Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

A Survey on Blockchain-based Internet Service Architecture: Requirements, Challenges, Trends and Future

WENLI YANG[1], ERFAN AGHASIAN[1] (Member, IEEE), SAURABH GARG[1] (Member, IEEE), DAVID HERBERT[1], LEANDRO DISIUTA[1] (MEMBER, IEEE) AND BYEONG KANG[1]

[1]Discipline of ICT, School of TED, University of Tasmania, Sandy bay, Australia (e-mail: yang.wenli@utas.edu.au, erfan.aghasian@utas.edu.au, saurabh.garg@utas.edu.au, david.herbert@utas.edu.au, leandro.disiuta@utas.edu.au, byeong.kang@utas.edu.au)

Corresponding author: Saurabh. Garg (e-mail: Saurabh.Garg@utas.edu.au).

ABSTRACT The emergence of Internet protocol suites and packet-switching technologies tend to considerations of security, privacy, scalability, and reliability in layered Internet service architectures. The existing service systems allow us to access big data, but few studies focus on the fundamental security and stability in these systems, especially when they involve large-scale networks with overloaded private information. In this research, we explored the blockchain-based mechanism that aims to improve the critical features of traditional Internet services, including autonomous and decentralized processing, smart contractual enforcement of goals, and traceable trustworthiness in tamper-proof transactions. Furthermore, we provide a comprehensive review to conceptualize the blockchain-based framework to develop decentralized protocols for the extensive number of Internet services. This comprehensive survey aims to address blockchain integration to secure Internet services and identify the critical requirements of developing a decentralized trustworthy Internet service. Additionally, we present a case study of block-chain based IoT for neuro-informatics to illustrate the potential applications of blockchain architectures. Finally, we summarize the trends and challenges of blockchain architectures that benefit a multitude of disciplines across all internet service fields of interest.

INDEX TERMS Internet service architecture; Blockchain; Security, Decentralized network; Multi-plane.

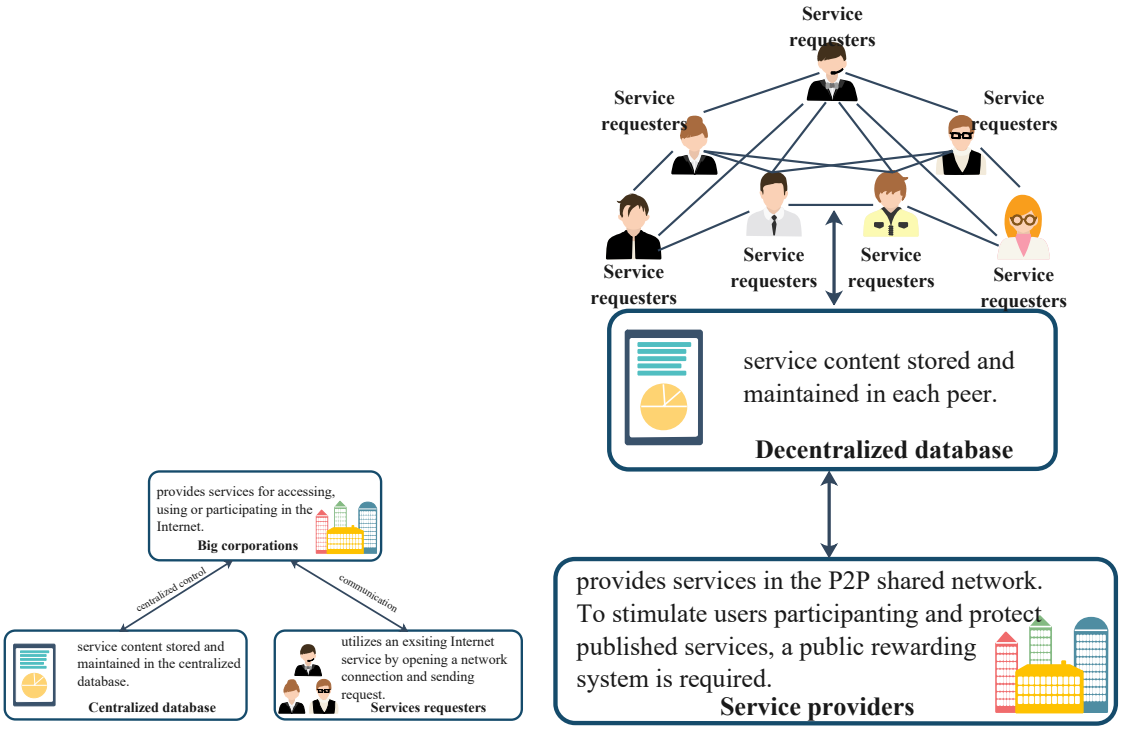
I. INTRODUCTION

THE original Internet service architecture was to build a common decentralized network with equal participation, that communicated using peer to peer interconnectivity without relying on a single computer [1]. Another important consideration of the original Internet's plan was that computers must be interoperable among dissimilar systems, so that more devices could be a part of the network.

However, after the first dot-com bubble [2], large corporations (such as Google and Amazon) realized that the largest value gained from this decentralized network involves gathering, organizing, and monetizing information through centralized services. These companies therefore built their value by growing massive centralized databases using freely-obtainable private, personal data that is then deployed on the Internet, and these changes led to the Internet's service architecture partially deviating from the original architectural intentions.

Today, the Internet is physically decentralized, but it contains critical components for data processing, social media, advertising and crowdsourcing that use large centralized services. The traditional Internet service consists of three groups of roles: service requesters, big corporations (service provider) and the centralized database (Figure 1). Service requesters are responsible for requesting services from service providers who provide various Internet services. Almost every service provider has its own data center, where it stores user data and runs its applications. As shown in Figure 1(a), it can be seen that as the public has a greater reliance on such services, it is of substantial fiscal benefit for the big corporations to keep their services maintained and remain proprietary.

However, such concentrated centralization has also created a growing number of issues [3]. First, traditional Internet service architectures are vulnerable to denial of service, which makes the services unavailable, such as the global financial



(a) Centralized Internet services

(b) Decentralized Internet services

FIGURE 1: The old and new way of Internet service

crisis (GFC) of 2008 [4]. Secondly, the majority of Internet services rely on the centralized database, which suffers from a single point of failure, as they provide attackers a single target to hack. For instance, when centralized services such as LinkedIn or Gmail Services fail, all the websites and applications that depend on them stop working. Third, users' identity information (e.g. name, email address and phone number) and task solutions are saved in a centralized database, which now may contain many aspects of concern to data privacy. Users can never tell about what goes on behind the walled gardens of centralized services. Therefore, they do not precisely know how much data these services collect about them and how that data is used. Furthermore, when a service requester and provider are in dispute, they need a trustworthy network to give a subjective arbitration, which may lead to a behavior known as 'error-reporting'. In short, it can be seen that the existing Internet service implementations achieve information transmission and sharing in a decentralized manner, but there has not been sufficient scrutiny and action in guaranteeing transactional trust and the exchange of wealth or value across the Internet.

