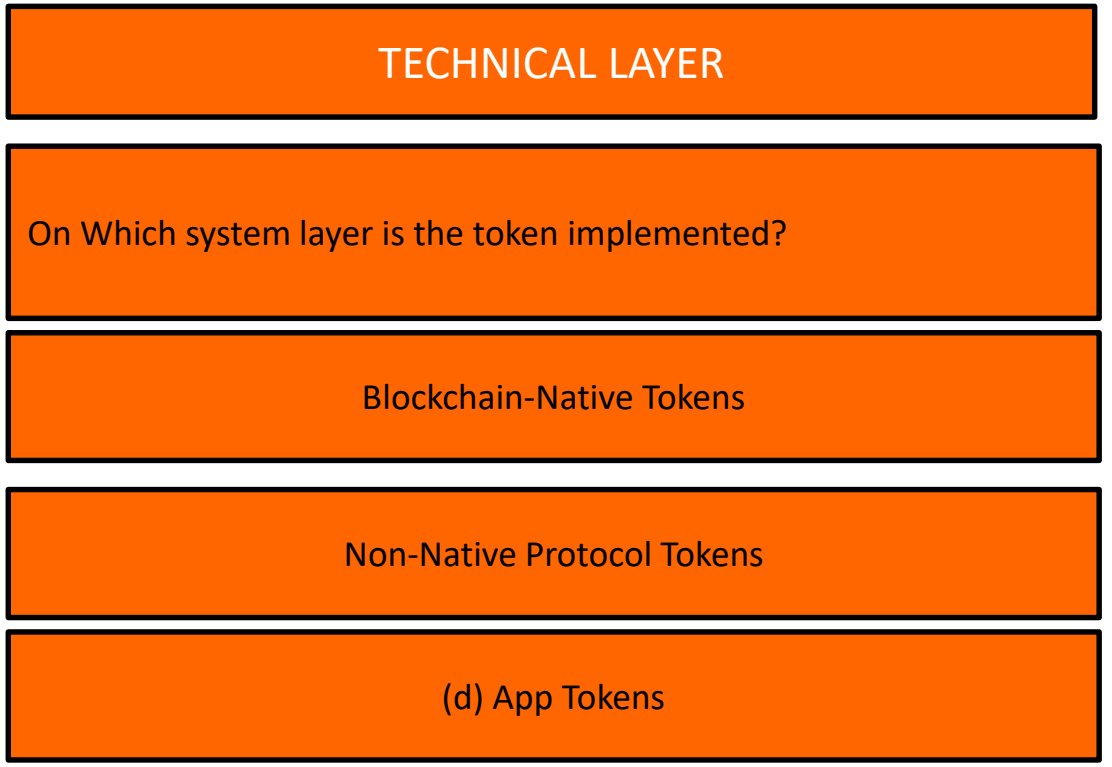
- Most tokens are created to have a monetary value. But the sources of their value differ considerably.
- Some basically work as IOUs to a real-world asset which they are tied to (asset-backed tokens)
- Tokens which are linked to the commercial success of the issuing entity (Share-like tokens)
- Tokens which are tied to the value of a network, not a central entity(network value tokens).



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Technical Layer

- Tokens can be implemented on different technical layers of blockchain based systems.
  - On the level of blockchain
    - As the chain's native token (blockchain-native tokens)
    - As part of cryptoeconomic protocol that sits on the top of blockchain (non-protocol tokens)
    - On the application level (d)App tokens



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Purpose: cryptocurrencies

- A token that is intended to be a “pure”cryptocurrency
- Example
  - BTC BITCOIN
  - ZEC Zcash
  - KIN Kin Kik

Characteristics:

- Intended as a global medium of exchange
- Functions as a store of value

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Purpose: network tokens

- A token that is primarily intended to be used within a specific system (e.g. network, application)
- Examples
  - GNO (Gnosis)
  - STX (stacks, blockstack)

Characteristics:

- Token has functionality within the issuers systems
- Not intended as a general cryptocurrency

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Purpose: Investment Tokens

- A token that is primarily intended as a way to passively invest in the issuing entity or underlying asset
- Examples
  - Neufund Equity Tokens (Neufund)
  - DGX (Digix Gold, DigixDAO)

Characteristics:

- Promises owners a share of asset value or in (future) success of the issuing entity.
- No or little significant functionality

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Underlying Value: Asset-backed tokens

- A token that functions as a claim on an underlying asset
- Examples
  - USDT (Tether USD, Tether)
  - GOLD (GOLD, GOLDMint)
  - Ripple IOUs (Ripple)

Characteristics:

- Allows trading via IOUs without actually having to move the underlying asset
- The issuer is responsible to hold the underlying asset
- Introduces counterparty risk

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Underlying Value: Network value tokens

- A token that is tied to the value and development of a network
- Examples
  - ETH (Ether, Ethereum)
  - STEEM (steem)

Characteristics:

- Tied to the value generated and exchanged on the network (e.g. transaction fee volume)
- Closely intertwined with key interactions of network participants.

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Underlying Value: Share-like Tokens

- A token with share-like properties
- Examples
  - DGD (DigixDAO)
  - LKK (Lykke)
  - Likely to be classified as a security token

Characteristics:

- The issuer promises token owners a share in the success of the issuing entity (e.g. dividends, profit-shares)
- May or may not come with voting-rights
- Mostly on no/weak legal basis

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Utility: Usage Tokens

- A token that provides access to a digital services, similar to a paid API key
- Examples
  - BTC BITCOIN
  - STX Stacks Blockstack

Characteristics:

- Grants holders access to exclusive functionality of the service

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Utility: Work Tokens

- A token that provides the right to contribute to a system
- Examples
  - REP Reputation, Augur
  - MKR Maker Maker DAO

Characteristics:

- Owning Tokens is the precondition for contribution to the system
- Contributions are either incentivized with a rewards system or holders get utility from the system/decentralized organization.

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Utility: Hybrid Tokens

- A token featuring traits of both usage and work tokens
- Example
  - ETH Ether, Ethereum, After Casper
  - DASH

Characteristics:

- Grants access to system functionalities
- Allows owners to contribute to the system.

