XIDEN

WHITEPAPER

CryptoDATA

VERSION 1.1

Table of Contents

1. Abstract

The XXIst century represents an era guided by highly rapid technological advancement. As a result, the global human connection and the digitization of markets have become fundamental for most human-performed activities, most of which require a connection to the Internet.

The Internet is a global network that connects users worldwide via smart devices, regardless of their geographical location, in order to facilitate the exchange of information. In an industry currently controlled by monopolies, we emphasize the free exchange of information between users and we aim to give any desiring human the opportunity to transmit any type of information while controlling both their own connection points and the integration process into the Global Internet Network.

Our solution gives people *free internet access and connectivity*. Furthermore, it empowers them to control the process through their smart devices, as mentioned, regardless of their location.

Smart devices are evolving rapidly as they have to meet the ever-changing needs of the users. Moreover, these smart devices possess resources that can be utilized and integrated into the Internet of Things concept.

A global network of smart devices can provide computing power and storage capacity in a decentralized manner that will provide environmentally-friendly benefits. The users will no longer have to discard their gadgets once their specifications become obsolete and can extend the devices' lives by continuing to make the most of their resources.

2. Introduction

As the world tends to become more and more decentralized, many projects are exploring different ways to exploit the blockchain ecosystem and mitigate its limitations. Alternative platforms, technologies, and services are moving from proprietary systems to decentralized, open ones.

Our contribution is a new decentralized system that aims to provide a wide-ranging concept that will simplify the process of connecting to the Global Internet Network, data transfer validations, and integration of devices into the blockchain.

3. XIDEN Blockchain

The *Xiden Blockchain* is a decentralized network that utilizes blockchain technology to ensure conformity to facts, the security of data, and operational procedures. This network is built to support the integration of all smart devices into the blockchain system in order to ensure optimal and efficient use of all available resources.

The Xiden network combines technologies such as the *Internet of Things* with blockchain to develop a protocol that will allow smart devices to perform tasks automatically and autonomously, thus ensuring high-speed data and operational procedures validation.

The Xiden network aims to become an open-source system that will provide users with the opportunity to have a free and permanent internet connection regardless of their location or device. Furthermore, by decentralizing the Internet, the user is no longer conditioned by affiliation or association with an internet service provider, no longer has limited access to certain parts of the Internet, and their identity will be permanently protected and secured.

An important aspect of the Xiden network's user identity protection is represented by its high cyber security features as a decentralized network with automatic and autonomous validations, ensuring protection against third-party manipulation.

3.1 Necessity

What is Xiden Blockchain (XDEN) and why do we need it?

The Xiden Blockchain aims to deliver a new internet concept consisting of multiple decentralized networks which will act as a single network connection of nodes governed by individual users.

XDEN's objective is to connect all the system's existing devices in order to facilitate the process of data transfer validation between users with no additional energy consumption, and no customized or special equipment required.

The system's integrated devices will fulfill multiple functions by successfully meeting the requirements of the users and the blockchain technology while using the same volume of resources.

3.2 Function

Multiple functions of the Xiden Blockchain:

I. Development of DApps
   A. Governance, access, and use through smart contracts
   B. Utilization of the undirected resources of the devices integrated into the system in order to ensure:
      1. Computing power
      2. Storage
      3. Security routes

II. Secure bridges with other blockchains
   A. Easy integration that facilitates user access to the Xiden blockchain

III. High Transaction Speed
   A. Low Validation Time
   B. Low Gas Fees

4. Overview

4.1 The need to develop a proprietary blockchain

Blockchain is a relatively new technology with various applicabilities, but it has not yet reached its intended maturity. Thus, as it develops and integrates into actual activities in our society, problems regarding limitations in satisfying user needs are identified more frequently. The issues encountered by blockchain users are becoming more and more pressing as they directly affect both developers and beneficiaries.

A blockchain system is comprised of three layers:

- *Application* - ensures responsibility for updates and interoperability between multiple systems for recording actions (*transactions*).
- *Networking* - responsibility for operating data and ensuring its propagation.
- *Consensus* - ensures the system's responsibility to validate data and maintain status updates.

CryptoDATA's extended experience in blockchain development allowed us to identify issues in current blockchains' utilization. By means of analyzing the evolution and the necessity of using current blockchains for meeting the needs of beneficiaries in order to achieve sustainable development, we have identified various issues that will be presented in detail.

4.2 Identified general issues

4.2.1 Interoperability

As aforementioned, due to the fact that blockchain technology is still in its incipient form, developed by programmers according to their initial applicability needs and requirements, it does not yet have the capability to meet the needs and requirements in all concerned areas due to different protocols, coding languages, consensus mechanisms, and privacy measures. This is not a bad thing, and it doesn't mean we have to give up the benefits brought by the use of blockchain technology. It means we need to implement an optimal solution that will interconnect all existing and future blockchains. Thus, we do not need to develop a new blockchain that will perform

the functions of two or more blockchains. We just need to interconnect them in order to directly and quickly benefit from the advantages of each one as needed.

The problem is that with so many different networks, the blockchain space is in a "*state of disarray*" due to a lack of universal standards that would allow different networks to communicate with each other.

The lack of such uniformity across blockchain protocols also takes away the consistency from basic processes like security, making mass adoption an almost impossible task.

The establishment of industry-wide standards with regard to various blockchain protocols could help enterprises collaborate on application development, validate proofs of concept, and share blockchain solutions, as well as make it easier to integrate with existing systems.

4.2.2 Update and adaptability

One of the most important identified issues is the lack of rapid adaptability of technology to meet the needs of people in carrying out specific tasks. As mentioned, the state of the art of blockchain technology is still in its infancy, and there is no perfect code to meet all needs.

The main challenge is to build a technology that combines existing blockchains with future ones. Thus, there is no need to rebuild ever-performing blockchains with ever-increasing costs of resources. The goal is to develop new features that will immediately meet the needs and can be easily integrated both with existing blockchains and with legacy systems.

For example, organizations that try to integrate blockchain in their legacy systems are required to completely restructure their previous system or design a way to successfully integrate the two technologies.

One problem is that due to the lack of skilled developers, organizations do not have access to the necessary pool of blockchain talent to engage in this process and high development costs. Reliance on an external party can soften this problem, but most solutions present on the market require the organization to invest a significant amount of time and resources in completing the transition.

4.2.3 Scalability

One major technology challenge of blockchain is related to the technical scalability of the network, which can put a strain on the adoption process, especially for public blockchains.

Once blockchain technology began to be used and integrated into various circumstances, data validation issues began to arise. Because Blockchain technology is by default characterized by decentralization, it means that validation operations must be performed by following a certain order and more validators must be added. To do this, multiple rewards and benefits must be offered to the community.

For example, the two largest blockchain networks, Bitcoin and Ethereum however, are far behind when it comes to transaction speeds. While the Bitcoin blockchain can process three to seven transactions per second, Ethereum can handle approximately 20 transactions in a second.

Analyzing the primitive validation technology based on the concept of proof of work, we found that the utilization rate of blockchain technology is higher than computational power and thus generates too much deficit between investment, administration costs, and the received rewards. As a result, the trust and sustainability power of network users declined, the network was unable to solve mathematical operations, and the cost per transaction became huge, turning a targeted network into an impossible-to-use network to improve the everyday life of the increasing number of users.

When the user number increases on the network, the transactions take longer to process. As a result, the transaction cost is higher than usual, and this also restricts more users on the network.

The purpose of the blockchain architecture is to develop a system of decentralized networks in order to ensure a balanced distribution of tasks. A solution and a big challenge are to combine the connections between blockchain networks with different applicability so that the validations can be operated in a timely, efficient, and correct manner in networks with proprietary rules.

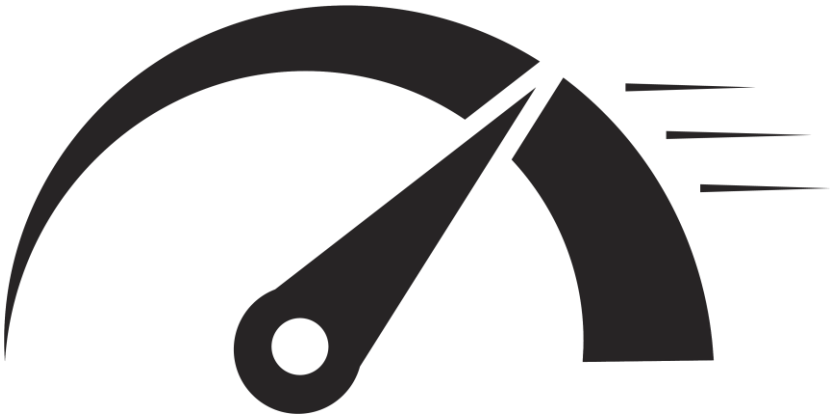
4.2.4 Validation and security

The security of a blockchain network is given by the decentralized network of validators whether it uses *POW* or *POS* for them. In the BTC blockchain, the security and conformity of the network are maintained and ensured by the mining devices. As we presented earlier, these devices depend on the profitability and energy consumption factor, so there exists the possibility for the network to be taken over at the right time by an entity that has unknown intentions. This situation has already happened in the case of BTC. Thus, it is a problem identified in the analysis of the evolution of blockchain technology.

4.2.5 Validation speed

Due to the fact that blockchain technology is not yet mature, it could not be tested under real stress in order to extract valuable insights on limitations. Most blockchains have tried to improve the validation process only by observing the behavior of highly developed networks such as *BTC* or *ETH*.

The solution to this problem is to develop a slightly upgradable, modular blockchain that uses multiple validation architectures. Ideally, this blockchain can easily interconnect with blockchains of different architectures but can also support the development and integration of applications in as many easily accessible and usable programming languages as possible.



5. Terms

Xiden Internet Network: We have built a private decentralized internet network to which all users can connect via any smart electronic device.

REALM Node: A validation node that registers the blocks into the blockchain network. As geographical locations, the *Realm nodes* are distributed around the globe in order to facilitate the registering process of the Guardian node, which will validate data according to user preferences and/or requirements.

GUARDIAN Node: A light validation node that verifies the conformity of both data and the system's integrated devices, contributing to the network's operation and providing resources for decentralized computing and storage power. It uses *Delegated Staking* and it allows miners to band together with their computing power in a single Krater Pool.

Miner: Any smart device that makes its resources available for the optimal operation of the network and validates the conformity of the other devices integrated into the system.

Devices: Any smart device that is not fulfilling the miner function and it does not have the minimum resources to meet the requirements of being a blockchain validator. A smart device that makes its resources available for the development and use of DApps.

SPECTRALIS NETWORK: A WiFi point unlocked by a Router that can open a pool through which smart devices can connect freely in order to access an anonymous internet network.

BDX: A dynamic list of Guardian Nodes and Miners stored in all the system's integrated devices, used to validate the existence and conformity of the respective devices.

Krater: The pool opened with the help of the node device and to which the user can connect other smart devices in order to validate transactions, but also boost the node's computing and storage power.

Age: Represents the current phase of the blockchain in terms of operation. Each phase brings different operating procedures, standards, and rules in order to ensure sustainable and continuous development. A predetermined finite number of blocks represent each blockchain Age. XDEN's Age details are available in the *Genesis Distribution* chapter.

DApps: Decentralized applications are digital apps or programs that exist and run on a blockchain or peer-to-peer (P2P) network of devices instead of a single device.

Epoch - Represents the blockchain period given by the number of blocks. At the beginning of each blockchain epoch, a census of the validators is made by checking their status.

Proof of Stake: A blockchain consensus algorithm that ensures the conformity of every device that aims to become a validator in the network by submitting a quantity of XDEN. The PoS is an algorithm that helps validate and maintain network data integrity.

Proof of Connectivity: A blockchain consensus algorithm used to ensure that all devices integrated into the system are genuine, connected and function within the system.

Proof of Existence: A consensus algorithm that ensures a specific data or digital transaction is associated with a timestamp and a signature, thus proving that the respective data was created on the mentioned date and time.

Smart distributed resources: The concept defines the smart use of the available resources on owned devices. These resources can train AI modules, generate computing power to solve mathematical equations or increase storage capacity.

Block Generation Time: Each block is generated and recorded at every two (2) seconds and will contain the exact amount of information and transactions that can be transmitted in precisely that time. The blocks will be validated and registered by the Realm Nodes, and the integrated system's devices conformity will be verified by the Guardian Nodes.

Reward: Financial benefit for users and integrated devices (miners) that contribute to the operation of the network.