Therefore, building a trustworthy Internet is a very important and fundamental task. There have been many research topics to deal with part of the above mentioned issues in Internet services. These topics are mainly related to attack detection and prevention, failing with single-point solutions and privacy protection. For example, data anonymization [5] [6], differential privacy [7] [8] and encryption schemes [9] [10] [11] are proposed to protect personal data privacy. Reputation-based security mechanisms are designed to identify and predict transaction safety based on overall use and reputation over a wide community of users. Distributed architectures are proposed to address the single point of failure problem. However, at present, none of the existing work has solved all issues simultaneously. Therefore, our research is motivated by how to design a decentralized framework with distributed data verification, scalability and security, where blockchain technology potentially fulfills this purpose - as shown in Figure 1(b).

Blockchain is a relatively new platform technology, which is widely known and it was developed primarily to use with Bitcoin cryptocurrency [12]. Blockchain is based on decentralized networking and one of its main characteristics is to guarantee the safety and integrity of data. The technology is scalable and robust and all participant nodes provide resources in a fair manner, which alleviates many-to-one traffic flow bottlenecks. This technique decreases traffic delays and defeats the errors due to a single point of failure [13] [14].

To address scalability and trust concerns, Hart claims that a network framework cannot be based on a single entity to manage the network's infrastructure. Instead it requires peer to peer (P2P) resource management [15]. Therefore,

blockchain would be an ideal solution to secure the Internet in addition to the various services layered upon it. This would increase the fundamental baseline security and as blockchain has excellent extensibility features such as scriptable programmability, and it supports new types of layered Internet services. In short, the contributions of this research are as follow:

1) Conceptualize a comprehensive survey on the current challenges of Internet service architectures and describe the vision of building a blockchain-based architecture to guide future design and implementation of decentralized protocols.
2) Present a concrete and key requirement of building a decentralized Internet service based on the blockchain technology to reach its full potential.
3) Discuss the future trends and challenges in the design of blockchain-based Internet service architecture for the future directions of research and development.
4) Demonstrate the blockchain-based internet service architecture through a blockchain-based IoT for neuro-informatics application to explain the feasibility.

The rest of the paper is organized as follows: Section II presents the related work, and we analyze the current challenges in Internet service architectures in Section III. Section IV describes the key concepts of the security mechanisms between blockchain and traditional solution through different planes. Section V describes the vision of building blockchain-based Internet service architecture and a case study of blockchain-based IoT for neuro-informatics is demonstrated to explain the integrated blockchain-based architecture. The detailed technical requirements under the architecture are described in Section VI. Finally, we conclude the future challenges and trends in Section VII.

II. RELATED WORK

Since 2009 to now, blockchain has attracted a considerable amount of attention in applied fields ranging from Bitcoin to financial services, supply-chain management, Internet of things, Internet services and so on. Many researchers think 'Internet+Blockchain' represents an ideal solution to build a new Internet architecture with value at a low cost. In this section, the existing blockchain-related academic papers are mainly reviewed from four primary areas: constructive technologies for blockchain, applications for blockchain, evaluation and opportunities as shown in table 1.

Constructive technologies for blockchain: this section focuses on improving the current components of blockchain such as data structure design, security enhancement and privacy protection as well as current consensus protocol improvement. The research on data structure was firstly based on hash-tables, however with the significant growth of blockchain usage, several new data structures with scalable, light-weight and decentralized features were proposed. In this regard, Directed Acyclic Graph (DAG) for maintaining transaction information and RadixDLT for scaling linearly in an unbounded and efficient manner are the proposed structures. Some researchers have discussed how to make a possible solution using blockchain for building mutual trust within society. For example, an automated manager without any third-party intervention was presented to turn a blockchain into access control. The decentralized system was proposed to retain transactional privacy from public view using cryptographic primitives such as zero-knowledge proofs. In addition, many researchers focus on consensus protocols, such as the improvement of the performance and efficiency of existing protocols as well as the creation of new consensus protocols.

Applications for blockchain: there are many papers which discuss improving previous applications, creating new applications, while designing smart contracts for different applications represents another key hot topic. Since a huge amount of the current Internet services are developed in a centralized manner, researchers have tried to explore decentralized structures to deal with increasing security problems and limitations of the current Internet services. Except for the initial financial applications, more research focusing on some certain areas related to Internet services, such as the Internet of Things(IoT) [14], public and social services [16], cloud services [17] and other Internet services such as reputation [18] and crowdsourcing [19] are also being conducted.

Evaluation and challenges: since blockchain combines multiple technologies to ensure an immutable, irrevocable and traceable ledger, there are some related works centred on evaluating and analyzing the overhead and performance of the proposed decentralized architecture, including throughput and latency, scalability, fault tolerance, protocol and network security. On the basis of evaluation, some challenges about current blockchain platforms can be found, such as storage capacity of blockchain, the process of automation, the security and efficiency of smart contracts and so on.

In summary, the above mentioned works are limited to some specific Internet services, whilst in comparison, this research mainly aims to conceptualize a blockchain-based decentralized framework with much broader goals, such as providing a direction or vision to guide future design and implementation of decentralized protocols, and presenting the key requirements of building blockchain-based internet service architecture for the future research and development.

III. CURRENT CHALLENGES IN INTERNET SERVICE ARCHITECTURES

Internet service architectures typically cover the basic communication between heterogeneous networks that may differ internally in terms of hardware, software, or technical design. Building a secure, layered service architecture is vitally important to ensure that all commercial requirements as well as the user's demands are achieved, but not at the expense of a robust and trusted security model. Software security mechanisms have evolved from a single-tier architecture, to two-tier architecture, and to the current multi-tier architecture [3], [46] (refer to table 2). Through this evolution, it can easily be seen that the existing security mechanisms are centralized or

TABLE 1: Summarization of current research topics related to blockchain-based Internet services

Research problem	Objective	Key points	References
Constructive technologies for blockchain	improving the current components of blockchain	data structure design security enhancing and privacy protection consensus protocol improvement	[20] [21] [22] [23] [24] [25] [26]
Applications for blockchain	improving previous application, creating new application and designing smart contracts for different applications	Finance IoT Public and social services Cloud Services Other Internet services	[27] [28] [29] [30] [31] [32] [16] [33] [34] [35] [36] [19] [37] [38]
Evaluation and challenges	evaluating of blockchain platforms and analyzing future trends and challenges	evaluating of blockchain platforms trends and challenges	[39] [40] [41] [42] [43] [44] [45]

have a locally centralized architecture.