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Technical layer: blockchain-native tokens

- It is implemented on the protocol level of a blockchain.
- Examples:
  - BTC BITCOIN
  - ETH ETHER/ETHEREUM
  - STEEM

Characteristics:

- Critical to operate the blockchain
- Integral component of the blockchain's consensus mechanism
- Part of the blockchain's incentive mechanism for block validators/other nodes

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Technical layer: non-native protocol tokens

- A token that is implemented in a cryptoeconomic protocol on top of a blockchain.
- Examples:
  - REP Decentralized Oracle Protocol
  - Augur

Characteristics:

- Integral component of the protocol's consensus mechanism
- Part of the protocol's incentive mechanism for nodes.
- Tracked on an underlying blockchain to which it is not integral e.g ERC20 Tokens on Ethereum

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Technical layer: (d) App Tokens

- A token that is implemented on the application-level on top of a blockchain and potentially protocol
- Examples:
  - WIZ (wisdom, Gnosis)
  - SAFE (Safecoin, SAFE Network)

Characteristics:

- Integrated with the application
- Part of the app's incentive mechanism for nodes and/or users
- Tracked on an underlying blockchain to which it is not integral e.g. ERC20 tokens on Ethereum

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Legal: Utility Tokens

- A token offering owners clearly defined utility within a network or decentralized application
- Example
  - GNO (Gnosis)
  - STEEM (Steem)

Characteristics:

- Closely tied to the functionality of the issuing network or application
- Internal network/app currency but not necessarily attempting to be a currency
- Grants owners the right to actively contribute to the system vs. passive investor role
- Avoids security-like features.

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Legal: Security Tokens

- A token that behaves like a security
- Example
  - SPICE (SPICE VS)
  - BITWALA tba

Characteristics:

- Showcases security-like features, e.g. voting on decisions regarding the issuing entity, dividends or profit shares.
- Holders are regarded as owners
- Little or insufficient utility.

Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Main Token Types Per Dimension

Legal: Crypotocurrencies

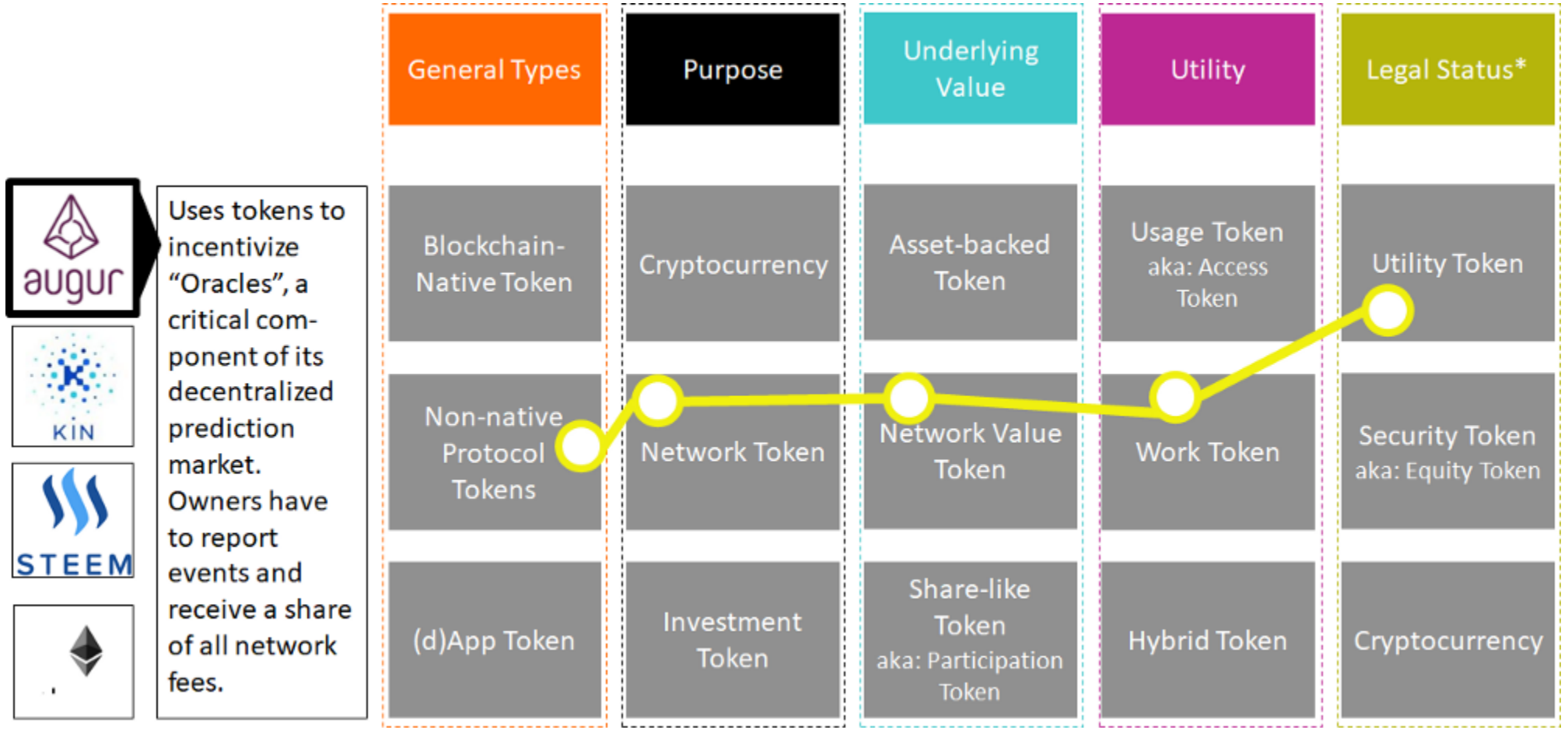
- A token that is pure cryptocurrency
- Examples
  - BTC BITCOIN
  - ZEC Zcash
  - LTC Litecoin

Characteristics:

- Acts as a store of value and medium of exchange
- Not emitted by a central authority against which owners have claims.
- In Germany (according to BaFin)
  - Currently not regarded as lawful, functional currency
  - Not regulated by e-money laws.