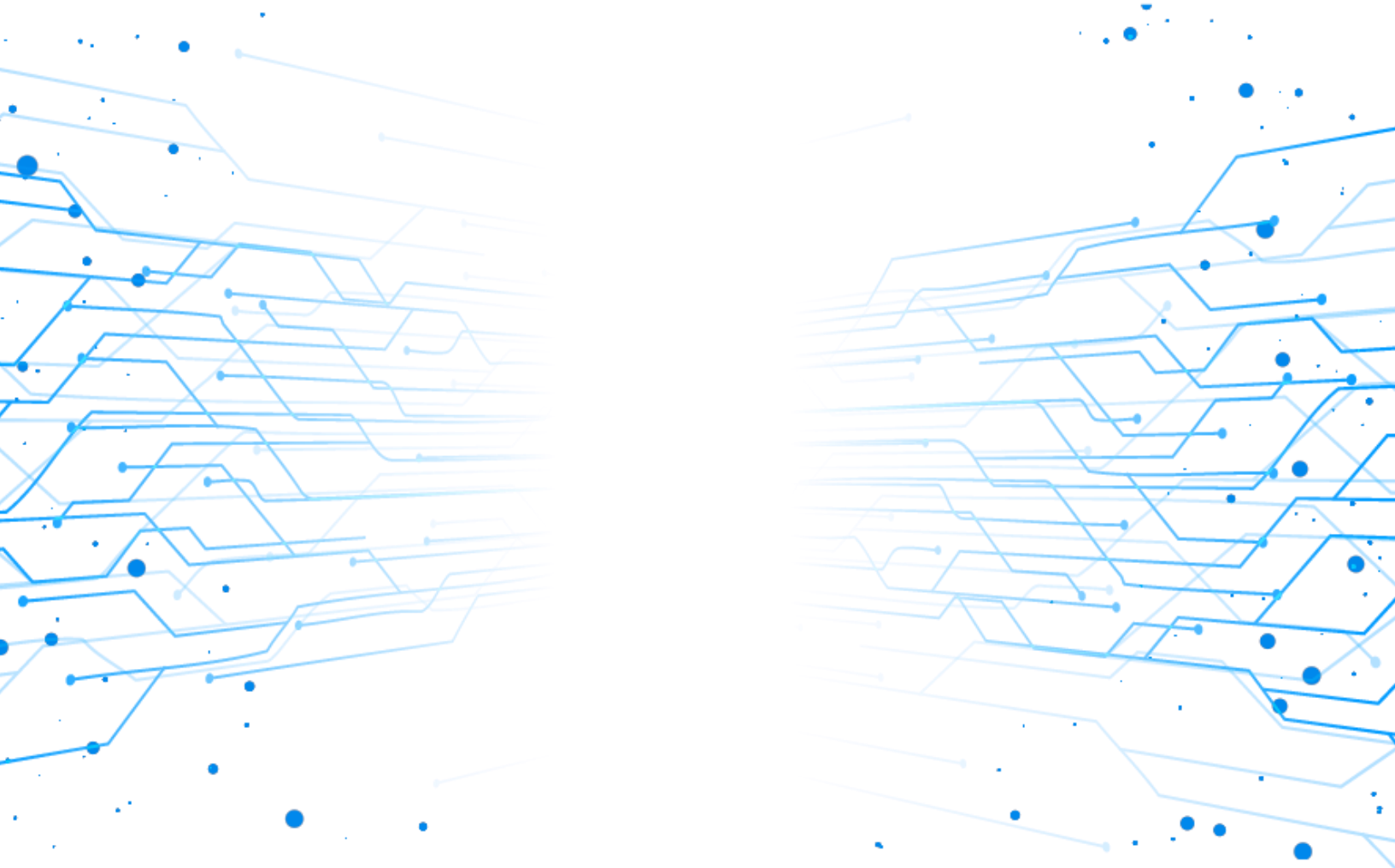
GAS Fee: The cost of validating a transaction. The transaction is considered an exchange of information between two wallet addresses that can be validated and registered in the blockchain system.

XDEN: A native transferable digital asset.

Wallet: A digital wallet is a software-based system that securely stores users' payment information and passwords for numerous payment methods and websites

Matrix ID: An independent mode that generates an anonymous digital identity for every device integrated into the system.

VOBP: Voice Over Blockchain Protocol is an encryption protocol that secures and encrypts information exchange channels thanks to its integrated algorithms such as AES-256, Extended Triple Diffie-Hellman, Double Ratchet, and asymmetric key generation.



6. System Overview

- The Xiden Blockchain is a new layer for the Internet that operates on a decentralized network of connectivity points around the world.

- The Xiden network is built to facilitate the integration of all smart devices in order to provide a free connection and internet access in a secure, anonymous, and unrestricted manner.

- The Xiden network integrates the IoT concept to converge all the resources within smart devices (*processing power and storage capacity*) into a decentralized architecture.

- The available resources of the devices integrated into the network serve as infrastructure for both the application developers and application users. The aim is to develop a common resource infrastructure to streamline costs, consumption, and maintenance.

- The Realm MetaNodes represent the 30 validation primary nodes that verify and record the information exchange between blockchain entities and will receive rewards through the gas fees for transactions.

- In order to process validations, a Realm Metanode must have a minimum staking deposit of 2.000.000 XDEN. A higher XDEN staked deposit will bring more validations and more rewards.

- The Guardian Nodes are smart devices with computing power and storage capacity that can distribute the network's resources like any electronic device.

- The Guardian Nodes can open Krater Pools and facilitate the connection for both smart devices and miners.

- The Guardian Nodes perform the same functions as the miners by verifying the conformity of the devices integrated into the system at the end of each epoch.

- The Guardian Nodes fulfill the Hotspot function by opening the Spectralis Private Network's access points and facilitating the access of smart devices into the Xiden network.

- The Krater Pool is a cluster in which miners and devices can be integrated and which centralizes the validation power. It behaves like a total validation power and validates the integrity of the devices integrated into the system.

- In order to perform validation, a Krater Pool must have 1000 XDEN in the staking wallet.

- Miners are devices that can perform validations and can be registered in the Krater Pool.

- Miners are smart electronic devices that have resources (storage capacity and computing power) and make those resources available to the Xiden network.

- Miners are rewarded with XDEN based on the provided resources, completed validations, and uptime.

- DApps can be configured by developers to work on the basis of XDEN or other tokens developed on the Xiden Blockchain smart contracts. The XDEN obtained from users can be distributed to the DApps' developer or to the devices that contributed with resources, depending on the rules set by developers.

- The more transactions and exchanges between information entities, the more gas fees for the Realm MetaNodes that will be distributed to Krater Pool owners.

7. XIDEN Layers

7.1 Decentralized Internet Private Networks

The Xiden project aims to build a new layer of decentralized internet to ensure that it cannot be controlled or manipulated by organizations or third-party entities. The main purpose of this Internet Layer is to eliminate the possibility of censorship and to allow everyone access to the worldwide web.

The Xiden network is built on integrated devices such as routers that will be owned by users and will be located in multiple locations around the world. The routers can open Hotspots called SPNs (Spectralis Private Networks) to which smart electronic devices can automatically connect in order to access the Global Internet.

SPN Hotspot owners will receive XDEN tokens depending on the number of devices that connect to their network.

The router is specially built to offer high-security features and it cannot be compromised. The security must be invulnerable as it hosts a public SPN Hotspot and facilitates a connection point for anyone in the system. The router uses the VOBP architecture and cryptographic algorithms in this protocol to secure connections between devices and the router.

The implemented protocol allows the opening of a connection only after an exchange of P2P encryption keys has been successfully completed between switches, has been successfully validated by the Proof of Existence consensus, and in an automatic manner, without human intervention.

A key attribute of the XIDEN network is the provision of privacy for all network users. The Hotspot is a public network owned by users and, therefore, an architecture and algorithm have been implemented in order to protect the digital identity of both the connection provider and the users who are part of the connection.

The Spectralis algorithm transforms each router, miner, and device into a distribution node for data traffic, and at the same time, each device becomes a client that transmits and initiates the exchange of data, but also a proxy that relays data to other users.

The management of this network is executed automatically by a Layer of the XIDEN blockchain. This means that there is no human intervention that could intervene in the manipulation or tracking of data, metadata, traffic, and identities.

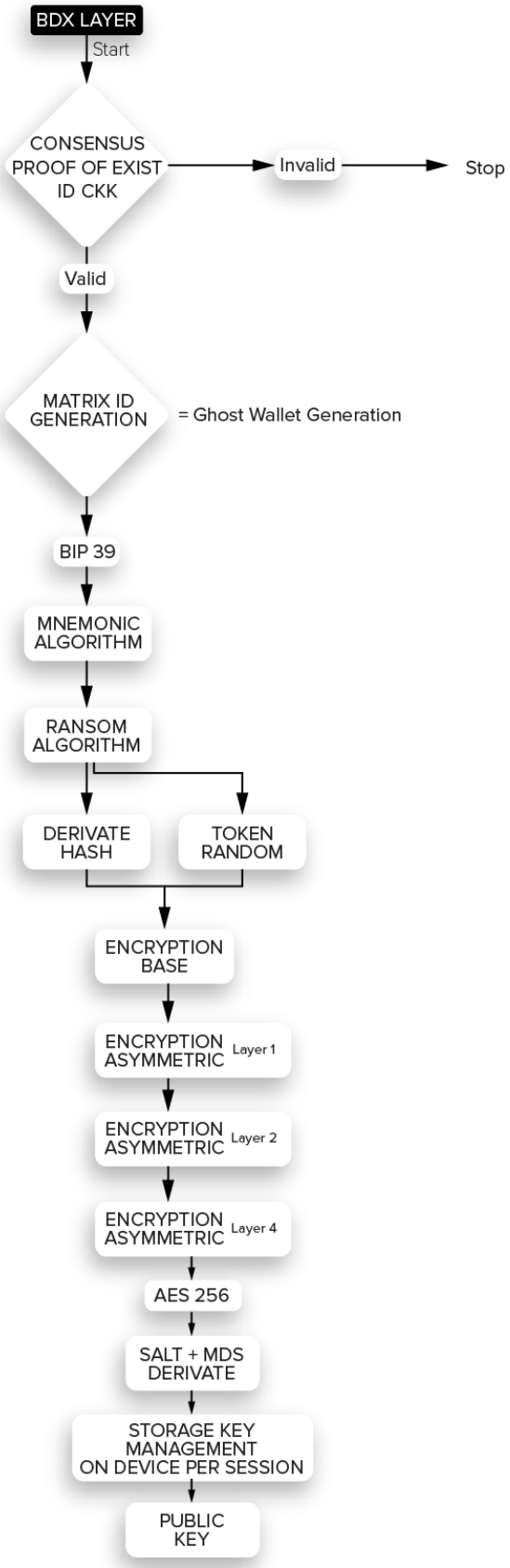
Each Node generates a blockchain identity each time it initiates a session. This identity is encrypted and used instead of the real device's identity in order to never clearly share the device's actual identity data. The Matrix ID generates data encryption keys valid for the open session. Thus, each Node will use aliases as temporary identities to connect and open routes with other nodes.

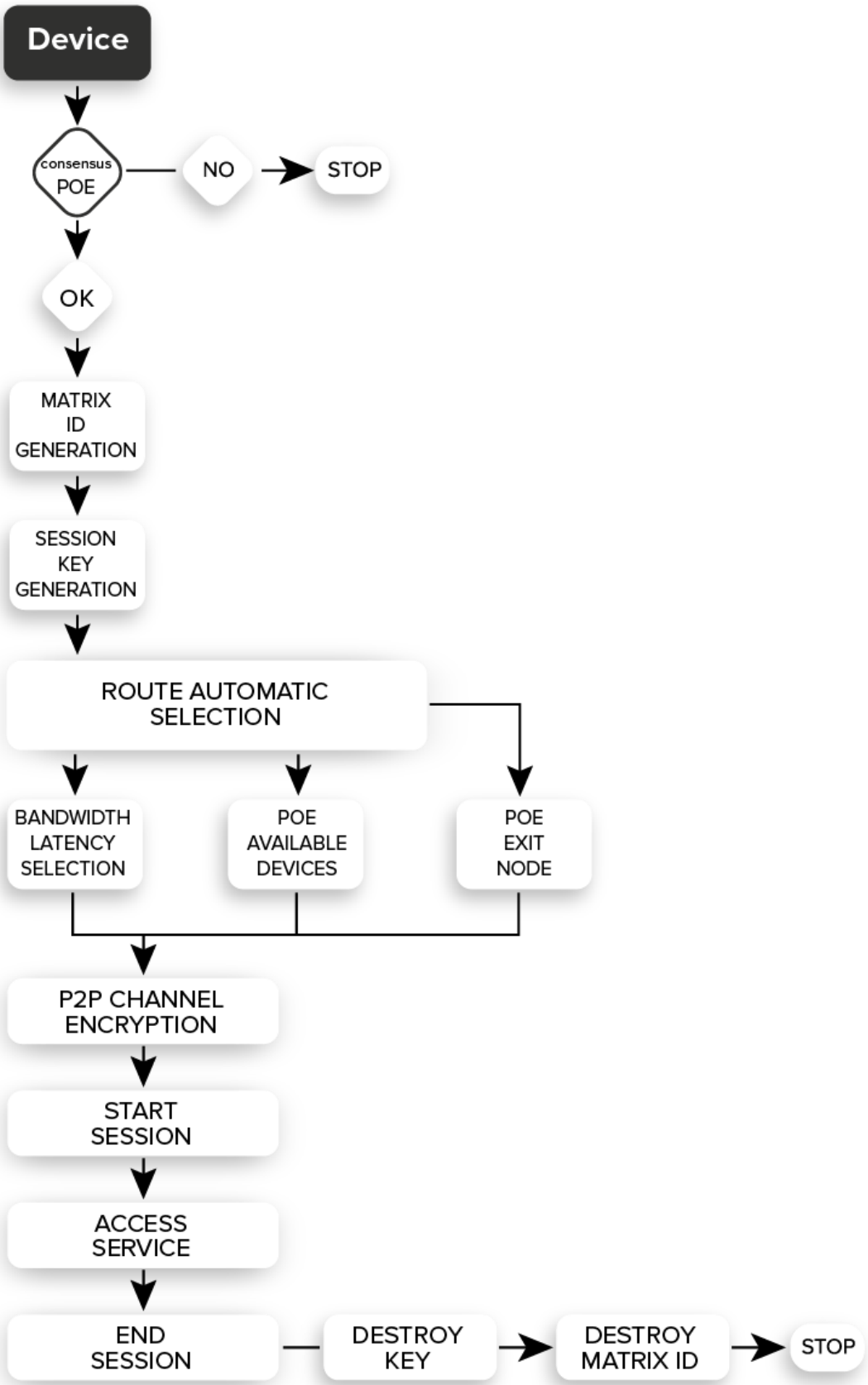
The Spectralis Private Network combines elements of TOR and VPN architectures to anonymize and protect the source, traffic, and the entities that are used as relay Nodes. The Spectralis protocol allows the user to select their exit point as the location. It cannot select a specific output node, it can only select the geolocation where it needs to exist with the new identity.