Single-tier service architecture: this architecture is used for simple Internet services in which the user interface and data access are combined into one single program integrated into a single platform [47]. In this architecture, the control and data plane share the same host server (figure 2). This architecture is very easy to implement in the early stages of service deployment, however, it is unable to satisfy complex applications as it introduces a single point of processing (bottlenecks) as well as a single point of failure. Also, the security mechanisms for single-tier services consider authentication and authorization. Authentication is used to verify the identity of a user, while authorization manages what a user can or cannot access, focusing on permissions.

Two-tier service architecture: this architecture separates the control plane acting as an interface for a single host machine, from the data plane which is used to store data on another server [48]. Separating these two components into different locations represents a two-tier architecture (as depicted in Figure 3). Although the database server is separated from the single server deployed in single-tier architecture, servers still remain a potential single point of failure within the two-tier service architecture.

Multi-tier service architecture: this architecture divides different components into multiple planes according to their functions. Each plane runs on a separated server [49]. Multi-tier can be classified into two main types depending on the control mechanism: distributed and centralized control (as shown in figures 4(a) and 4(b)). A distributed control plane allocates control protocol functions across multiple processor levels in the network, while a centralized control plane, like the SDN network architecture, aims to improve network performance in terms of providing centralized network management capabilities [50]. Both methods provide compartmentalization and avoid a single point of failure. Although the implementation of a multi-tier service architecture could help to enhance system security, it still uses several controllers to concentrate on published Internet services or applications.

Existing Internet service architectures can utilize high speed data transmission and enable the efficient use of resources. However, as shown in Table 2, there are several limitations and challenges that need to be addressed, especially with regards to application security and scalability issues [51]. Some of these issues are:

Data obtained from non-verified sources: Currently, the huge amount of power which services such as Google and Facebook have as reliable sources of information, has turned them into gatekeepers of information - the public can only believe them based on trust. For example, if Google wants to express some fake and misleading content to the users, there is virtually no method to stop them. The recent anecdotal swing of the 2016 USA federal election to the Republican Party due to the spread of fake news via trusted social network platforms like Facebook and Twitter highlighted that the trust can be misplaced [52].

Many sources rely on their own data: Almost every Internet company or business has its own data centre, where it stores user data and runs its own applications. This requires some serious security, as they are large and obvious targets for hackers attempting to steal sensitive data. But, due to self-reliance, when centralized services such as LinkedIn or Gmail fail, all associated applications that depend on them are unavailable. This creates a very visible and widespread concern when such services fail.

Lack of security for private data: The existing Internet service architectures also involve privacy concern problems. Users are unaware of what occurs behind the walled gardens of centralized Internet services. In other words, users are not notified of how much of their private data is being gathered by these services and what purposes the data will be used for. Recent (May 2018) [53] legislative changes in Europe with the introduction of the General Data Protection Regulation (GDPR) highlight the seriousness of the issue. Application service providers with clients in Europe scrambled, some seemingly at the last moment, to be compliant with the legislation. Unfortunately, such compliance did not necessarily extend to clients in other non-European countries, and a universal, international regulatory protection is currently lacking.

The birth and development of blockchain aims to solve the privacy and trust problems faced by the current Internet services. It would remove single points of failure due to distributed ledgers. Blockchain would prevent single data storage based on peer-to-peer networking, as opposed to traditional client-server models. Blockchain would also enhance