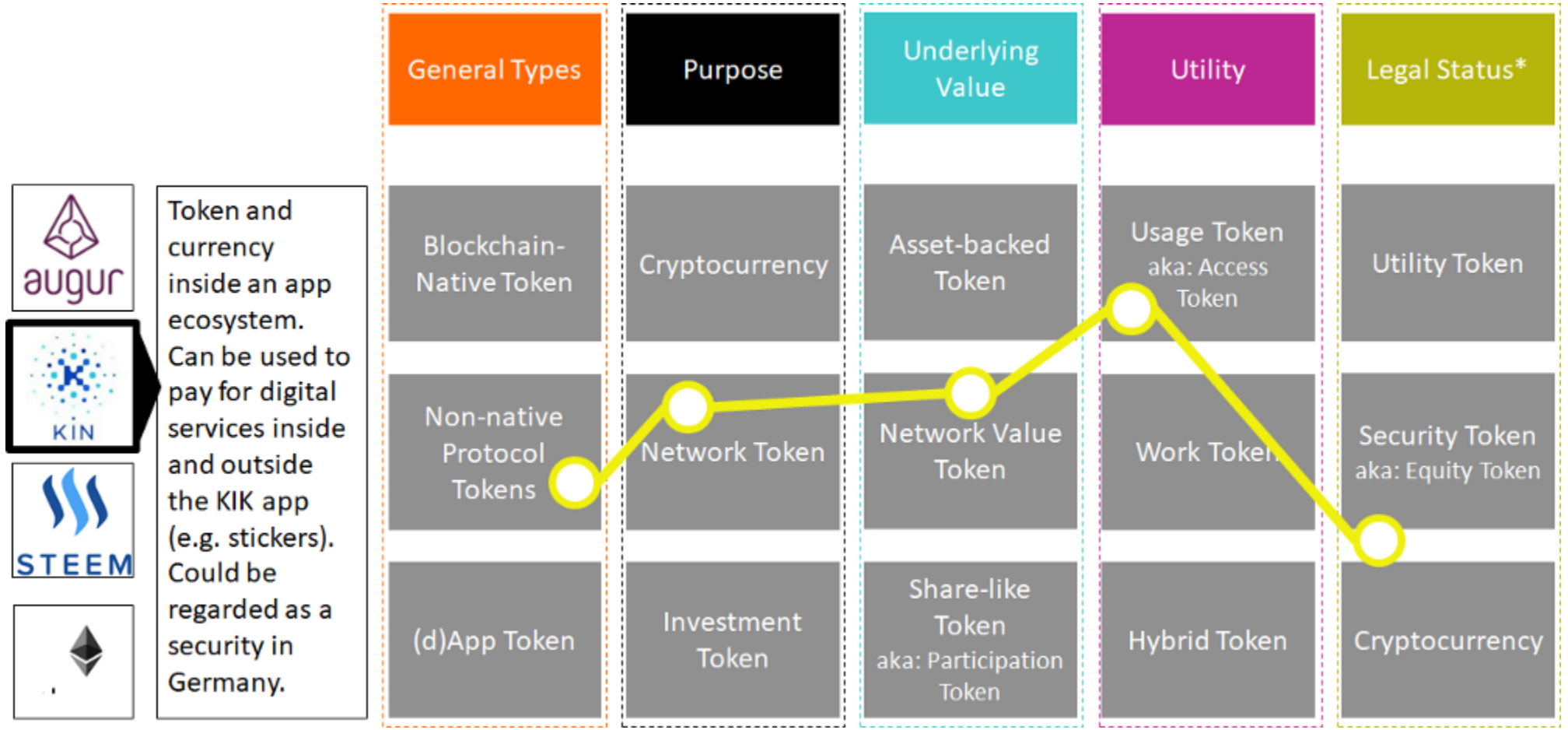
Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

TCF-TEST 1



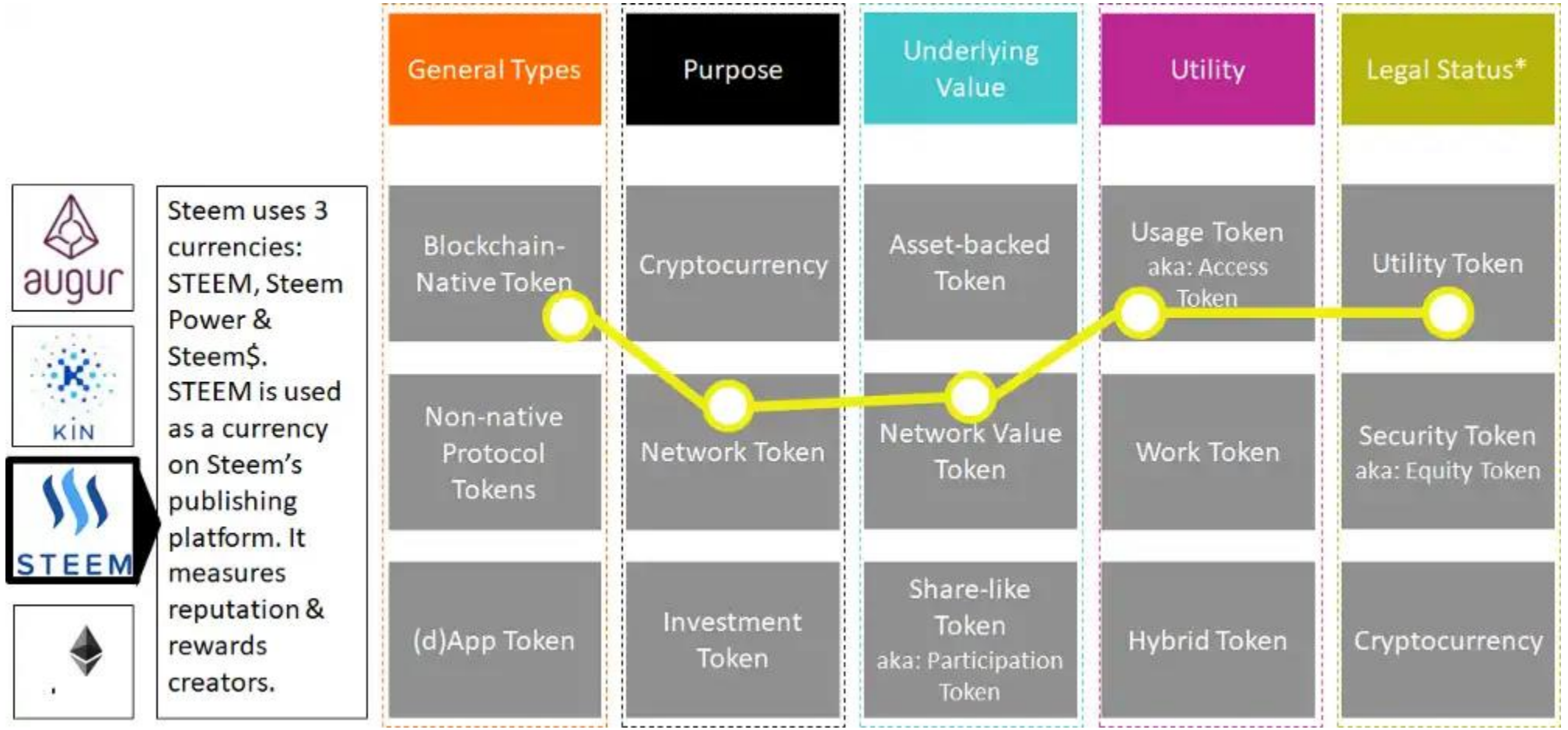
Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