Thus, any entity or organization that tries to block traffic or certain sources will no longer be successful as the output nodes will change periodically.

The automatic assignment of the routes is executed according to the internet bandwidth necessary for communication and also according to the propagation distance. Thus, each initialized session should always have high speed and reduced latency to not impede the process of communication and transmission of information.

Matrix ID Session Generation




Device
consensus POE
NO
STOP
OK
MATRIX ID GENERATION
SESSION KEY GENERATION
ROUTE AUTOMATIC SELECTION
BANDWIDTH LATENCY SELECTION
POE AVAILABLE DEVICES
POE EXIT NODE
P2P CHANNEL ENCRYPTION
START SESSION
ACCESS SERVICE
END SESSION
DESTROY KEY
DESTROY MATRIX ID
STOP

7.2 Smart Distributed Resources (SDR) layer

Another layer that defines the XIDEN Blockchain is called Smart Distributed Resources (SDR). The SDR layer forms a network with a decentralized architecture and provides computing power and storage for all participants with smart electronic devices registered in the XIDEN network.

SDR is an eco-friendly concept as its implementation aims to bring together all devices that have or have had a utility for the owner user. In this system, the resource management is done automatically by the SDR layer in order to not affect the activity of the device and facilitate specific access to the respective device by the other users of the network.

Many devices such as smartphones, computers, or servers have become obsolete with the advancement of technology and operational requirements, and recycling them represents a difficult option. Thus, the SDR layer represents a solution for optimizing their utility by using their resources within the network and it will reward the owners with XDEN for making their resources available for the other users in the network.

Current up-to-date smart electronic devices are not being utilized at full capacity at all times, even though they maintain the same power consumption. Through the SDR layer, the owners will be incentivized for making their resources available to the network and this process will not affect their performance for regular activities in any way.

This concept empowers users to utilize their smart devices' power in different areas. It represents a good thing as these devices use only a minimum level of their energy output to function for their intended purpose, and do not need special reconfiguration to be used on this blockchain.

The most important part of the SDR layer is represented by the security provided for the devices integrated into the system. To connect to the XIDEN network, users must install the PAIR application for the XIDEN network. This application has Worm Guard installed by default, a system that protects devices from any unauthorized attack.

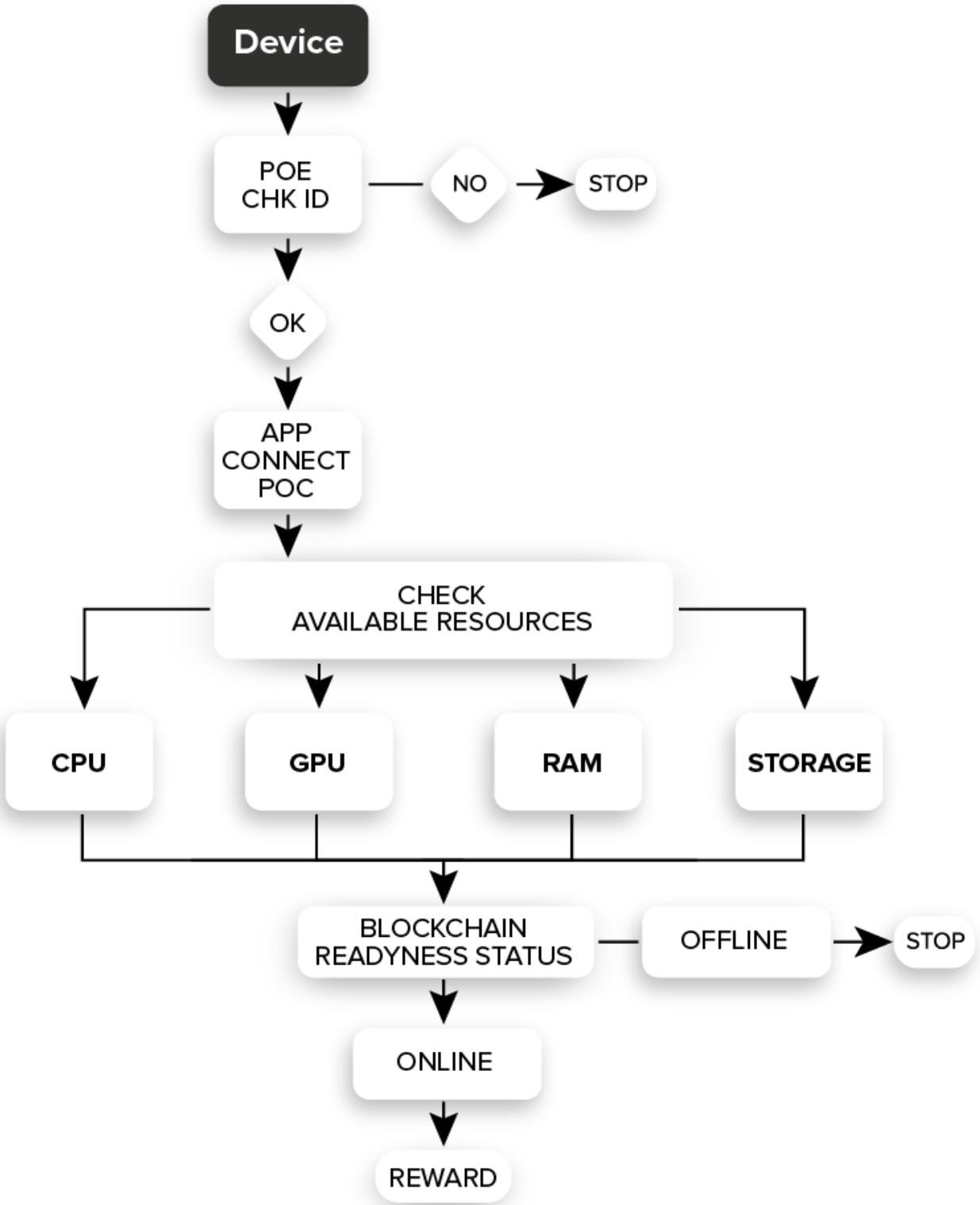
The application is an interface that only distributes computing or storage jobs to registered devices and does not allow system access or access to any other functions.

The compound of Proof of Staking (POS), Proof of Connectivity (POC), and Proof of Existence (POE) protect users from any attack. When an irregularity is identified by the three consensus layers or by the Worm Guard system, an ODI attempt or attack will be reported; The initiator of the attack will be isolated and punished by receiving burn for his amount of XDEN available in his staking supply. The three consensus algorithms work actively to maintain the integrity and security of devices and the functionality of the system.

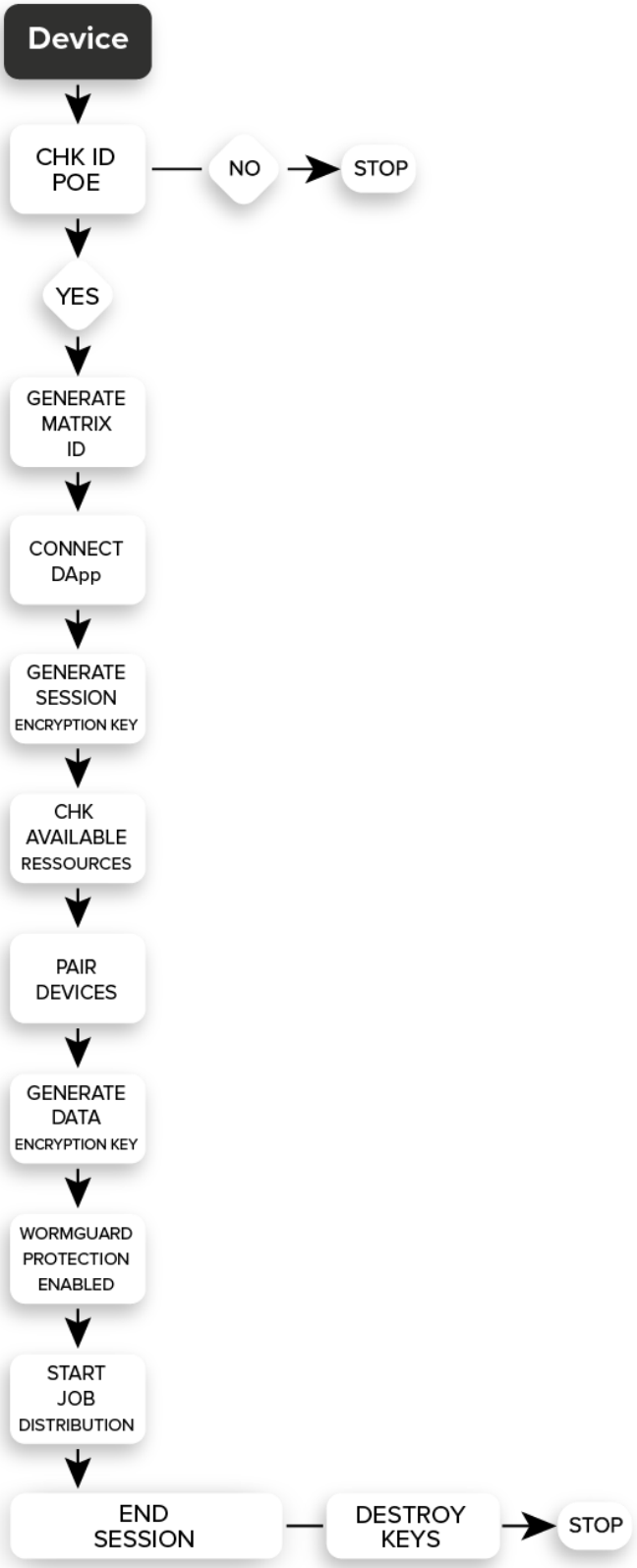
Communication between the devices is P2P and is subject to the consensus rules. Each device uses a generated Matrix ID and does not use its actual identity to communicate. Data transfer uses the VOBP protocol when accessing the device's resources, to encrypt and protect jobs. Being a decentralized network, privacy is extremely important and any task distributed to the network must be visible only for the user that initiated it.

Session encryption keys will be generated for securing the channel and message encryption keys (JOB) will be generated for protecting the content. The storage protocol combines the operating architecture of torrents with the operating architecture of IPFS and organizes them in a decentralized manner.

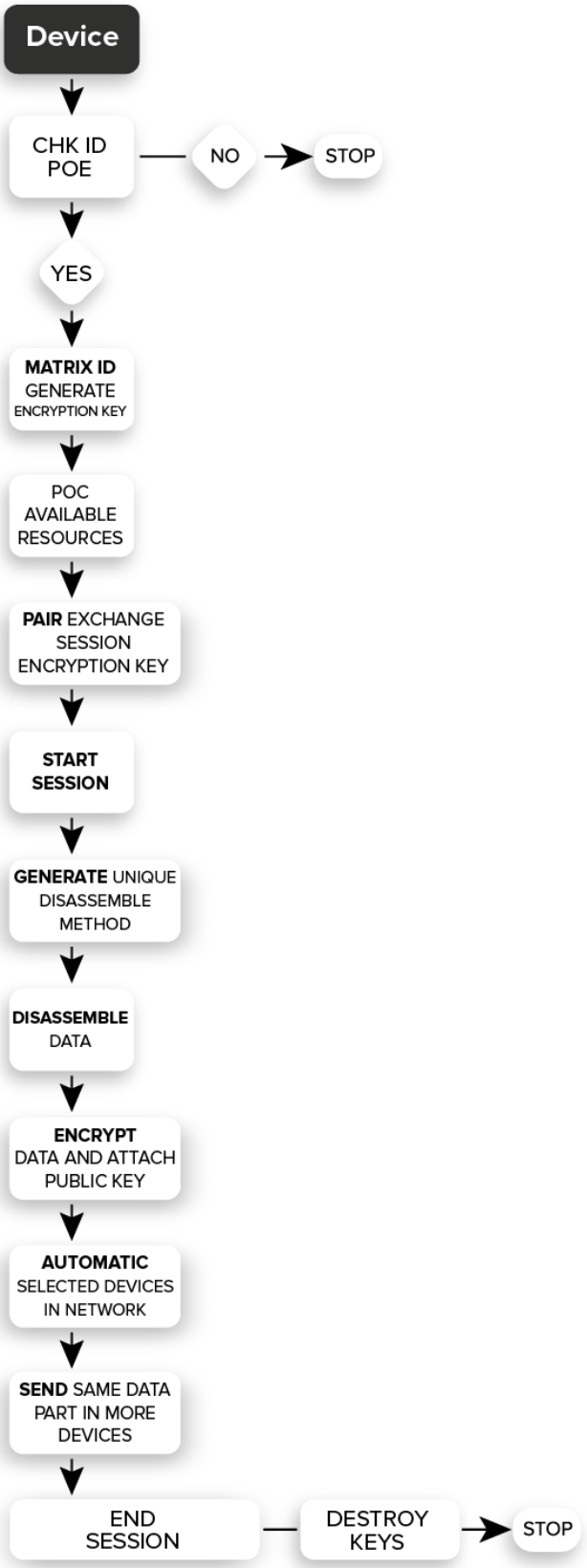
The POE and POC consensus layers verify the validity and conformity of the devices integrated into the system. The storage information is disassembled in several parts through a unique process for each user. The parts are encrypted and sent to devices that have valid storage capacity. The same piece of information is stored in multiple devices ensuring access to the respective information at all times. Because one or more devices can go offline, the protocol constantly checks that the information is valid and available on a minimum number of devices. If it reaches the minimum number, then the information is replicated on another online device. These decentralized algorithms provide 100% uptime for network data.