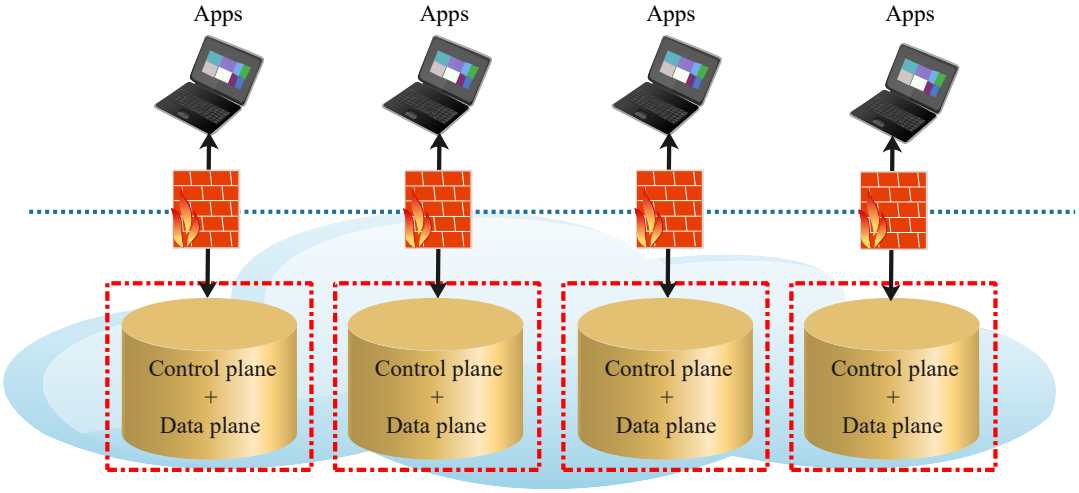


FIGURE 2: Single-tier service architecture

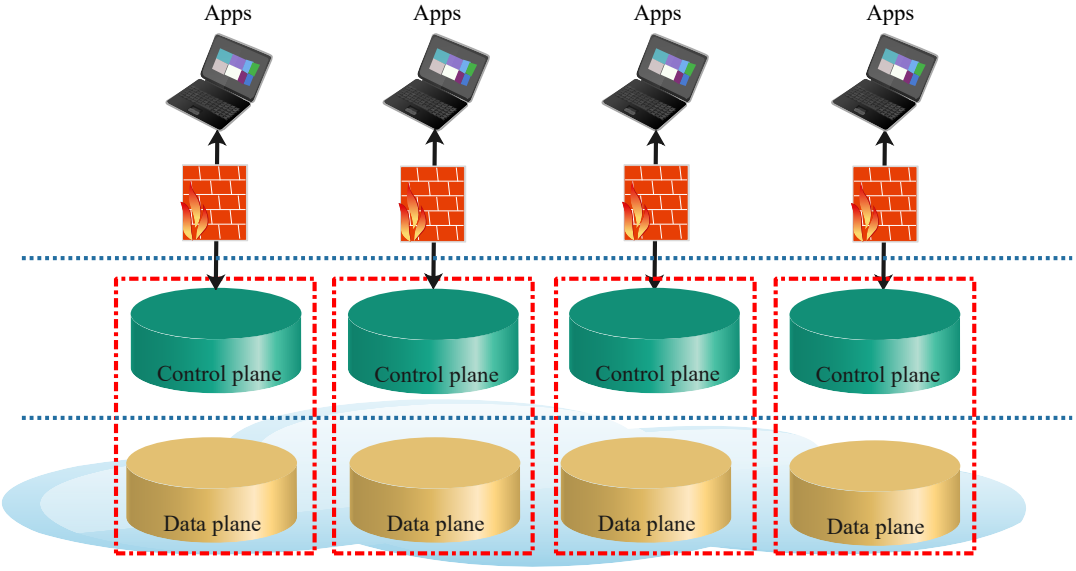


FIGURE 3: Two-tier service architecture

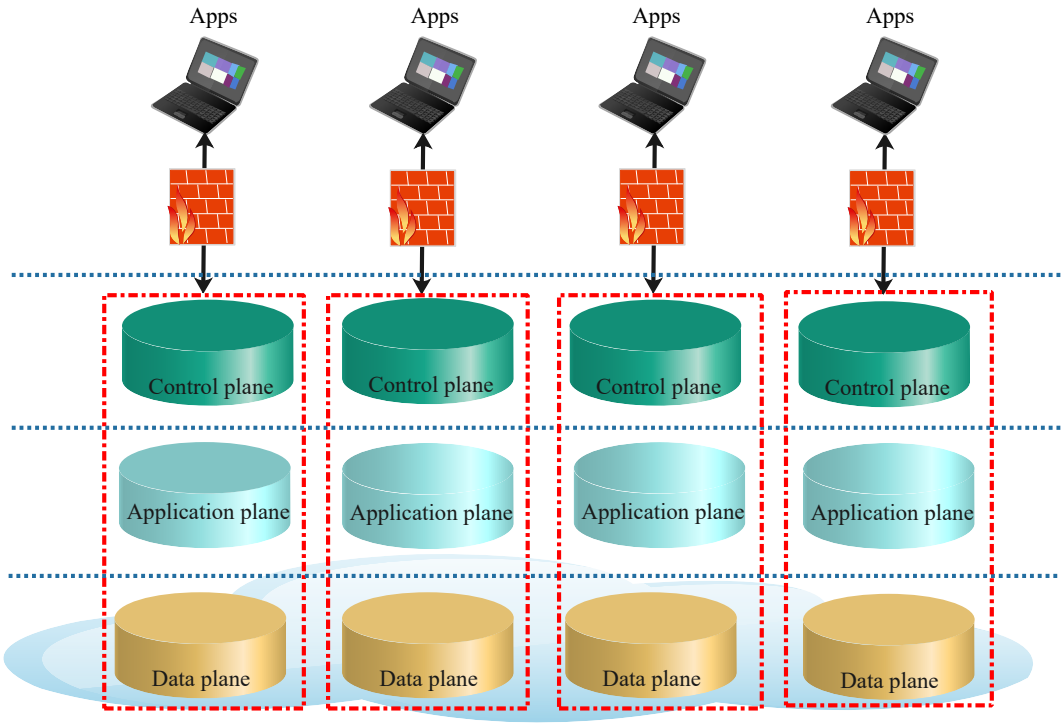
competition by avoiding lock-ins and giving users full control of their data [54].

IV. COMPARISON BETWEEN BLOCKCHAIN *VS* TRADITIONAL SECURITY MECHANISM FOR THE INTERNET SERVICES

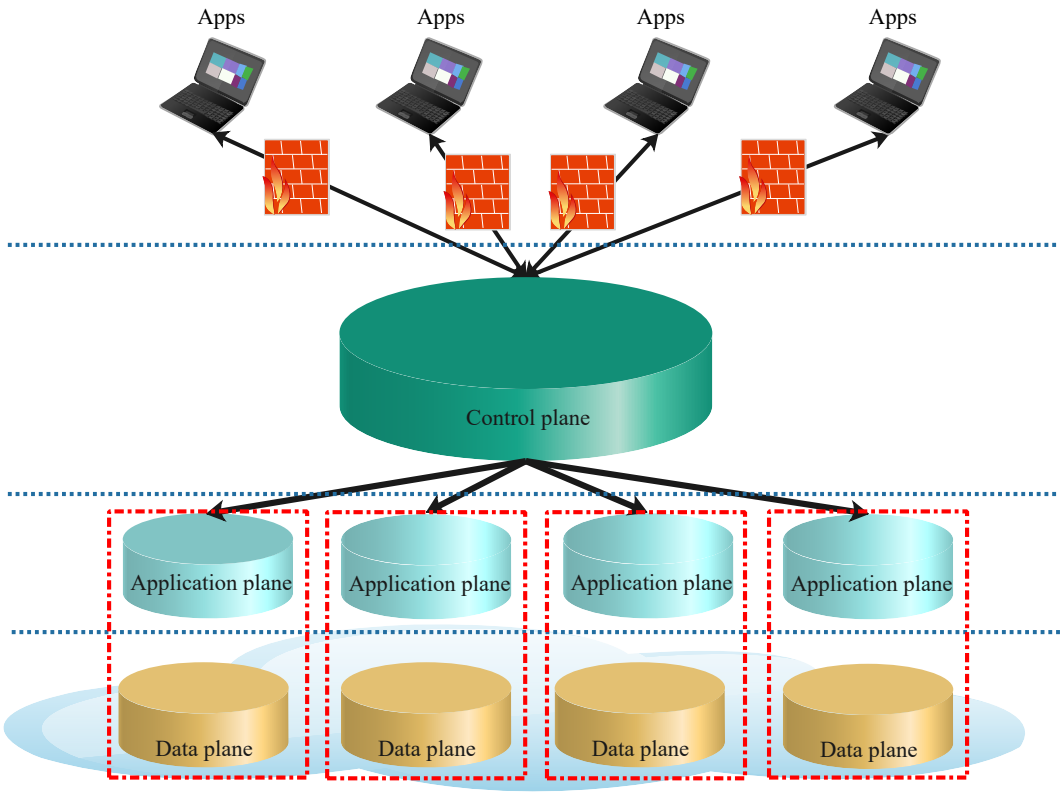
Contrary to traditional security mechanisms for common Internet services, blockchain technology is based on decentralized transaction and data management which is able to provide anonymity, safety and data integrity [55]. There is no need for a third-party organization to control the blockchain transactions, making this field a vast area of research to deal with limitations and enhancements within the current Internet service architecture. Blockchain combines multiple technologies to ensure an immutable, irrevocable and traceable blockchain ledger [56]. This section will discuss the security of blockchain technology through different planes in the Internet service architecture (data, control, network and application planes) compared with traditional centralized-based mechanism.