TCF-TEST 2



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

TCF-TEST 3

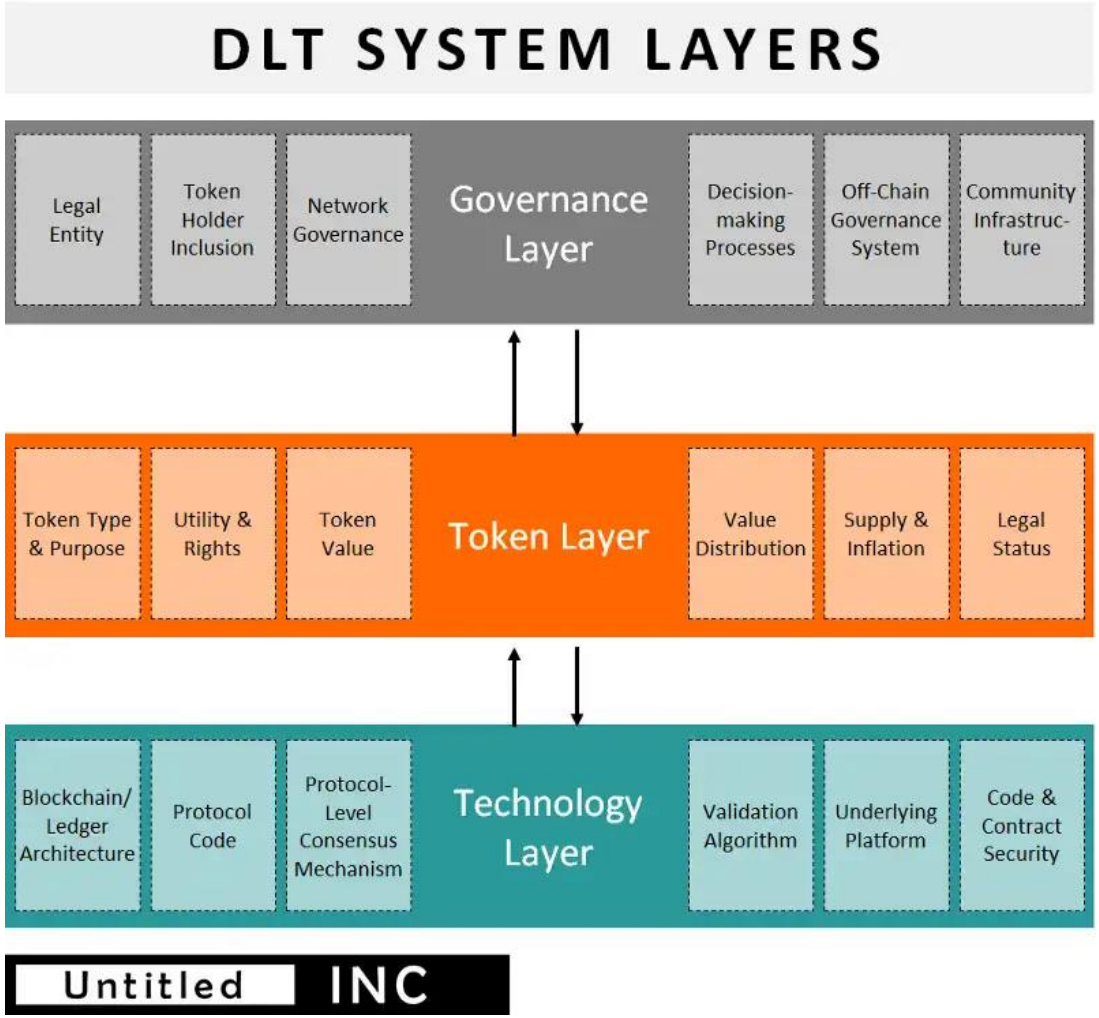


Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

Archetypes

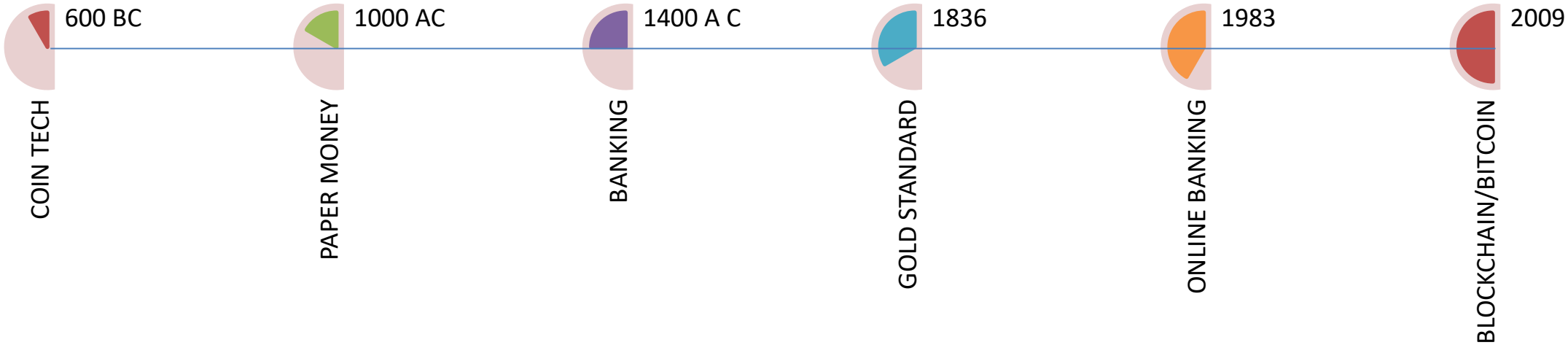
Token	Description
Cyrptocurrency	• Used as store-of-value or means-of-payment • Unit of account • Not issued by a central authority • Can be mineable or pre-mined
Tokenized asset	• Gives access to assets like gold, even in a micro transaction scale • The underlying asset needs to be held by the issuing party • Counterparty risk, contrary to cryptocurrency
Tokenized platform	• Platform-like network, not owned and operated by a single entity • Before users had limited roles in a platform, now roles are distributed and available to every network participant.. • Value (financial/utility) flows freely through the network
Token-as-a-share	• A tokenized instrument to invest in companies (through currently on no regulated basis) that has characteristics of stock and currency (ICO replacing IPO) • Shared on steroids:flexible, programmable via smart contract • Currently a highly uncertain token class a regulatory frameworks are only beginning to emerge

Distributed Ledger Technology System(DLT) Layers



Source:http://www.untitled-inc.com/the-token-classification-framework-a-multi-dimensional-tool-for-understanding-and-classifying-crypto-tokens/

HISTORY OF MONEY



SYED AWASE KHIRNI

BITCOIN

BITCOIN

- A largest cryptocurrency built on top of the blockchain technology.
- It removes the dependency on a trusted third party mediator such as banks or regulatory body to execute the transaction.
- Transactions get executed by peers based on consensus based algorithm and recorded in the blocks by the miners.
- Bitcoin is a decentralized virtual cryptocurrency built on top of blockchain.
- It enables cross-border transaction within seconds without third party intermediation.

Mining

"Mining is the process of recording the transactions as a block into the blockchain"

- Easy to acquire a bitcoin using daily use currency through bitcoin exchange.
- Alternative ways to earn bitcoin is through mining by investing computational power into it.
- Mining operations involves a very complex computational task to identify that the next block in the blockchain.
- A reward for identification of the next block in the blockchain is a bitcoin.
- Miners use their computing power to identify **a nonce (number used only once)** through which they can create **the digest/signature of the next block in the blockchain which is less than the recently added block.** The nonce is announced to the entire network and 25 BTC are awarded

SYED AWASE KHIRNI

https://lisk.io/

SECTION:

INTRODUCTION

Setting up and Installing Cycle.js

EXERCISE DEMO: 1.1

LEARNING OUTCOMES

- https://github.com/cyclejs-community/create-cycle-app

Installing Cycle Application

- Global installation of cycle.js application.

- npm install –g create-cycle-app
- Create-cycle-app tpriappone

Setting up and Installing Cycle.js

EXERCISE DEMO: 1.2

LEARNING OUTCOMES

Step 1:

Empowering You

TPRI-SYCLIQ PROGRAMS OVERVIEW

Artificial Intelligence

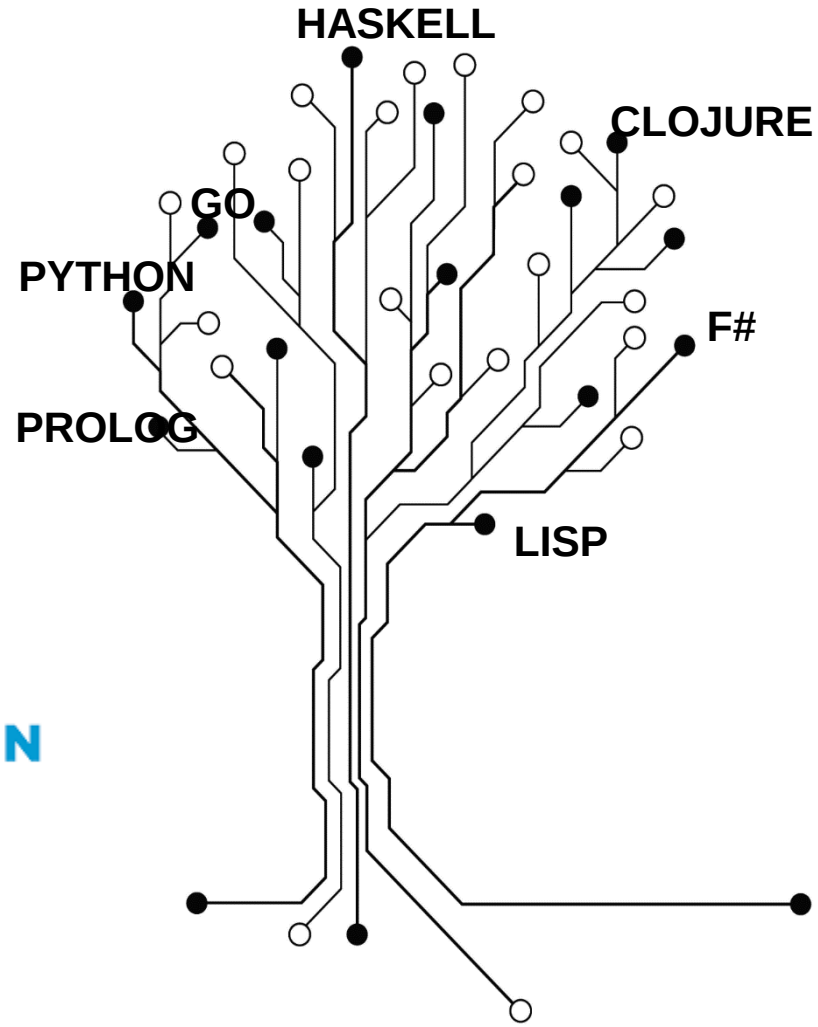
We also train on AI Stack

Reach out to us sak@sycliq.com
or sak@territorialprescience.com

www.territorialprescience.com

www.sycliq.com

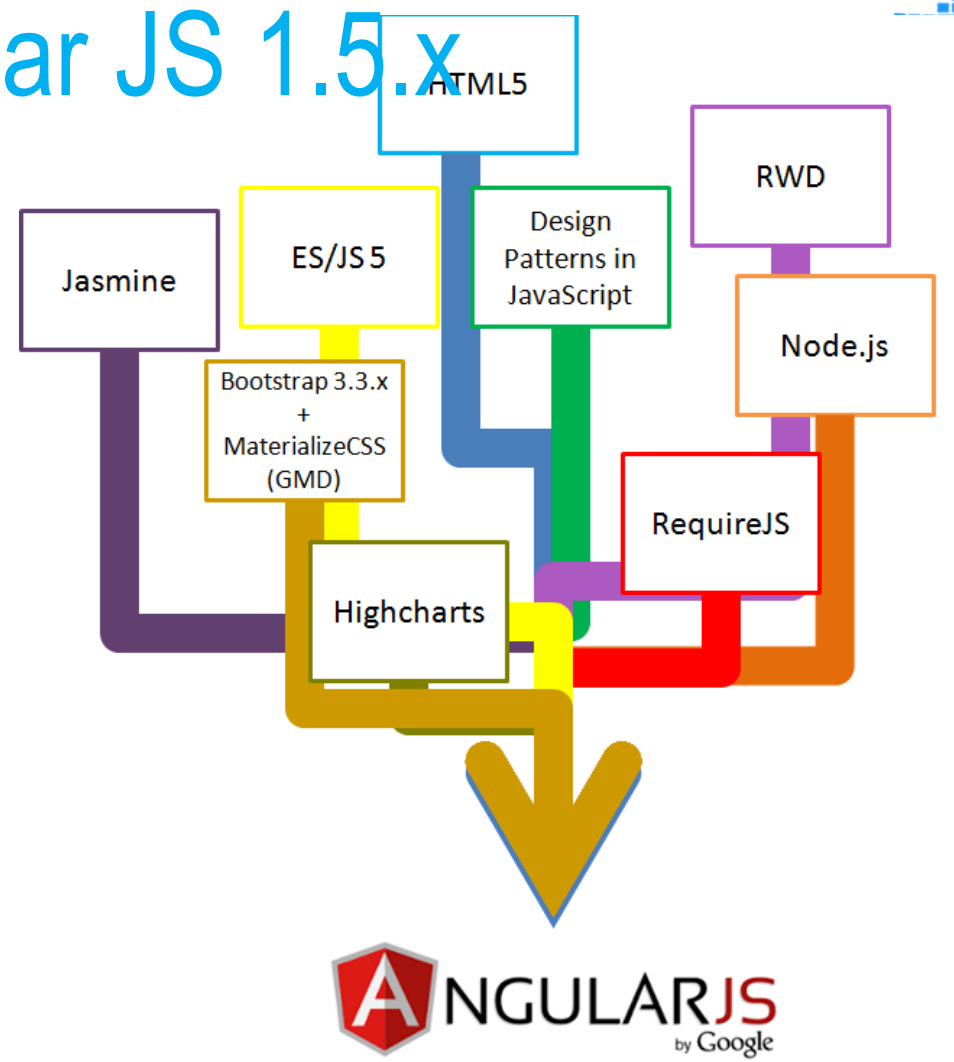
+91.9035433124



Angular 2.x/Angular JS 1.5.x

Dr. Syed Awase 2016 Session Feedbacks: http://bit.ly/2hhNg58

Reach out to sak@territorialprescience.com/+91.9035433124



Dr. Syed Awase also offers Machine learning Stack, R Statistical Stack, .NET Stack, Java Stack, RaspberryPi Stack. Get the pulse of performance from here http://bit.ly/2hhNg58