Device
POE
CHK ID
NO
STOP
OK
APP
CONNECT
POC
CHECK
AVAILABLE RESOURCES
CPU
GPU
RAM
STORAGE
BLOCKCHAIN
READYNESS STATUS
OFFLINE
STOP
ONLINE
REWARD

ACCESS RESOURCES



STORAGE AUTOMATIC MANAGEMENT



7.3 Consensus

Consensus mechanisms (*also known as consensus protocols or consensus algorithms*) allow distributed systems (*networks of computers*) to work together and stay secure. For example, consider a group of people going to the cinema. If there is not a disagreement on a proposed choice of film, then a consensus is achieved.

That is how the consensus mechanism works, and in the blockchain system, it is responsible for all validation mechanisms to maintain the integrity and correctness of the data transferred inside the blockchain. Consensus is a hybrid between *proof of stake, proof of existence, and proof of connectivity*.

In any centralized system, like a database holding key information about driving licenses in a country, a central administrator has the authority to maintain and update the database. The task of making any updates—like adding/deleting/updating names of people who qualified for certain licenses—is performed by a central authority who remains the sole in-charge of maintaining genuine records.

Public blockchains that operate as decentralized, self-regulating systems work on a global scale without any single authority. They involve contributions from hundreds of thousands of participants who work on verification and authentication of transactions occurring on the blockchain.

In such a dynamically changing status of the blockchain, these publicly shared ledgers need an efficient, fair, real-time, functional, reliable, and secure mechanism to ensure that all the transactions occurring on the network are genuine and all participants agree on a consensus on the status of the ledger. This all-important task is performed by the consensus mechanism, which is a set of rules that decides on the legitimacy of contributions made by the various participants (i.e., nodes or transactors) of the blockchain.

This consensus is designed to reduce the risks of a hostile takeover of the network by injecting V.M. or devices and ensure there is no possibility of malicious interference. This eliminates the risk of damaging the network, and, in addition, any node that is identified as fraudulent will lose all tokens deposited during staking.

From the analysis executed on the consensus methods implemented in the functional blockchains available in the market, it was found that once they reach maturity, they cannot cope with the processes of scalability and sustainable and efficient development.

The classic Proof of Work (PoW) process implemented in BTC or ETH blockchains has proven to be inefficient in terms of validation time and power consumption. Already at the maturity stage, the size of these blockchains has reached an increased size and requires more and more efficient devices dedicated to validation tasks. These things

make the PoW consensus type dependent on the advancement of technology, and any delay affects the operation of the entire blockchain mechanism. Elements such as computing power, rewards, investments, benefits, or energy consumption are elements that ensure the functioning of the PoW consensus type and are interdependent. Thus any change of one parameter will affect other parameters and will generate changes in the mechanism.

The consensus layer that underlies the XIDEN blockchain is built on three different protocols that solve the identified issues from other types of blockchains. These three protocols redefine the concept of POW. Therefore, this consensus is an optimized PoW that meets the requirements of scalability and sustainable and efficient development. It is primarily *low energy* and *eco-friendly* as the validation nodes are devices that the user/owner utilize regularly for their daily activities. Thus, there is no requirement for specially built devices that consume a high amount of energy just to perform the mathematical operations required for blockchain operation.

In the consensus mechanism *Mining Resistance* protocol has been implemented. This protocol is developed to protect the network from attacks, increased difficulty, or hostile takeovers to entities or organizations that have the ability to build technologically advanced devices or own large amounts of XDEN.

The three protocols implemented into the XIDEN consensus layer (*POS, POE, and POC*) work independently but are interconnected. When the user wants to initialize a function, the blockchain mechanism accesses the consensus protocols to verify its validity and conformity. If an item is not validated by any protocol then the function will not be executed.

7.3.1 Proof of Stake (PoS)

Proof of Stake is a cryptocurrency consensus mechanism for processing transactions and creating new blocks in a blockchain. A consensus mechanism is a method for validating entries into a distributed database and keeping the database secure. In the case of cryptocurrency, the database is called a blockchain — so the consensus mechanism secures the blockchain.

Proof of Stake reduces the amount of computational work needed to verify blocks and transactions that keep the blockchain, and thus a cryptocurrency, secure. Proof-of-stake changes the way blocks are verified using the machines of token owners. The owners offer their tokens as collateral for the chance to validate blocks. Token owners with staked tokens become "*validators.*"

Validators are then selected randomly to "*mine,*" or validate the block. This system randomizes who gets to "*mine*" rather than using a competition-based mechanism like proof-of-work.

Proof-of-stake is designed to reduce the scalability and environmental sustainability concerns surrounding the proof-of-work (PoW) protocol. Proof-of-work is a competitive approach to verifying transactions, which naturally encourages people to look for ways to gain an advantage, especially since the monetary value is involved.

Bitcoin miners earn Bitcoin by verifying transactions and blocks. However, they pay their operating expenses like electricity and rent with fiat currency. What's really happening then is that miners are exchanging energy for cryptocurrency. The amount of energy required to mine PoW cryptocurrency profoundly affects the market dynamics of pricing and profitability. There are also environmental aspects to consider since PoW mining uses as much energy as a small country.

The PoS mechanism seeks to solve these problems by effectively substituting staking for computational power, whereby an individual's mining ability is randomized by the network. This means there should be a drastic reduction in energy consumption since miners can no longer rely on massive farms of single-purpose hardware to gain an advantage.

Long touted as a threat for cryptocurrency fans, the 66% attack is a concern when PoS is used, but it is very unlikely. A 66% attack is when someone controls 66% of a cryptocurrency and uses that majority to alter the blockchain. In PoS, a group or individual would have to own 66% of the staked cryptocurrency.

It is not only very expensive to have 66% of the staked cryptocurrency—staked currency is collateral for the privilege to "*mine*"—the miner(s) that attempt to revert a block through a 66% attack would lose all of their staked tokens. This creates an incentive for miners to act in good faith for the benefit of the cryptocurrency and the network.[1]

Most other security features of PoS are not advertised, as this might create an opportunity to circumvent security measures. However, most PoS systems have extra safety features in place that add to the inherent security behind blockchains and the PoS mechanisms.

The PoS protocol implemented in the XIDEN Blockchain Consensus Layer is specially developed so that there is no possibility of taking over the network to alter the integrity of data by an entity or organization.

XIDEN PoS consists of two validation layers that are mutually verified.
The layer that validates the transactions and submits the blocks to be registered in the blockchain consists of 30 validation nodes called *Realm MetaNodes* (*RMNodes*). These RMNodes have several active roles in the consensus. The main validation nodes store for the stake the amount of XDEN deposited for staking by the Guardian Nodes and manage the infrastructure composed of these Guardian Nodes.

RMNodes must have at least 2,000,000 XDEN in stake to be active. The first 2,000,000 XDENs are automatically allocated from the Total XDEN Supply when the blockchain is deployed, will be stored forever in the RMNodes, and will be used for validation.

To perform validation, 66% of RMNodes must validate the transaction. This percentage is calculated so that the possibility of taking over a major part of validators with the purpose of altering data is as small as possible.

The 2nd validation layer in the PoS protocol is called *Delegated Stake* and consists of the entire Guardian Nodes network. These light validators check if the Realm MetaNodes changed their status or identity. This validation is performed at each epoch change. The assignment of validation jobs is performed randomly and it is not possible to choose who

to validate. To confirm the integrity of an RMNode's identity, 66% of the active Guardian Nodes must successfully validate.

In order for a Guardian Node to be active in the network, it must deposit the amount of 1000 XDEN required for staking. This quantity is stored in the staking wallet of Realm MetaNode. The total amount of XDEN in the RMNode wallet only affects the rewards.

This does not mean that if one RMNODE holds a higher amount of XDEN than the other RMNodes, it can take control and alter the integrity of data.

7.3.1.1 Unstake

The amount of XDEN submitted for stake can only be controlled by the user who initiated the stake function for that amount. The user can unstake at any time as long as the cooldown is not active.

In order to protect the XIDEN system from flooding, the Cooldown function has been implemented and gas fees are active. Any transaction between two wallets regardless of their purpose must ensure a gas fee amount for transaction validation.

The Cooldown function is activated when a stake or unstake process is executed. That is, the amount of money that executed one of the functions is blocked to perform the stake or unstake process for 50 Epochs.

7.3.1.2 Fault

The amount of XDEN deposited for stake represents a guarantee from the owner who wants to become a validator. Any validator who attempts to defraud the system or alter the integrity of data will be punished by having their amount of XDEN deposited for stake confiscated. The confiscated XDEN amount is then distributed in the network to other validation nodes that have discovered the irregularities. Thus, the participation of validating nodes is encouraged for discovering and reporting irregularities, with a significant amount of XDEN as a reward.

The Realm MetaNodes and Guardian Nodes are independent entities even if the XDEN for stake is deposited in the RMNode wallet. If an RMNode is discovered as an entity that

attempted to defraud, only the amount owned by it will be confiscated and redistributed within the network. If it does not have enough available XDEN for stake then the RMNode will become an inactive node and can no longer validate. The initially

assigned Guardian Nodes to the now inactive RMNode will be redistributed as validation tasks to other active RMNodes in order to maintain the system's activity and integrity.

7.3.2 Proof of Existence (PoE)

Proof of Existence is a consensus layer mechanism that integrates and verifies the identities of the validators within the system. This layer is designed to work together with the other consensus mechanisms - PoS and PoC - in order to replace the classic Proof of Work consensus and improve the PoS through increased security.

Within the XIDEN blockchain, there are entities and devices that have important roles in ensuring the integrity and functionality of the system. Realm MetaNodes, Guardians Nodes, and validators are physical devices with a software component that perform certain functions. In addition to the primary functions, each of these devices runs a function that checks the fingerprint of the other devices at each epoch change and votes in a decentralized way on the integrity of a new device in the system or its status change.

The implementation of this layer came after an analysis of a potential problem. Since validation is performed on the principle of majority voting, there is a possibility that an entity could integrate devices in the form of validators that perform the stake function in order to take control of the majority and thus the network. By implementing Proof of Existence, a device is accepted into the network by decentralized voting and thus no devices can be integrated to intentionally take over the network.

The Proof of Existence consensus layer is built independently of the PoS and PoC layers but works in an interconnected way so that a device cannot perform functions without meeting the other rules.

7.3.2.1 PoE functionality

Each device has unique hardware and software identifiers that form a fingerprint. This fingerprint is encrypted and transformed into a Matrix ID via VOBP.

Initially, at the start of the blockchain, a number of devices are added to the BDX dynamic list in order to have validators that can check the status, integrity, and activity

of the previously integrated devices. Basically, the initial devices that are integrated into BDX will check each other and will play a role of acceptance through a decentralized vote of future devices that will enter the system, also with the purpose of validation.

The initial devices integrated into the system are distributed in a decentralized way to the users of the network who want to become validators. Device validation jobs are randomly distributed so that a device is not validated deliberately, even if it does not meet the criteria.

BDX is a dynamic list that securely stores Matrix IDs of device fingerprints. This list is updated every epoch. This means that devices are verified and receive an accuracy and validity status for the next epoch.

Validation of an existing device in the system needs 10 confirmations from the other existing devices.

Integrating a new device into the system needs 66% votes from existing devices in BDX. Since it is a decentralized network based on the work of each validating device, we have also developed a reward system. Each validator is graded according to the difficulty and the reward mode preferred. Each validator receives XDEN for the work done in the XIDEN network.

7.3.2.2 Fault

The PoE layer works in conjunction with the PoS layer. To become a validator, the device must be integrated into a pool that has a minimum of 1000 XDEN staked as a guarantee that it will perform a correct job.

If a validator is found to have attempted to duplicate the identity of another validator, or to have operated under a different identity, they are punished by removing the XDEN from staking and redistributing it in the network to other validators.

If a device launches attacks on other validators to try to influence their voting or affect their voting ability, then the XDEN from staking is seized and the same procedure of redistribution to the network is applied.

7.3.3 Proof of Connectivity (PoC)

PoC is a layer consensus mechanism that verifies the status, validity, and availability of all the devices integrated into the Xiden blockchain, whether they are validators or devices that are part of the SDR layer.

The validation process in this layer is performed in a decentralized, autonomous, and automated manner without any human intervention. Xiden validators have the role of constantly communicating with the devices within the system so that we have a clear historical record and accurate availability of the devices and resources available in the network.

This layer has been implemented with the purpose of having real-time data about the availability of the integrated devices in a decentralized way.

7.3.3.1 PoC functionality

Each device has a unique fingerprint that is transformed into a Matrix ID to confer security for its identity. Validators open secure VOBP sessions to securely communicate with devices registered in the BDX, but also with devices registered in the SDR layer.

These sessions transmit data sets that contain the Matrix ID, available resources, and the time when the device became online and the time when it became offline. All this data is stored in the form of blocks in a database with blockchain architecture. Both BDX and SDR are up-to-date data on all validation devices integrated into the Xiden network.

Validation of these device data is performed by several validators before being registered. This means that devices open multiple sessions with multiple devices to communicate in real-time.

This consensus layer helps the system to eliminate the possibility for a device to attempt to defraud it by compromising a validator.

To validate this data, PoC requires six confirmations from the available validators within the network.