A. DATA PLANE

The data plane manages the required data, such as data storage, sharing and retrieval. The main difference between a traditional database and the blockchain database is data structure. The most common data structure of traditional Internet services is a database table that in essence consists of a two-dimensional array indexed by a row and column value. Other data structures such as b-tree and a user-defined vector are also in common use. Traditional database management is operated by one or several controllers on the basis of a hierarchical data structure and have been principally secured against hackers over network security mechanisms like network-based intrusion detection systems (IDS) and



(a) Decentralized control



(b) Centralized control

FIGURE 4: Multi-tier service architecture

TABLE 2: Comparison of existing Internet service architectures

Evolution	Single-tier	Two-tier	Multi-tier	
			Distributed control	Centralized control
Typical application	Local desktop database e.g. Microsoft Access with local presentation services.	Desktop applications, e.g. spreadsheet and word processing via file sharing.	Almost all web applications user a three or Multi-tier architecture.	
Points of failure or maintenance	Easy to maintain as there is only a single point of failure.	Easy to maintain and modification is relatively easy.	More difficult to maintain.	Easy to maintain and deploy.
Easy of development /Creation	Simple to create. Standardized separation of data and presentation e.g. MVC (model, view, controller) framework assists with development.	Slightly more complex to create and develop issues such as contention and concurrency need to be considered.	More complex, need to pre-establish lower plane details such as data sharing and transmission capabilities.	Fast creation. Apply to everywhere with the single framework.
Network performance	Lower relative performance, and difficult to support large and complex network traffic access patterns.	Communication is faster than single-tier, but the server request response rate is a bottleneck, as a result it can cause data integrity issues.	High performance, but network operations cannot be easily reprogrammed or re-tasked.	Highest performance without a device-centric configuration on each location.
Scalability	Very Poor.	Still poor scalability, application performance will be degraded with increasing user count.	Each tier can scale horizontally, but at the expense of increasing complexity or effort.	Tiers (except the control plane) can scale horizontally.
Security	Locally centralized, rely on authentication and authorization between one server and users.	Locally centralized, similar with single-tier, if one server crashes, the corresponding application will stop.	Locally centralized, although client does not have direct access to the database, it still relies on authentication and authorization between servers and users.	Totally centralized, but will be highly unstable.

firewalls. However, these security mechanisms are still high risk. For instance, if one table in the database is corrupted, the operation of the whole database is potentially compromised and the data access would be lost [51]. Even if appropriate maintenance processes are in place, data loss may still occur even after a rollback or table restoration. Unlike the traditional Internet services, blockchain is based on Distributed Ledger Technology (DLT) [57], which is spread across several nodes or computing devices. Blockchain uses a chain data structure based on cryptography algorithms (such as Merkel tree and hash function) consisting of a transaction, block and a chain as shown in Figure 5. A transaction is an operation that causes a change to the whole ledger between nodes and a block is composed of a header and a long list of transactions. All nodes in the system maintain a long chain of blocks which are linked and secured against tampering by the application of cryptography techniques. The composite structure "block (complete history) + chain (complete verification) = timestamp" provides an integrated and immutable database. This structure provides a better data integrity for the system when compared with traditional services.

B. CONTROL PLANE

The control plane advertises and displays information related to services available on the Internet. The control protocols used in Internet services can be divided into three main types: centralized, distributed and decentralized models. Contemporary Internet services use a globally centralized controller or a locally-centralized controller to communicate with the data plane as well as the application plane. Centralized control is usually comprised of one device that deals with tasks such as I/O connectivity, motion control and so on [58].By using a centralized mechanism, administrators have the ability to effectively manage the traffic data from different locations. Since the control calculations are performed in the central device, the computing capacity demands have to be significantly higher with corresponding and security requirements which have to mitigate the associated risks. In order to overcome this, a distributed control structure was illustrated [59] using locations and facilities re-optimizing, which shows good scalability in simulations. Research has found that using a distributed model to provide Internet service could prevent service break resulting from the loss of networking or power [60]. Although the implementation of distributed control model focuses on allocating control protocol functions across multiple processor levels in the network, they had a centralized platform to provide services, which is not consistent with the requirement of building Internet services in a decentralized way. Decentralization is basically to distribute constraint and dominance from the central authority to peripheries in order to weaken the centralized organizations' function with secured benefits [61]–[63], which can makes use of the information exchanged between distributed controllers allocated within the control plane. This process can ease the access control and revocation

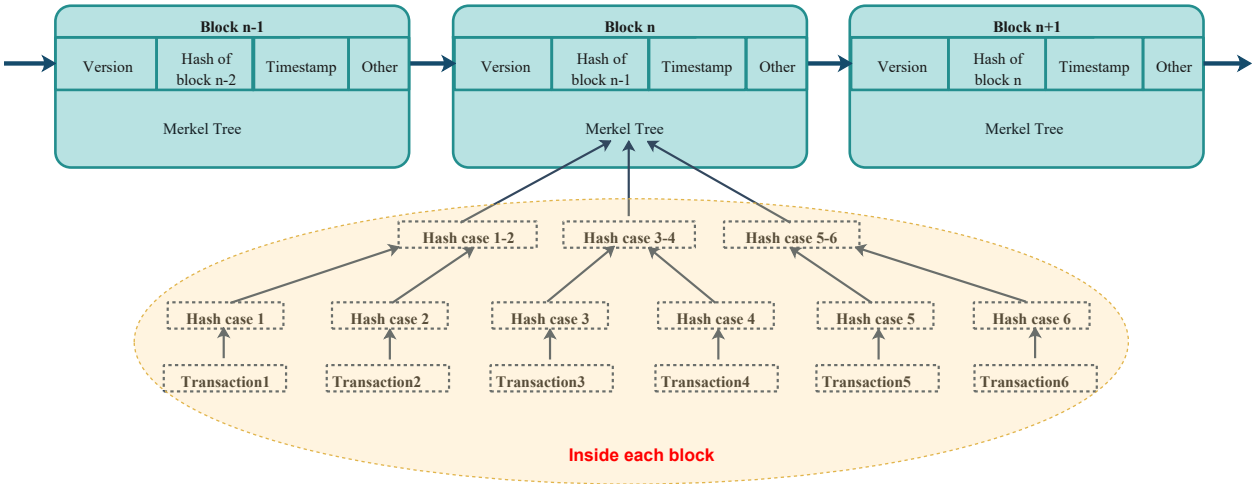


FIGURE 5: The basic data structure of blockchain.

within the system [58]. The blockchain utilizes decentralized control as independent organizations or individuals are usually distributed geographically. The main advantage of decentralized control is that the presentation authority is delegated to the individual nodes throughout the network rather than limiting it to a few executive nodes. Figure 6 depicts a comparison between the centralized, distributed and decentralized control plane.