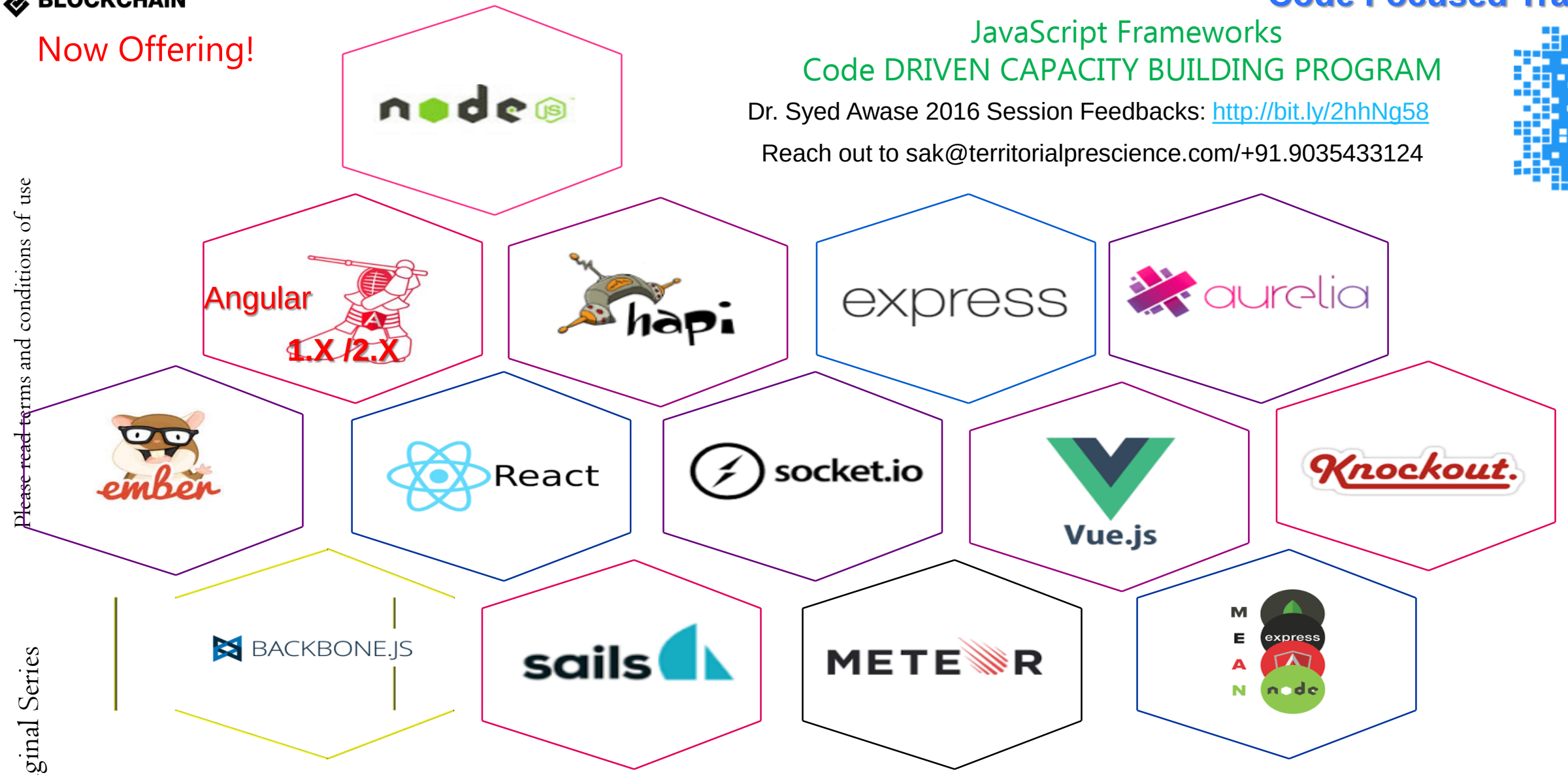
Now Offering!

Code Focused Training

JavaScript Frameworks
Code DRIVEN CAPACITY BUILDING PROGRAM

Dr. Syed Awase 2016 Session Feedbacks: http://bit.ly/2hhNg58

Reach out to sak@territorialprescience.com/+91.9035433124