Depending on the PoC consensus layer, the rewards are distributed for both BDX status and SDR status. Depending on the data accuracy transmitted as resources and on the time of connection to the network, the devices are rewarded with XDEN.

7.3.3.2 Fault

To become a validator, the device must be integrated into a pool that has a minimum of 1000 XDEN staked as a guarantee that it will perform a correct job. If a validator is found to have attempted to transmit erroneous connection data, it will not receive rewards. A device that permanently transmits erroneous data will be banned from participating in the reward system to eliminate the possibility of flooding with invalid data.

If a validator is found to be manipulating device data, it will be punished by confiscating XDEN from the stake and the same procedure of redistribution to the network is applied.

7.4 XDEN - Digital Transferable Asset (DTA)

Xden is defined as a Digital Transferable Asset and serves as native token which fuels the operation of XIDEN blockchain and the applications developed.

The property of the Xden Digital Transferable Asset comes from the fact that it can be transferred between users through the Xiden Network. Xden DTA has no physical representation, it is represented by a decentralized balance constantly updated by validators, based on the transfers executed between users. Each user can generate wallet addresses that are integrated into the decentralized database so that they are integrated

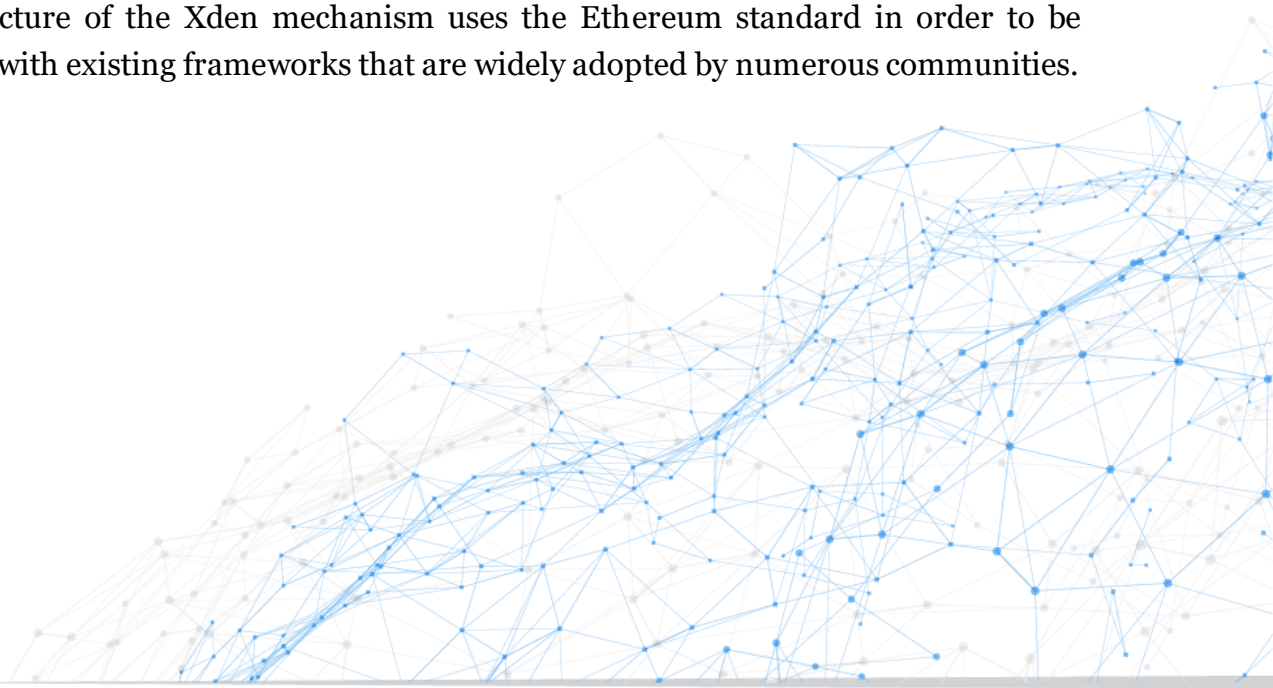
into the system's economy. The Xiden network updates the balance status of wallet addresses based on transfers authorized and validated by validators.

Xden is represented by a dynamic accounting process that is kept at the same time by all validators in the network so that it is transparent, incorruptible and unalterable.

Xden DTA is used for:

- Reward validators which perform validations within the network.
- Reward devices which make their resources available for the network.
- Access mechanism to the services provided by the network's apps.
- Validation mechanism for the smart contracts' generated tokens.

The architecture of the Xden mechanism uses the Ethereum standard in order to be compatible with existing frameworks that are widely adopted by numerous communities.

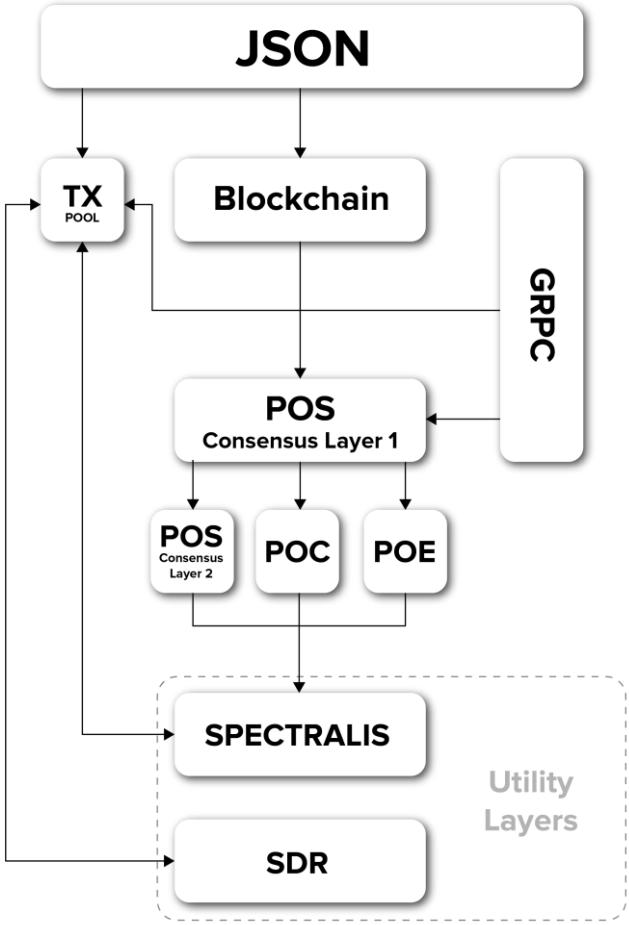


8. Core arhitecture

Xiden is a Polygon Edge fork in which a new consensus algorithm and extended utility layers have been implemented. Polygon Edge architecture was chosen because it is a modular and extensible solution based on a development framework compatible with Ethereum blockchain network, sidechains and general scaling solutions.

Xiden uses existing Polygon modules that allow communication with multiple blockchain networks and comply with the ERC 20 and ERC 721 standards. This allows the initiation of data transfer between blockchains using Bridge solutions.
Xiden is a blockchain with a hybrid architecture, as Xiden's Core is Ethereum and uses Polygon Edge's Proof of Stake architecture, over which we have integrated the PoE and PoC improved Consensus Layer.

The Xiden blockchain layer is combined with the utility module consisting of the Spectralis Network layer and the SDR layer. Thus Xiden is a new approach to the Eth solution using Polygon Edge as an intermediate layer.



8.1 Merkle Tree Algorithm

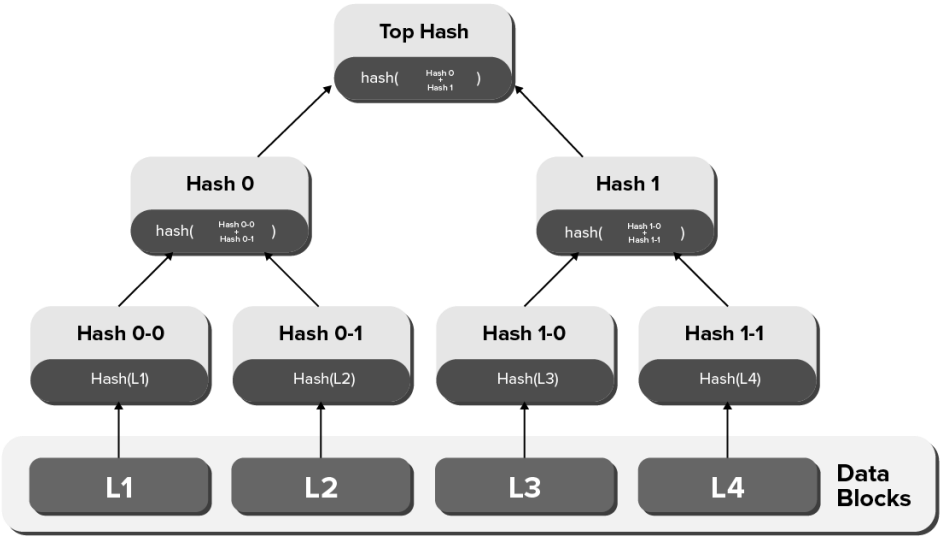
A Merkle tree is a tree data structure, where the leaf nodes contain the hash of a block of data and the non-leaf nodes contain the hash of its children nodes.
In a Merkle tree, any change to the underlying data causes the hash of the node referring to the data to change. Since each parent node hash depends on the data of its children, any change to the data of a child node causes the parent hash to change. This happens to each parent node up to the root node. Therefore, any change to the data at the leaf nodes causes the root node hash to change.

From this, we can derive two important properties:

1. We don't need to compare all the data across the leaf nodes to know if they have the same data. We can just compare the root node hash.

2. If we want to prove that specific data is part of the tree, we can use a technique called *Merkle proofs*. We won't dive into details here but it is an easy and effective way to prove that a piece of data is in the Merkle tree.

The first property is important because it makes it possible to store only a hash of the root node to represent the data at that point in time. This means we only need to store the root hash of the tree representing the block on the blockchain (as opposed to storing all the data in the blockchain) and still keep the data immutable.



8.2 Benefits and Protocol

In various distributed and peer-to-peer systems, data verification is very important. This is because the same data exists in multiple locations. So, if a piece of data is changed in one location, it's important that data is changed everywhere. Data verification is used to make sure data is the same everywhere.

However, it is time-consuming and computationally expensive to check the entirety of each file whenever a system wants to verify data. So, this is why Merkle trees are used. Basically, we want to limit the amount of data being sent over a network (like the Internet) as much as possible. So, instead of sending an entire file over the network, we just send a hash of the file to see if it matches. The protocol goes like this:

- Computer A sends a hash of the file to computer B.
- Computer B checks that hash against the root of the Merkle tree. If there is no difference, we're done! Otherwise, the following will happen:
- If there is a difference in a single hash, computer B will request the roots of the two subtrees of that hash.
- Computer A creates the necessary hashes and sends them back to computer B.

This will be repeated until the inconsistent data blocks(s) are found. It's possible to find more than one data block that is wrong because there might be more than one error in the data.

Note that each time a hash is found to match, we need nn more comparisons at the next level, where nn is the branching factor of the tree.

This algorithm is predicated on the assumption that network I/O takes longer than local I/O to perform hashes. This is especially true because computers can run in parallel, calculating multiple hashes at once.

Because the computers are only sending hashes over the network (not the entire file), this process can be performed very quickly. Plus, if an inconsistent piece of data is found, it's

much easier to insert a small chunk of fixed data than to completely rewrite the entire file to fix the issue.

The reason that Merkle trees are useful in distributed systems is that it is inefficient to check the entirety of a file to check for issues. The reason that Merkle trees are useful in peer-to-peer systems is that they help you verify information, even if some of it comes from an untrusted source (which is a concern in peer-to-peer systems).

The way that Merkle trees can be helpful in a peer-to-peer system has to do with trust. Before you download a file from a peer-to-peer source—like Tor—the root hash is obtained from a trusted source. After that, you can obtain lower nodes of the Merkle tree from untrusted peers. All of these nodes exist in the same tree-like structure described above, and they all are partial representations of the same data.

The nodes from untrusted sources are checked against the trusted hash. If they match the trusted source (meaning they fit into the same Merkle tree), they are accepted and the process continues. If they are no good, they are discarded and searched for again from a different source.

8.3 Use Cases

As stated above, Merkle trees are especially useful in distributed, peer-to-peer systems where the same data should exist in multiple places. These systems use Merkle trees or variants on the Merkle tree in their implementation.

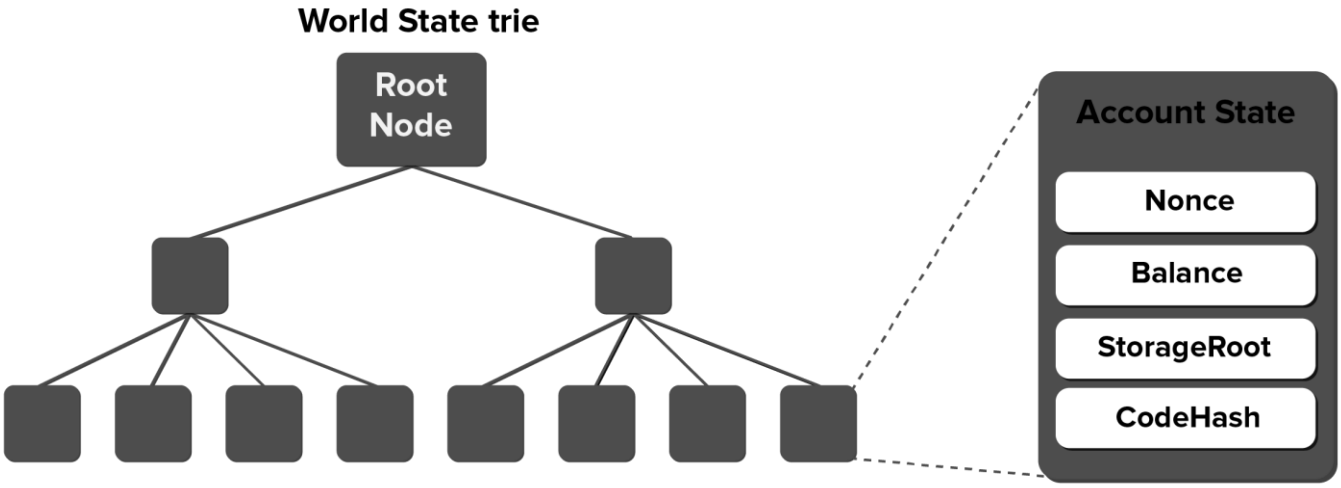
Git is a popular version control system mainly used by programmers. All of the saved files are saved on every user's computer at all times. So, it's very important to check that these changes are consistent across everyone's computer.