C. NETWORK PLANE

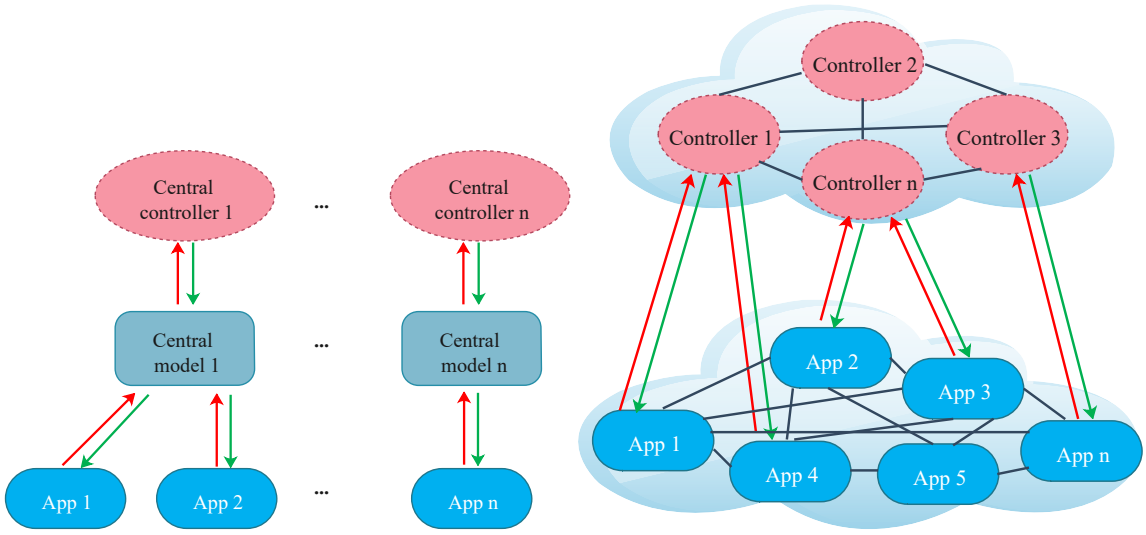
Traditional Internet services use a client-server infrastructure. Each user acts as client and can query data that is stored on a centralized server. Since the centralized control is accountable for database administration, if the authority's security is compromised, the data can be modified or even deleted [64]. In contrast, blockchain is based on a peer-to-peer (P2P) network structure consisting of several decentralized peers. In terms of data integrity, blockchain defines a set of protocols, which verify each participating node in the network when a new transaction is created. Then, the new transaction record is integrated into the block only after the majority of nodes reach a verification consensus. Regarding data storage, blockchain is based on a distributed architecture, where each node has a backup of the whole ledger. This means that if a node is corrupted or in-accessible, the integrity of the database will not be affected. Hence, through the distributed transmission of data, record of transactions and distributed storage, the entire architecture can be defined as decentralized in nature. This decentralized architecture improves the speed, flexibility and security by reorganizing the application service network, and it provides for a more efficient local control and execution capability of a service [65] (Figure 7).

D. APPLICATION PLANE

Many Internet applications can be generally considered as centralized applications that focus processing in one host or in a cluster of coupled computers in a single location. For instance, the purchase process from EBay website can go through PayPal. PayPal is a typical centralized application that concentrates all transactions between the seller and buyer. If PayPal's data-centre or cluster is compromised, its transactional history and balances can no longer be trusted leading to further service disruption to those that rely on PayPal. Decentralized applications (Dapps) differ from centralized applications and are a type of software program on the Internet that are designed in a way that they are not being controlled by any single entity. In order to have an ideal service or blockchain application, there should be no human intervention in the operation which leads the formation of an autonomous organization that is decentralized. The autonomy can help to share the profit and the cost into the blocks [66]. There are noticeable common features of Dapps which are completely hosted by peer-to-peer blockchain system:

- Applications must be completely open-source with no entity controlling the majority of its tokens.
- The application's data and records of operation must be cryptographically stored in a publicly-accessible distributed manner. In this way, it can avoid any central points of failure.
- The application must use a cryptographic token - this is required for accessing the application and any contributions should be rewarded with the application's tokens.
- The application will reward contributors in the community according to a proof of value concept which is predefined by standard cryptographic techniques.

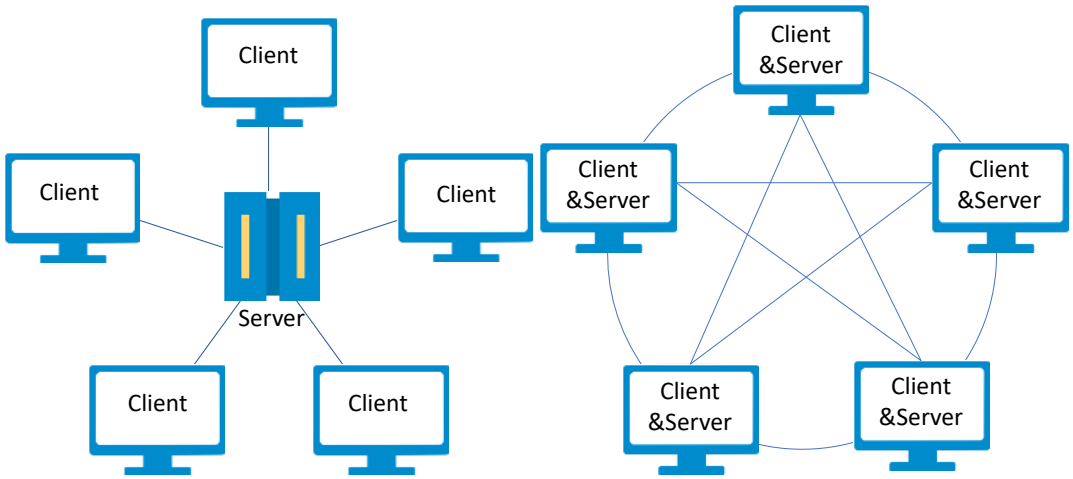
Figure 8 illustrates the differences of centralized and decentralized application plane. Table 3 lists the comparisons between traditional Internet services and blockchain-based Internet services.