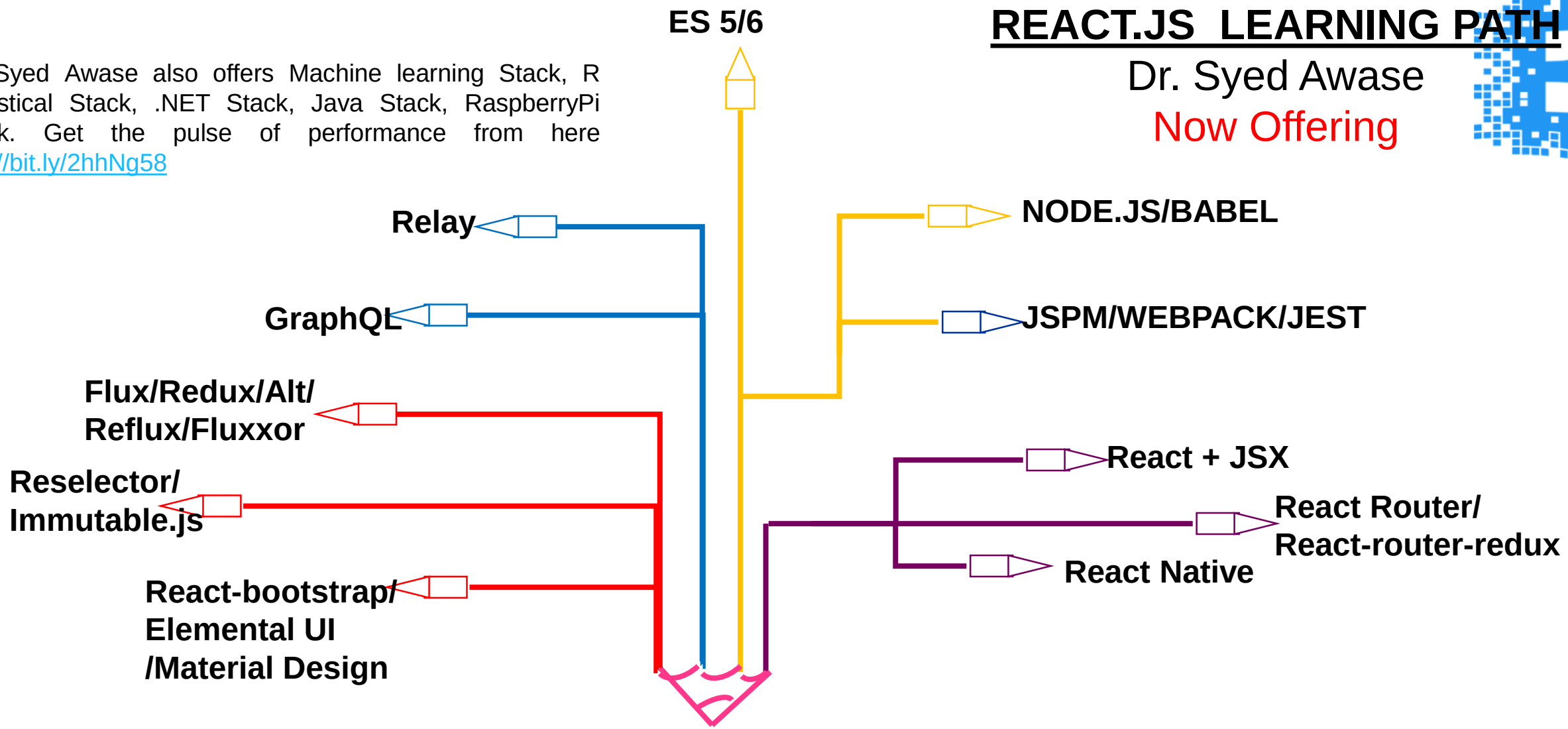


Dr. Syed Awase also offers Machine learning Stack, R Statistical Stack, .NET Stack, Java Stack, RaspberryPi Stack. Get the pulse of performance from here http://bit.ly/2hhNg58