Merkle trees can be used to check for inconsistencies in more than just files and basic data structures like the blockchain. Apache Cassandra and other NoSQL systems use Merkle trees to detect inconsistencies between replicas of entire databases. Imagine a website that people use all over the world. That website probably needs databases and servers all over the world so that load times are good. If one of those databases gets

altered, then every single other database needs to be altered in the same way. Hashes can be made of chunks of the databases, and Merkle trees can detect inconsistencies.

8.4 World state

The world state is a mapping between addresses (accounts) and account states. The world state is not stored on the blockchain, but the Yellow Paper states it is expected that implementations store this data in a trie (also referred to as the state database or state trie). The world state can be seen as the global state that is constantly updated by transaction executions.
All the information about Xiden accounts lives in the world state and is stored in the world state trie. If you want to know the balance of an account, or the current state of a smart contract, you query the world state trie to retrieve the account state of that account. We'll describe how this data is stored shortly.



8.5 Account State

In Xiden, there are two types of accounts: External Owned Accounts (EOA) and Contract Accounts. An EOA account is an account that regular users have, that they can use to send Xden to one another and deploy smart contracts with.
A contract account is an account that is created when a smart contract is deployed. Every smart contract has its own Xiden account.

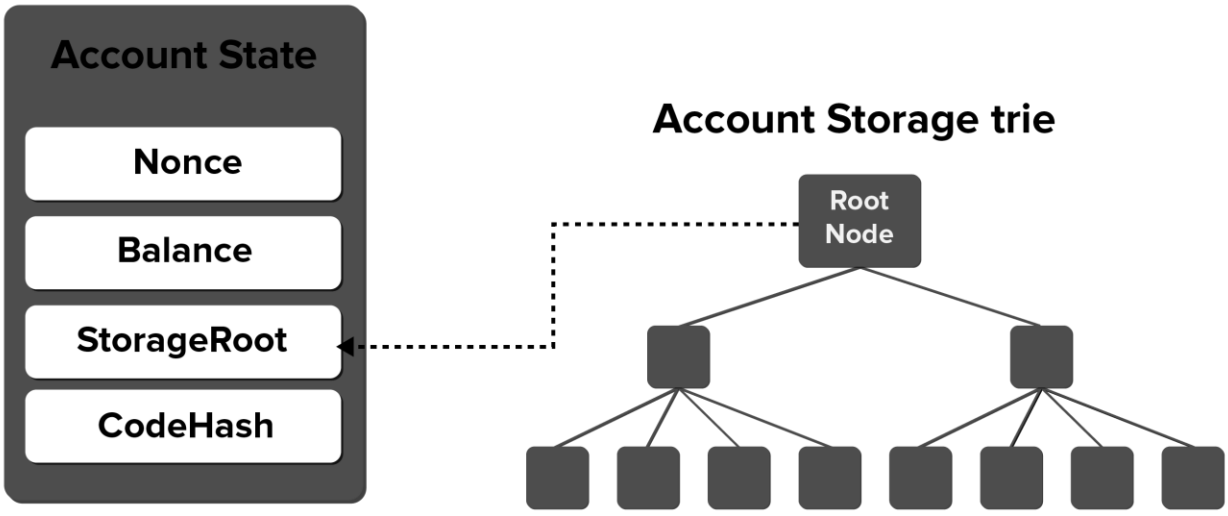
The account state contains information about an Xiden account. For example, it stores how much Xden an account has, and the number of transactions sent by the account. Each account has an account state.

Let's take a look into each one of the fields in the account state:

- nonce - Number of transactions sent from this address (if this is an External Owned Account - EOA) or the number of contract creations made by this account
- balance - Total Xden (in Wei) owned by this account
- storageRoot - Hash of the root node of the account storage trie
- codeHash - For contract accounts, the hash of the EVM code of this account. For EOAs, this will be empty.

One important detail about the account state is that all fields (except the codeHash) are mutable. For example, when one account sends some Xden to another, the nonce will be incremented, and the balance will be updated to reflect the new balance.

One of the consequences of the codeHash being immutable is that if you deploy a contract with a bug, you can't update the same contract. You need to deploy a new contract (the buggy version will be available forever). This is why it is important to use tools like Truffle to develop and test your smart contracts and follow the best practices when working with Solidity.
The Account Storage trie is where the data associated with an account is stored. This is only relevant for Contract Accounts, as for EOAs the storageRoot is empty, and the codeHash is the hash of an empty string.



8.6 Transactions

Transactions are what make the state change from the current state to the next state. In Xiden, we have three types of transactions:

1. Transactions that transfer value between two EOAs (e.g, change the sender and receiver account balances)
2. Transactions that send a message call to a contract (e.g, set a value in the smart contract by sending a message call that executes a setter method)
3. Transactions that deploy a contract (therefore, create an account, the contract account)

These are the fields of a transaction:

- nonce - Number of transactions sent by the account that created the transaction
- gasPrice - The value that will be paid per unit of gas for the computation costs of executing this transaction
- gasLimit - Maximum amount of gas to be used while executing this transaction
- to
    - If this transaction is transferring Xden, the address of the EOA account that will receive a value transfer
    - If this transaction is sending a message to a contract (e.g, calling a method in the smart contract), this is address of the contract

    - If this transaction is creating a contract, this value is always empty

- value
    - If this transaction is transferring Xden, the amount that will be transferred to the recipient account
    - If this transaction is sending a message to a contract, the amount payable by the smart contract receiving the message
    - If this transaction is creating a contract, this is the amount that will be added to the balance of the created contract

- v, r, s - Values used in the cryptographic signature of the transaction used to determine the sender of the transaction
- data (only for value transfer and sending a message call to a smart contract) -Input data of the message call (e.g, imagine you are trying to execute a setter method in your smart contract, the data field would contain the identifier of the setter method, and the value that should be passed as a parameter)
- init (only for contract creation) - The EVM-code utilized for initialization of the contract.

8.7 State Transition Function

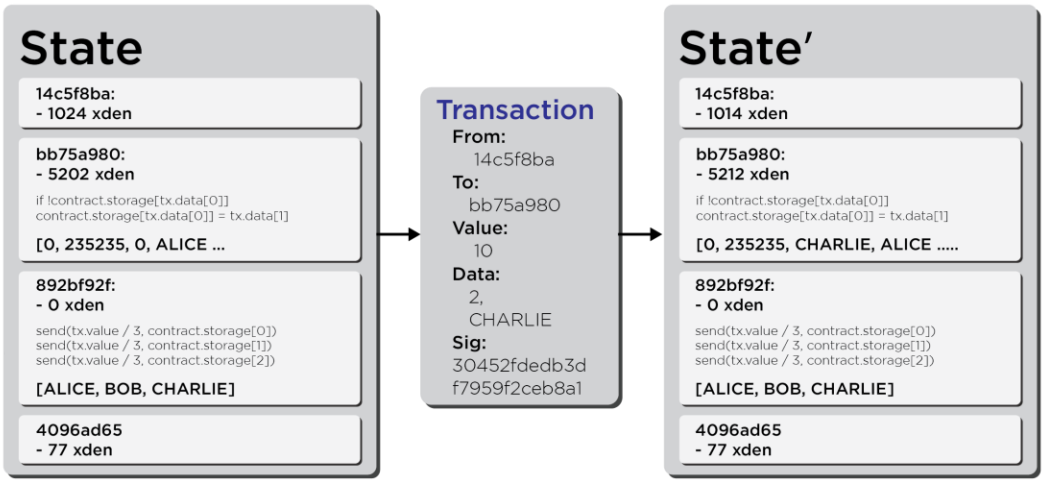
The XIDEN state transition function, APPLY(S,TX) -> S' can be defined as follows:

Check if the transaction is well-formed (ie. has the right number of values), the signature is valid, and the nonce matches the nonce in the sender's account. If not, return an error.

Calculate the transaction fee as STARTGAS * GASPRICE, and determine the sending address from the signature. Subtract the fee from the sender's account balance and increment the sender's nonce. If there is not enough balance to spend, return an error.
Initialize GAS = STARTGAS, and take off a certain quantity of gas per byte to pay for the bytes in the transaction.

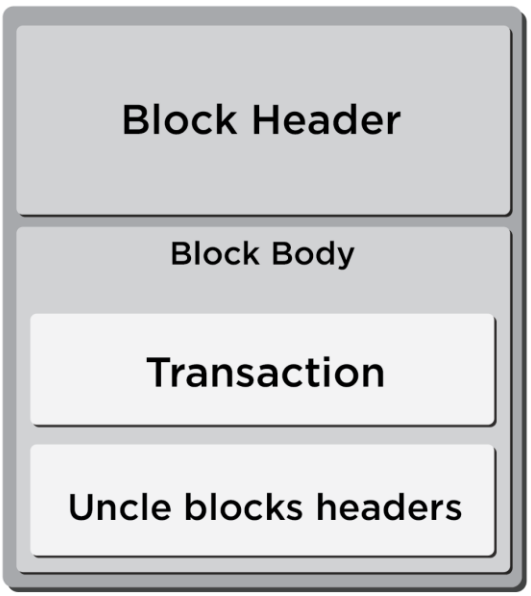
Transfer the transaction value from the sender's account to the receiving account. If the receiving account does not yet exist, create it. If the receiving account is a contract, run the contract's code either to completion or until the execution runs out of gas. If the value

transfer failed because the sender did not have enough money, or the code execution ran out of gas, revert all state changes except the payment of the fees, and add the fees to the validator's account. Otherwise, refund the fees for all remaining gas to the sender, and send the fees paid for gas consumed to the validator.



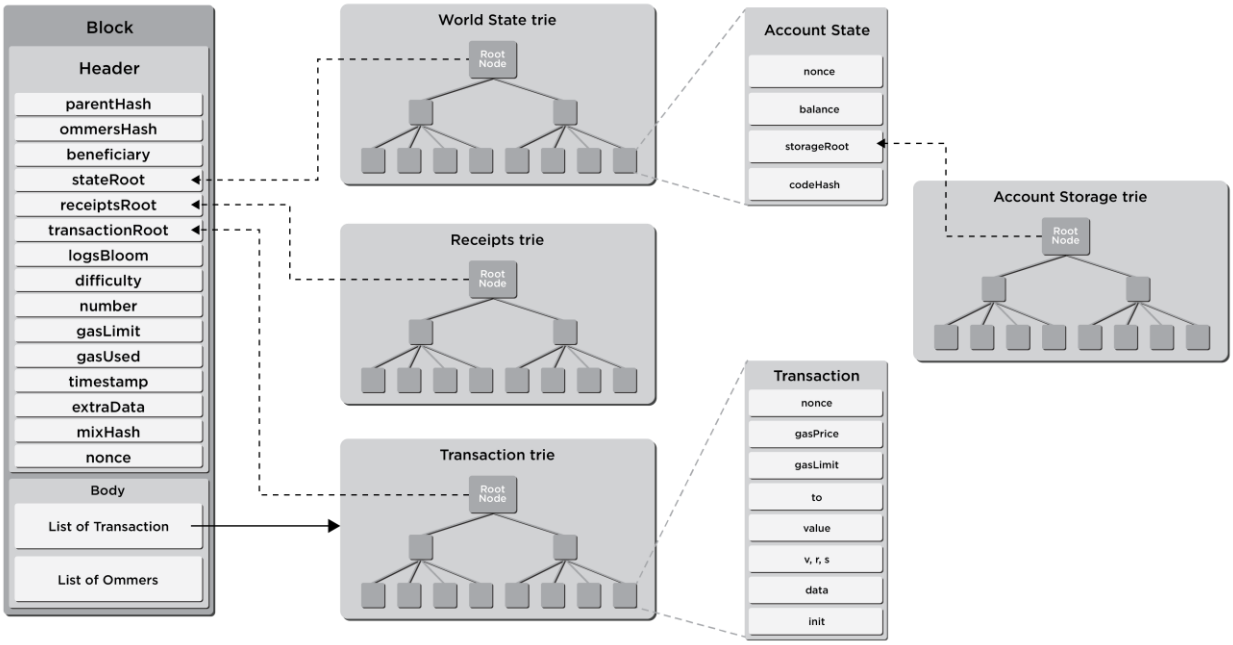
The block header is divided into two parts, the block header and the block body. The block header is the blockchain part of Xiden. This is the structure that contains the hash of its predecessor block (also known as parent block), building a cryptographically guaranteed chain.

The block body contains a list of transactions that have been included in this block, and a list of uncle (ommer) block headers.