(a) Centralized or local centralized control

(b) Decentralized control

FIGURE 6: Comparison between control plane



(a) Centralized network

(b) Decentralized network

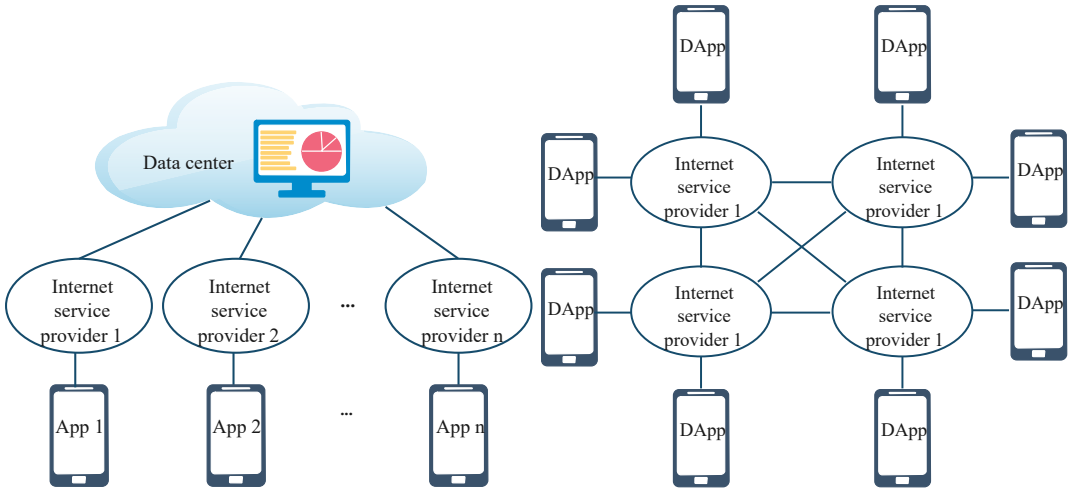
FIGURE 7: Comparison between network plane

TABLE 3: The comparisons between traditional Internet service and blockchain-based Internet service

Topic	Traditional Internet Service	Blockchain-based Internet Service
Data plane	Tradition database structure such as table, b-tree, vector, etc.	Chain data structure with cryptographic methods such as hash, asymmetric encryption, etc.
Control plane	Totally or locally centralized control mechanism.	Decentralized control mechanism.
Network plane	Client/Server network through centralized management.	P2P network through distributed recording, transmission and storage.
Application plane	many large corporate-based Internet entities.	user-centric.

E. BLOCKCHAIN-BASED INTERNET SERVICE ARCHITECTURE IN DEMAND

Based on the above discussion, we can see that contemporary Internet service architectures are showing an inability to efficiently respond to the increasing challenges in many aspects, especially in terms of service security and privacy. We explained the main reasons why blockchain technology can



(a) Centralized Application (b) Decentralized Application

FIGURE 8: Comparison between application plane

improve the security of traditional centralized-based Internet service.

End-data monopolies. While a data monopoly provides an appropriate business for tech giants, from a user's perspective, it is not fair that this data can be freely obtained from end-users and then monetized. [67].

End-surveillance on the Internet. The private data and activity of users is monitored and collected by various services, typically without the consent or knowledge of the user. This is at the expense of a user owning and controlling their identity and security.

Permissionless innovation is reintroduced to the Internet. We need to build an open or public application service network instead of private or proprietary services. Then, regardless of where you are and which service or application you use, interoperability and sharing of information is facilitated and transparent.

In summary, the blockchain-based Internet service architecture is to build a decentralized structure with distributed data verification on which modern internet services can run. The innovation of blockchain-based Internet service architecture is the database technology serving as "the chain of blocks linked using cryptography", which is to provide constant and security connectivity for dynamic network. In addition, the consensus and incentive mechanism of the blockchain will also provide fairness, trustworthy and scalability to upper-layer applications.

V. BLOCKCHAIN-BASED INTERNET SERVICE ARCHITECTURE

In the previous section, the key concepts of blockchain through different planes in the Internet services were discussed which try to address the issues of security of the information maintained by the network. This section will describe how blockchain technology can be built into a layered Internet service architecture and a case study for blockchain-based IoT for neuro-informatics was proposed to explain the feasibility of the proposed architecture before implementation.

A. VISION OF BUILDING BLOCKCHAIN-BASED INTERNET SERVICE ARCHITECTURE

This section presents a totally decentralized, multi-tier Internet service architecture for characterizing and standardizing the typical features and main components of blockchain and briefly describes the underlying structure of each plane. As shown in Figure 9, blockchain-based Internet services can run on a fully peer-to-peer (P2P) basis. Each node in the network can act as both client and server and compared to current services, clients do not reply on a central server which thereby facilitates interaction. The new architecture is a web of connected nodes which make up the network itself. These nodes communicate with each other to maintain, measure and update the new entries in the database. All nodes work together to guarantee they reach a consensus to provide the network with in-built security.

Data Plane: this plane manages multichain data with related cryptography methods to maintain the blockchain database in an ACID (Atomicity, Consistency, Isolation, Durability) style. The data plane also performs necessary required database actions such as create, insert and update [68]. A basic blockchain selects a peer based on the winner of a consensus competition of block hash and it will be authorized to create a new block and add it to the chain structure, encapsulating all transaction data with a specific timestamp generated over the Internet between nodes. For the design of multi-chain databases, the storage structure, data man-