REACT.JS LEARNING PATH

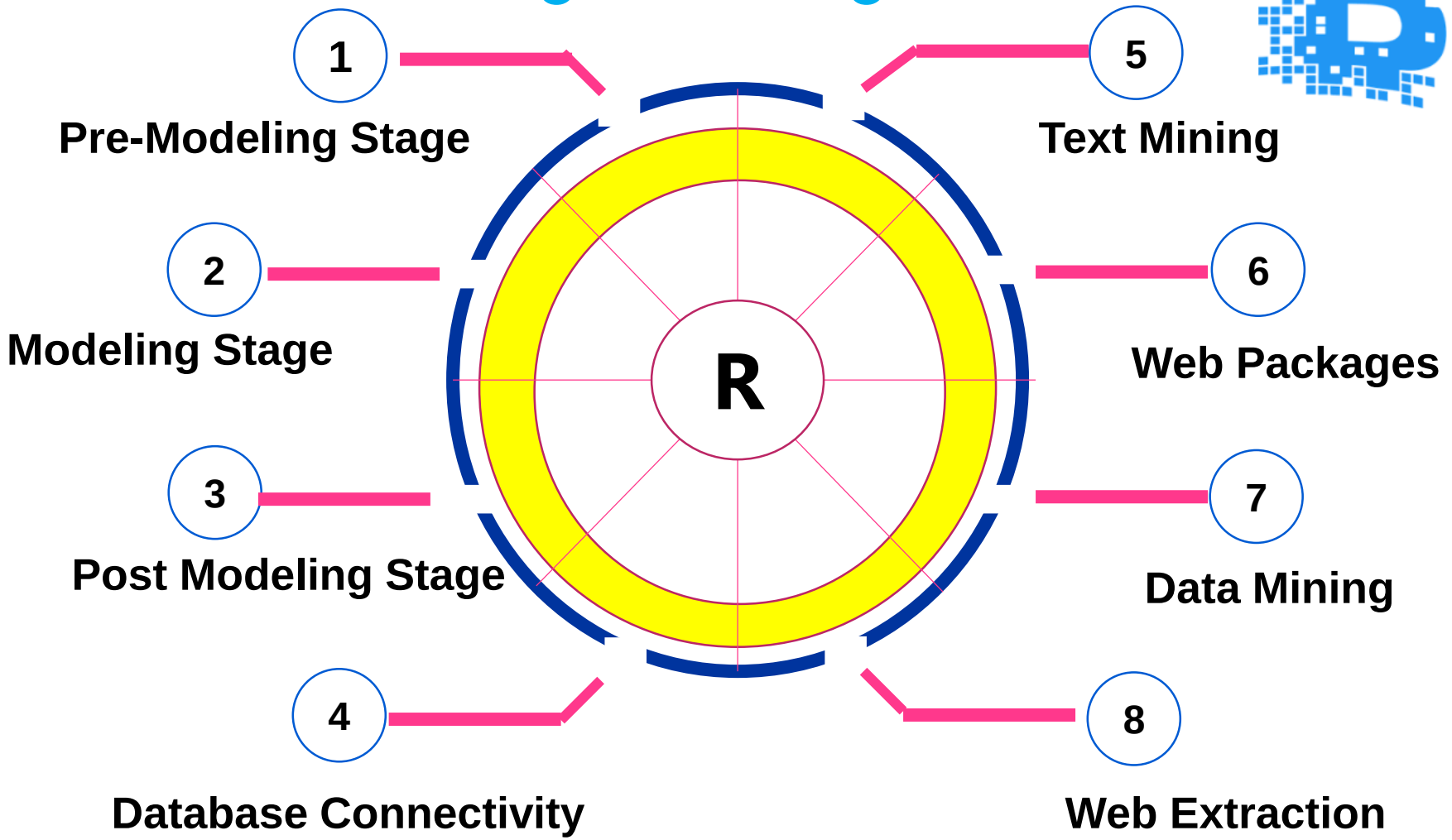
Dr. Syed Awase also offers Machine learning Stack, R Statistical Stack, .NET Stack, Java Stack, RaspberryPi Stack. Get the pulse of performance from here http://bit.ly/2hhNg58

Dr. Syed Awase
Now Offering



React.js Ecosystem

R-Statistical Programming



Dr. Syed Awase 2016 Session Feedbacks: http://bit.ly/2hhNg58

Reach out to
sak@territorialprescience.com/
+91.9035433124

Dr. Syed Awase also offers Machine learning Stack, R Statistical Stack, .NET Stack, Java Stack, RaspberryPi Stack. Get the pulse of performance from here http://bit.ly/2hhNg58

Thank You

We also provide Code Driven Open House Trainings : sak@territorialprescience.com or sak@sycliq.com

Java Technologies

- Core Java
- Hibernate
- Spring Framework
- Play Framework
- Hadoop
- Groovy & Grails

Microsoft Technologies

- C# Core
- Entity Framework
- MVC 5/6
- Web Api
- OWIN/KATANA
- WCF
- WPF

Python

- Python
- Django
- Flask
- Numpy
- Scipy
- Machine Learning

Data Science

- R Statistical Programming
- Julia

SQL and NoSQL

- Oracle
- PostgreSQL
- MSSQL
- MongoDB
- Neo4j
- Redis
- Firebase
- Apache Cassandra

Client-Side Frameworks

- Angular JS 1.5.x
- Angular 2.4.x
- React JS
- KnockOut JS
- VueJS
- Backbone JS
- EMBER JS
- Hapi JS
- METEORJS
- MEANJS
- Coffeescript
- Dart

Others

- LISP
- CLOJURE
- RUST
- GO
- RaspberryPI
- Coming Soon
  - PHP
  - Robotic OS