The block header contains the following fields:

- parentHash - Hash of the block header from the previous block. Each block contains a hash of the previous block, all the way to the first block in the chain. This is how all the data is protected against modifications (any modification in a previous block would change the hash of all blocks after the modified block)
- ommersHash - Hash of the uncle blocks headers part of the block body
- beneficiary - Xiden account that will get fees for mining this block
- stateRoot - Hash of the root node of the world state trie (after all transactions are executed)
- transactionsRoot - Hash of the root node of the transactions trie. This trie contains all transactions in the block body
- receiptsRoot - Every time a transaction is executed, Xiden generates a transaction receipt that contains information about the transaction execution. This field is the hash of the root node of the transactions receipt trie
- logsBloom - Bloom filter that can be used to find out if logs were generated on transactions in this block (if you want more details check this Stack Overflow answer). This avoids storing logs in the block and facilitates storage saving.
- difficulty - Difficulty level of this block. This is a measure of how hard it was to mine this block.
- number - Number of ancestor blocks. This represents the height of the chain (how many blocks are in the chain). The genesis block has number zero
- gasLimit - Each transaction consumes gas. The gas limit specifies the maximum gas that can be used by the transactions included in the block. It is a way to limit the number of transactions in a block
- gasUsed - Sum of the gas cost of each transaction in the block
- timestamp - Unix timestamp when the block was created.
- extraData - Arbitrary byte array that can contain anything. When a miner is creating the block, it can choose to add anything in this field
- mixHash - Hash used to verify that a block has been mined properly
- nonce - Same as the mixHash, this value is used to verify that a block has been mined properly.



8.9 Code Execution

The code in Xiden contracts is written in a low-level, stack-based bytecode language, referred to as "EVM code". The code consists of a series of bytes, where each byte represents an operation. In general, code execution is an infinite loop that consists of repeatedly carrying out the operation at the current program counter (which begins at zero) and then incrementing the program counter by one, until the end of the code is reached or an error or STOP or RETURN instruction is detected.

The operations have access to three types of space in which to store data:

- The stack, a last-in-first-out container to which values can be pushed and popped
- Memory, an infinitely expandable byte array
- The contract's long-term storage, a key/value store. Unlike stack and memory, which reset after computation ends, storage persists for the long term.

The code can also access the value, sender and data of the incoming message, as well as block header data, and the code can also return a byte array of data as an output.
The formal execution model of EVM code is surprisingly simple. While the EVM is running, its full computational state can be defined by the tuple (block_state, transaction,

message, code, memory, stack, pc, gas), where block_state is the global state containing all accounts and includes balances and storage.

At the start of every round of execution, the current instruction is found by taking the pcth byte of code (or 0 if pc >= len(code)), and each instruction has its own definition in terms of how it affects the tuple. For example, ADD pops two items off the stack and pushes their sum, reduces gas by 1 and increments pc by 1, and SSTORE pushes the top two items off the stack and inserts the second item into the contract's storage at the index specified by the first item. Although there are many ways to optimize EVM's execution via just-in-time compilation, a basic implementation of Xiden can be done in a few hundred lines of code.

9. Ecosystem Components

9.1 Age

The Age represents a period in which the Xiden blockchain functions in accordance with a set of specific rules specially created for its consolidated and stable development. The Age is calculated in the number of Epochs, with the transition from an age to the next being automatically executed.

Our accumulated experience in this area has shown us that a blockchain needs approximately 10 years in order to reach maturity without facing any more problems that can affect its functionality. Therefore, the Xiden blockchain has been designed to be developed in 3 essential phases in which to be closely followed and modified in such a way that it will adapt to the technological needs and requirements of the present time.

**The Xiden blockchain can undergo architectural, structural, or functional modifications during each Age transition.*

All changes will be made according to the following steps:

1. **Proposal - Voting**
2. **Testing**
3. **Implementation - Voting**

Modification proposals can be made not only by developers, but also by the community, while the validation or implementation decision of these proposals will be achieved through decentralized voting by RMNode owners. During these transition periods, the Xiden network may be subjected to functionality issues. Due to the fact that Xiden is a decentralized technology, if such an issue arises that affects one or more users, no one is liable for any effects or losses caused by these issues. Any possible problems discovered that can affect security can be remediated through network changes and updates during any Age, not only during the transition between Ages.

9.1.1 Phase 1: the BIG BANG Age

The first Age is known as BIG BANG and represents the birth of the XIDEN Ecosystem. The genesis block is formed during this time, containing the entire quantity of XDEN that serves as fuel for the XIDEN blockchain.

Each Guardian Node will receive 1000 XDEN in the locked version in order to be able to open a Private KraterPool so as to have the necessary quantity of XDEN for the PoS layer. A user can connect multiple validators in this KraterPool to unlock the locked quantity of XDEN or to receive XDEN from the RMNodes based on the amount of work executed in the network.

9.1.1.1 Motivation

This Age's main objective is the creation of an active Xiden Network community attracted through rewards and through the Xiden blockchain's utility. The distribution of XDEN represents a necessity during this incipient phase of the Xiden Network's development in order for any user to be able to participate as a validator through Delegated Staking.

Due to the fact that it is a technology in its initial stages with multiple extensive applicabilities, we consider that an accommodation period is necessary in which the users will be able to test the XIDEN network in order to understand it and to try to integrate it as a technology in the their products, services, and activities.

9.1.1.2 Specifications

Period duration: 477 Epochs
Number of active RMNodes: 10
Public Difficulty: Available
Private Difficulty: Available
Public KraterPool: Unavailable
Private KraterPool: Available
RMNode Reward Custom: Unavailable

9.1.2 Phase 2: METEORA

The second phase of the Xiden blockchain carries the name Meteora. The distribution of 1000 XDEN in the locked version at the initial configuration of a Guardian Node is stopped in this Age and the Public Krater Pool function is made available, making it possible for users to use the XDEN in circulation.

Both the locked and unlocked (that can be transferred between users) XDEN tokens are considered circulating XDEN. Users are thus offered the possibility to unite in DAO-type management groups in order to activate the POS layer necessary for validation.

9.1.2.1 Motivation

The Meteora Age's main purpose is to strengthen the relationships between the network's users and to encourage the creation of Public KraterPools, through which the network's users can pool their XDEN so as to create a decentralized-governed entity that can fulfill the POS function.

9.1.2.2 Specifications

Period duration: 477 Epochs
Number of active RMNodes: 20

Public Difficulty: Available
Private Difficulty: Available
Public KraterPool: Available
Private KraterPool: Available
RMNode Reward Custom: Unavailable

9.1.3 Phase 3: ATLAS

The third and final phase of the Xiden blockchain is known as Atlas. This Age defines the Xiden blockchain's operating rules, these being definite for this technology's entire lifespan.
The RMNode customization functions are made available in the Atlas Age to allow the community to enter into an active competition so as to organize the RMNodes in the most efficient and effective possible method.

During this final phase the quantity of locked XDEN that could not be unlocked will be burned to reduce the Total Supply and to eliminate the possibility of inflation.

9.1.3.1 Motivation

During the Atlas age, which is presented as being the final development phase, we consider that the Xiden blockchain will have reached maturity and from this point forward can sustain itself through the activity defined through time by the community's needs.
The purpose of this Age is to ensure the continuity of the Xiden ecosystem through the technology's maturity and its implementation in users' future necessities.
We are of the opinion that from this moment the network will already have an active community that will not only grow, but also expand Xiden's technological possibilities.
We also consider that during the Atlas Age XDEN's value will be built in a sustainable way through the technology's utility so that it can function and develop in a lasting way over the time to come.

9.1.3.2 Specifications:

Period duration: 477 Epochs
Number of active RMNodes: 30
Public Difficulty: Available
Private Difficulty: Unavailable
Public KraterPool: Available
Private KraterPool: Available
RMNode Reward Custom: Available

9.2 Realm MetaNode (RMNode)

The RMNode is a central validator that is part of the POS consensus layer of the Xiden blockchain. It is a participant that fulfills multiple functions to ensure the network's operation.

The Xiden blockchain has a network of 30 RMNodes that are activated according to each specific Age. RMNodes have a decentralized distribution from both an ownership and geolocation perspective. An RMNode is made up of a hardware layer and a software layer that must meet a set of technical requirements in order to function.

Minimum Specifications:

- RAM: 64 GB
- CPU: 16-core
- Storage: 1 TB SSD
- Bandwidth: 1 Gbit/s

Functions:

- Block Producer and Submiter
- Transaction Validator
- Central Staking Entity
- Gas Collector
- Reward distribution manager

9.2.1 Functionality

In order to work as a validator and block producer on the Xiden blockchain, an RMNode must hold 2,000,000 XDEN locked in stake as collateral for the correctness of the validations it will execute in exchange for the reward it will receive.

RMNodes have a management function in POS Layer 2 for Delegate Staking. All the Guardian Nodes that hold a connection through the KraterPool with the other devices must transfer 1000 XDEN to an RMNode in order to become validators in Layer 2 of consensus.

RMNodes add everything that is transferred from the Guardian Nodes to the total stake quantity. The quantity of XDEN deposited by Guardian Nodes cannot be accessed or manipulated by an RMNode. The stake and unstake functions are controlled through the smart contract, which allows only the Guardian Node OWNER to execute them. Thus, an RMNode is just a holder of the XDEN locked for stake.

The main function of a RMNode is as a block producer and submitter for the network's blocks. It has the role of validating the Xiden network's transactions and to permanently maintain the integrity of the data transfer registry within the network.

An RMNode has the role of collecting the gas that is used when executing a data transfer between two wallets. Gas management is customizable based on the Age in which the blockchain is at the time. In the BigBang and Meteora Ages the gas is distributed integrally to the KraterPools connected to the RMNode. In the last Age, Atlas, the gas can be distributed percentually between the RMNodes and the KraterPools connected to them.

An RMNode's chance to validate and to receive rewards is influenced by the quantity of XDEN locked in stake - the chance is higher the more deposited XDEN it has. Thus, a competition begins in the Atlas Age between Realm MetaNodes to attract as many KraterPools to connect to them in order to have as big a chance as possible to validate.

An RMNode can be configured by any user that accomplishes the KYC process and owns 2,000,000 unlocked XDEN. An RMNode can be owned by DAO-type organizations but also by centralized organizations.

The quantity of 2,000,000 XDEN staked by the Node Owner can be withdrawn at any time by executing the UNSTAKE function. An RMNode's owner cannot control the XDEN deposited by KraterPools. If the total staked quantity of XDEN owned by the RMNode decreases below 2,000,000, then it becomes inactive and can no longer validate or integrate blocks.

9.3 KraterPool

KraterPool is a platform that centralizes the validators and smart devices and that is managed by a Guardian Node. The role of this platform is to integrate smart devices into the Xiden network with the purpose of ensuring them an environment in which to execute certain functions.

The KraterPool is divided into two categories:

9.3.1 Private KraterPool

- This private KraterPool can be owned by a single entity and cannot be customized for the reward distribution to each device or validator based on each of their contributions.
- A single wallet is attached to the platform for the transfer of XDEN for POS but also to deposit the rewards obtained as a result of validations executed or resources integrated.
- It allows the integration of 10 validators and 10 smart devices for the SDR layer.
- It can be managed by a single Guardian Node that makes the connection to the RMNode.
- Boost Power: Up to 10% * Total Cycles

9.3.2 Public KraterPool

- It is available starting with the Meteora Age.
- The platform can be owned and administered by an entity or a DAO.
- SDR validators and devices can organize into groups and a wallet can be attached to each group.
- The platform has a central wallet managed by a smart contract that automatically collects XDEN for POS. Each group wallet can control the quantity it sends to the POS wallet.
- Reward distribution is automatic and is managed through the smart contract.
- Public KraterPools can be managed through decentralized voting by all the participants of that particular KraterPool.
- The management and connection to an RMNode is made through a single Guardian Node.
- It allows the integration of 20 validators and 20 smart devices for the SDR layer.
- Boost Power : Up to 20% * Total Cycle

9.3.3 KraterPool Functionality

To open a KraterPool a user or an organization needs an active and fully functional Guardian Node. The Guardian Node establishes a connection with an RMNode.

To activate a KraterPool the user must attach a wallet containing 1000 Xden in Locked or Unlocked version that will be sent and locked in the POS Layer. Having these conditions met, the Guardian Node is automatically registered as a validator in the KraterPool. For the pool to be active, the Guardian Node needs to be maintained online at all times in order to ensure its connection to the RMNode.

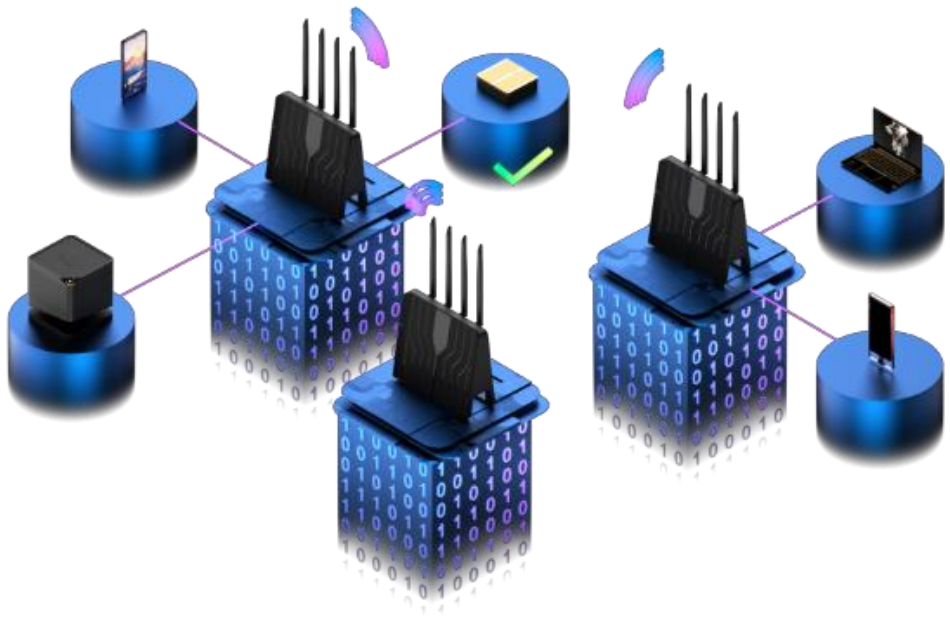
More validators that can be non-Guardian Nodes (as per SDR layer) can be integrated into the KraterPool. This will increase the validation power and will allow owners to perform more validations and to receive a larger amount of XDENs, depending on the system's difficulty.

The KraterPool also has the function that allows smart devices to make available their resources for the SDR Layer. KraterPool is actually a management platform that distributes jobs from DApps to embedded devices. The validation devices that are integrated in the KraterPool are under the direct management of the platform in the SDR Layer.

Non-validating smart devices can only be integrated into the KraterPool for SDR Layer via Guardian Nodes that are part of the Spectralis Network. These devices boost the overall computing power. Each device integrated in the KraterPool increases the total validation power by 1% so that the Kraterpool will receive more rewards for the resources integrated in the SDR layer.

The KraterPool connection with the integrated devices both as validators and in the SDR Layer but also with Guardian Node and RMNode is a secure connection empowered by the VOBP secure communication protocol. Strong cryptographic protocols are used so that all data sets are secure. Devices communication is P2P and uses Matrix ID with unique encryption keys for each data packet transferred. Communication is performed automatically between devices and encryption key management is done for each device so the same encryption is not used for multiple devices or multiple communication sessions.

Each device generates its own protocol to securely transmit information to any other device. After each session, the encryption keys are destroyed so that there is no risk of compromising data packets that may have been intercepted by various methods.



10. XIDEN network economics

10.1 XDEN - Digital Transferable Asset (DTA)

Xden is defined as a Digital Transferable Asset and serves as native token which fuels the operation of XIDEN blockchain and the applications developed.

The property of the Xden Digital Transferable Asset comes from the fact that it can be transferred between users through the Xiden Network. Xden DTA has no physical representation, it is represented by a decentralized balance constantly updated by validators, based on the transfers executed between users. Each user can generate wallet addresses that are integrated into the decentralized database so that they are integrated

into the system's economy. The Xiden network updates the balance status of wallet addresses based on transfers authorized and validated by validators.

Xden is represented by a dynamic accounting process that is kept at the same time by all validators in the network so that it is transparent, incorruptible and unalterable.

Xden DTA is used for:

- Reward validators which perform validations within the network.
- Reward devices which make their resources available for the network.
- Access mechanism to the services provided by the network's apps.
- Validation mechanism for the smart contracts' generated tokens.

The architecture of the Xden mechanism uses the Ethereum standard in order to be compatible with existing frameworks that are widely adopted by numerous communities.

10.2 Xden Distribution

The Total Supply of Xden is determined from the time the Xiden mainnet is deployed. In order to have sustainable development, the economy of the system is based on a strategy of gradual and consolidated growth at each stage. The Xden proposed amount ensures that the needs of the system are automatically managed by smart contracts that meet the security standard.

The wallet addresses are public and can be tracked in the Xiden Explorer for complete transparency and traceability of the Xiden blockchain.

Total Supply
Amount: 300.000.000 XDEN

RMNode POS Layer 1 Supply
Amount: 60.000.000 XDEN

Guardian Node POS Layer 2 Locked Supply
Amount: 100.000.000 XDEN

Validation Reward Supply
Amount: 100.000.000 XDEN

Developers Supply
Amount: 30.000.000 XDEN

Marketing & Airdrop Supply
Amount: 10.000.000 XDEN

Total Supply represents the total amount of Xden generated to be distributed according to the system's requirements.

Circulating Supply is the total amount of Xden that is unlocked and can be transferred between users. This amount is dynamic and will increase as the rewards will be allocated according to the work performed by users in the network.

10.2.1 RMNode POS Layer 1 Supply

In order for the RMNode to work in the Layer 1 POS consensus, it needs to have 2.000.000 Xden locked in stake. This amount is locked by the system by default, in a specially assigned wallet, which is managed by a smart contract from the moment the RMNode is activated, according to Age.

This amount can reach Circulation Supply if the owner executes the Unstake function.

10.2.2 Guardian Node POS Layer 2 Locked Supply

To offer users the ability to enable a KraterPool via a Guardian Node and to integrate validators into the network, an amount of locked Xden has been allocated for use in Layer 2 POS in Delegated Staking.

This amount is distributed to each Guardian Node in the BigBang Age. Each Guardian Node will receive 1000 locked Xden. The amount can reach Circulating Supply if it is unlocked by users by means of validating according to Private Difficulty requirements.

From the Atlas Age, the unlocked amount will be redirected to the Validation Reward Supply to be distributed according to Public Difficulty to network validators.

10.2.3 Validation Reward Supply (VRS)

Validation Reward Supply is the amount of Xden allocated for network validators. It will be distributed according to the work submitted by validators on both the consensus layer and the SDR layer.

This amount is fixed in the first two Ages (BigBang & Meteora). The total amount of Xden in the Reward Supply will be increased in the Atlas Age by allocating locked Xden from Guardian Node Locked Supply.

**This amount is distributed to users only if they validate according to Public Difficulty requirements.*

10.2.4 Developers Supply

The amount of Xden allocated for the development team, partners and advisors. This amount is used to reward the work of development teams, but also to ensure distribution to future partners. The purpose of this amount is to build over time a sustainable ecosystem adopted by as many entities or organizations as possible.

10.2.5 Marketing & Airdrop Supply

From the beginning, an amount of Xden is strategically allocated to be distributed so that we can develop an active community around Xiden and to make the project and its benefits known through various promotion campaigns.

10.3 Rewards

The Xiden system's functionality is built around its utility through the Spectralis Network Layer and through the SDR layer, using XDEN as its fuel mechanism.

The two utility layers become functional only if there exists an active community to populate them with users and devices. Dapps are built on these two utility layers so as to develop a decentralized ecosystem whose technological modules can be integrated into the products, services, and technologies present in users' various activities.

In order to make sure this system is beneficial to all its participants, we have developed a reward structure through which each participant can receive XDEN fuel for activity rendered.

Xiden reward types:
Gas Transaction Reward & Validating Reward
Unlock Guardian Node Locked Supply
Custom SDR Reward
Custom RMNode Reward

10.3.1 Gas Transaction Reward & Validating Reward

In order to ensure a quantity of XDEN in such a way that the devices within the KraterPool can receive a considerable reward, the Gas Transaction Amount has been supplemented with a quantity of Xden. There will not be many transactions to distribute a substantial quantity of XDEN to validators immediately after the birth of the Xiden system, which led to the creation of this solution.
This quantity of XDEN is distributed to the KraterPool wallet based on the total power (Cycles), which is made up of Validator Cycles plus Boost from the devices included in the SDR Layer and integrated into the KraterPool. This reward can be received only if the KraterPool is configured for Public Difficulty.
The total gas owned by the RMNode is added to the quantity of XDEN from the Validation Reward Supply and is then distributed according to the amount of work done by the participating devices.

11.3.2 Unlock Guardian Node Locked Supply

Each Guardian Node receives 1000 XDEN for POS and to be able to validate. This quantity of XDEN can be unlocked by configuring the KraterPool to Private Difficulty, which means that the user receives a reward for the work done - a faster reward than for Public Difficulty.

Unlocking XDEN does not lead to an increase of the amount of Xden owned by a user. They simply unlock a quantity of XDEN which they can then continue to use for staking or which they can transfer to other users.

11.3.3 Custom SDR Reward

Dapps developers have the possibility to customize the distribution of the XDEN they have received for users using their Dapps. Thus, they can establish a certain percentage

to be distributed towards the SDR Layer so that the devices that make their resources available can receive more XDEN.

10.3.4 Custom RMNode Reward

During the Atlas Age, RMNode owners can customize XDEN distribution towards the connected KraterPools. This will create a competitiveness between RMNodes, leading to the development of a real economy. The RMNode that has the most staked XDEN will have the chance to validate more, thus receiving gas. This will cause the KraterPool validators to receive more XDEN according to the work done.

10.4 Difficulty

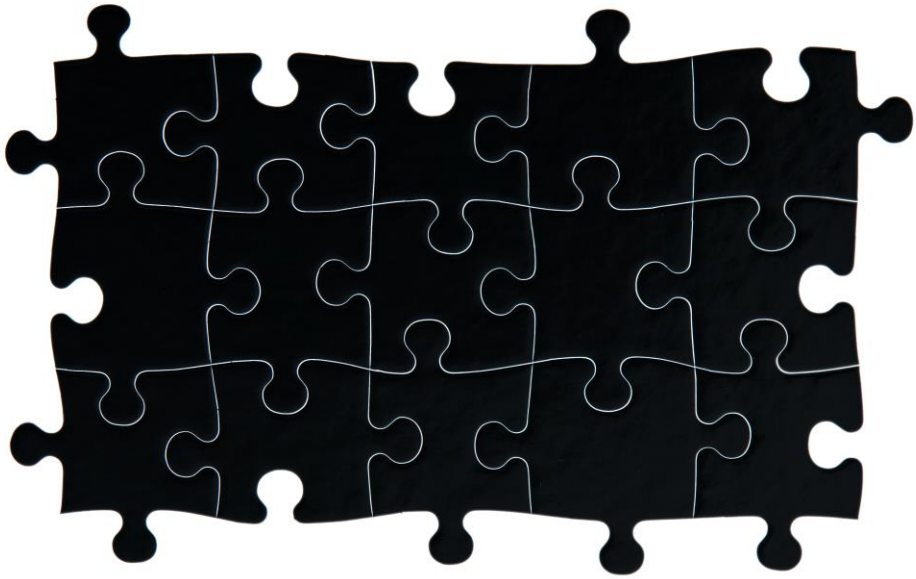
Difficulty is the indicator resulting from a Xiden blockchain calculation formula. These formulas are influenced by Total Supply, Circulating Supply, Validation Speed, and the Ecosystem's Need for XDEN. It is split into Public Difficulty and Private Difficulty to meet the needs of each validator owner but also to maintain an efficient distribution of XDEN in the system.

10.4.1 Public Difficulty

This is the difficulty with which the quantity of XDEN is released to the validators in order to maintain a stable economy without causing an XDEN inflation in the ecosystem.
Users have the possibility to receive rewards from Gas and from the SDR Layer.
Public difficulty is modified every 43,200 blocks so that XDEN distribution is well adjusted. It is available in the BigBang, Meteora, and Atlas Ages.

10.4.2 Private Difficulty

This is the calculation formula through which the locked XDEN initially received by a Guardian Node in the BigBang Age is unlocked. The users have the possibility to configure the KraterPools in order for these to unlock the locked XDEN.
Private difficulty is modified every 43,200 blocks in order to optimize the flux of XDEN that may reach circulating supply.
It is available in the BigBang and Meteora Ages.



11. Application Environment

This environment is designed to maintain the interoperability of networked systems and to keep up to date with the versions of DApps that are built to run on the Blockchain core.

The Blockchain core supports the development of applications in the form of smart contracts and, thus, multiple DApps and NFT protocols can be developed to run on the network infrastructure, but also to use the computing and storage power of the devices provided by the entire network.

Blockchain is compatible with the following Frameworks so that it is easily accessible to a wider community and interoperable with other technologies in various blockchains such as:

- ***HardHAT*** - an environment developers use to test, compile, deploy and debug DApps based on the Ethereum blockchain. As such, it helps coders and developers to manage many of the tasks that are inherent to developing dApps and smart contracts.
- ***Truffle*** - a development environment, testing framework, and asset pipeline for blockchains using the Ethereum Virtual Machine (EVM).
- ***Web3.js*** - a collection of libraries that allow users to interact with a local or remote Ethereum node using an HTTP or IPC connection. The web3 JavaScript library interacts with the Ethereum blockchain, it can retrieve user accounts, send transactions, interact with smart contracts, etc.
- ***Ethers*** - The ethers.js library aims to be a complete and compact library for interacting with the Ethereum Blockchain and its ecosystem.
- ***Metamask*** - a browser extension designed to make accessing Ethereum's Dapp ecosystem easier. It also serves as a wallet for holding ERC-20 tokens allowing users to access services built on the network via the wallet.
- ***Solidity*** - an object-oriented, high-level programming language used to create smart contracts that automate transactions on the blockchain.
- ***EVM*** - The Ethereum Virtual Machine is the software platform that developers can use to create decentralized applications (DApps) on Ethereum.
- ***RemixIDE*** - an open-source web and desktop application. It fosters a fast development cycle and has a variety of plugins with intuitive GUIs.

12. Conclusion

The *Xiden Blockchain* is a decentralized network that will support and facilitate the integration of all smart devices in a system that utilizes blockchain technology to create opportunities for increasing income by ensuring optimal and efficient use of all available resources of the respective devices.

The Xiden network combines technologies such as the *Internet of Things* with blockchain to develop a protocol that will allow smart devices to perform tasks automatically and autonomously, thus ensuring high-speed data and operational procedures validation.

Xiden aims to become an open-source system that will provide the opportunity for users around the world to have a free and permanent internet connection regardless of their location or device.