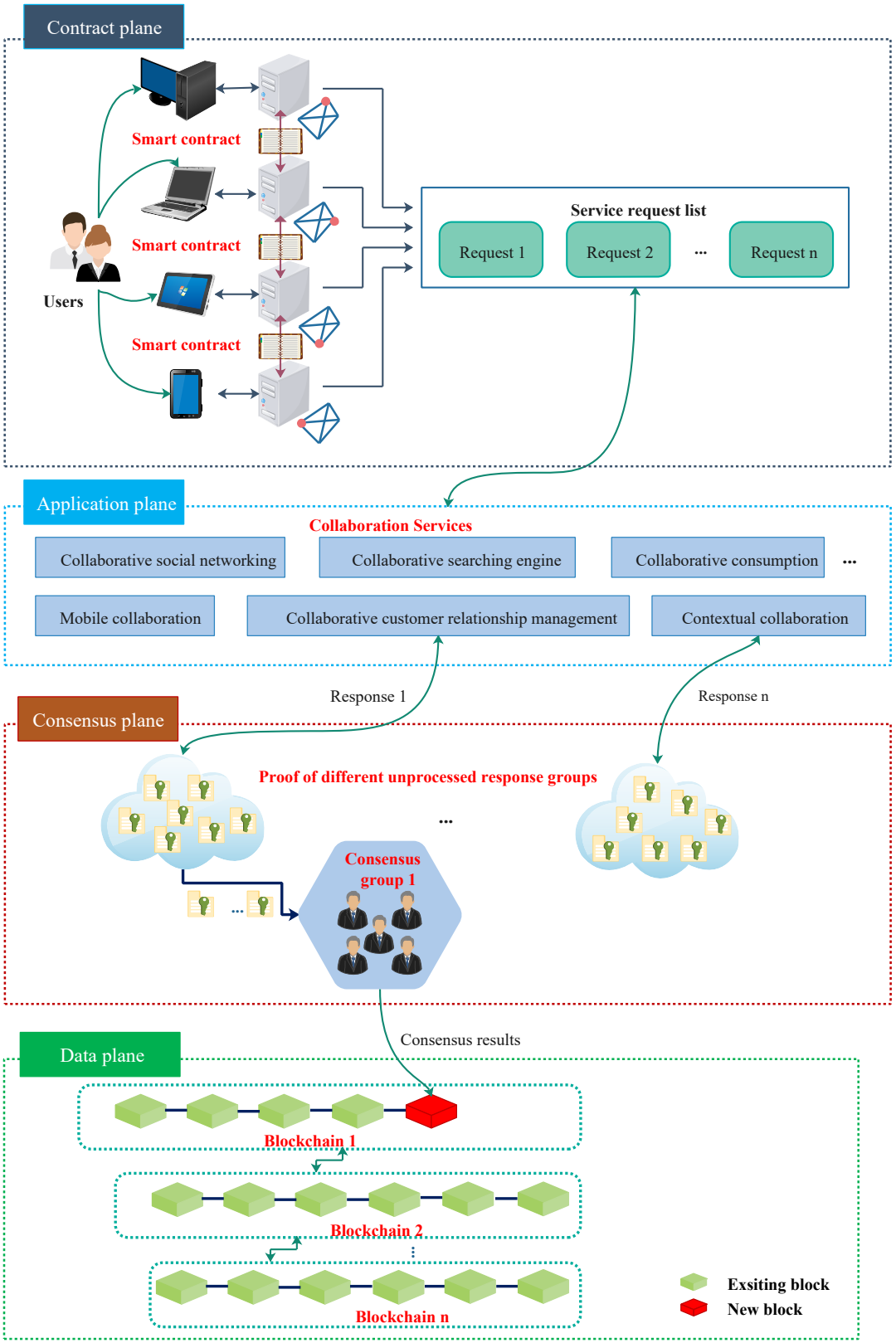


FIGURE 9: Blockchain-based Internet service architecture.

agement, verification mechanism and cross-chain anchoring method are four key components. The Merkle tree and block hash are used to secure verification of content in a large dataset, and help to verify the content and consistency of the data while block hash combined with timestamp makes block chain manipulation harder for an adversary. The traceability of the blockchain data is also enabled [33]. Another aspect in this plane is the anchoring between multichains, with each multichain blockchain having a set of blockchain parameters determining the chain's behaviour. Different blockchains can also use predefined proofs such as Simplified Payment Verification (SPV) [69] to ensure data security and non-tampered. In this way, data can be transmitted between different blockchains, which engenders more extensive application prospects.

Based on the decentralized multichain structure, peers are equally privileged without central administrators or hierarchical entities and can be considered as full user-centric and light-weight peers. Any new datum or block created by one peer will be broadcast to all monitoring nodes in the network. Every node stores all blockchain data, which can be easily synchronized and maintained in the event of the node's failure. In this way, massive amounts of data can be shared amongst completely decentralized Internet entities.

Consensus Plane: this plane packages all consensus algorithms for all Internet peers in the network. These algorithms enable participants to agree on the contents of the blockchain in a distributed and trustless manner. Essentially, a consensus algorithm is used for Internet tasks that can be crowd-sourced. Current consensus algorithms are relatively slow to converge and do not support the satisfactory processing and confirmation speeds required for instant services. Therefore, there is a need to design a reasonable crowd-sourcing mechanism with an incentive capability that enables rewards for each peer across the Internet while ensuring data security [70]. Specifically, it is related to intrachain proof and interchain proof, and the overall consensus service is based on the dynamic collaborations between different service providers. Since the transaction verification is the key problems of consensus process, it is better to select verification nodes dynamically rather than the whole nodes. This can greatly increase the cost of malign peers and reduce the communication delay in the consensus process, thus the designing of consensus algorithms could considered the adjustment of workload (such as service transaction volume+ transaction age) to determine the difficulty of mining nodes and the consensus representative selection.

Application Plane: this plane is commonly accessed to provide Internet services. This acts as an interface between users and the underlying planes, where actual applications are defined such as applied data mining, machining learning, intelligent assistants and other Internet applications. Traditional applications follow a centralized client-server model that directly controls the flow of information from a single centre. All individual clients are totally dependent on centralized services such as Google, Facebook, Amazon and other mainstream services [69] to send requests and receive responses. A decentralized application plane allows different types of applications that use a point-to-point communication model. Designing application is mainly composed with three main modules: a construction module which designs the internetworking mode between multiple service providers to ensure scalable data storage and secure access control, an authority module which is responsible for the permissions related to the contents or contributions of each participants, and a transaction module which is responsible for exchanging the information or value between nodes. Section V-C presents a detailed case study of a blockchain-based IoT for neuro-informatics application.

Contract Plane: this plane encapsulates various scripts, algorithms and smart contracts. Users can define self-request, self-verifying, self-executing and self-response rules via a personalized smart contract. The contract plane provides essential services to the decentralized application plane as well as the control plane, making them programmatic smart properties. For instance, when executing a web service using the HTTP protocol, the contract plane will self-execute and return the corresponding HTTP responses to predefined HTTP requests without any intervention from a third-party. Each response needs to satisfy the consensus algorithms deployed in the consensus plane. After response verification, the new response can be updated in the data plane. The key points to design smart contracts are transaction processing, storage mechanism and complete status identification. The transactions mainly include request and response messages between service providers and users, and when these transactions are transferred into smart contracts, the status identification will be triggered and updated. If the predefined conditions (such as agreed time and event) are satisfied, then smart contracts are executed to guarantee all the chains run the deployed functions automatically.

B. TECHNICAL SUPERIORITY

The integration of blockchain in the internet service architecture could solve many problems that the current architectures face. The role of a blockchain-based mechanism for internet services is detailed by the following aspects:

1) **Improve data security for personal content storage**: personal information is very important for each customer during service interaction. Thus, these contents should be clearly identified and data integrity should be ensured. Blockchain can provide a reliable peer-to-peer communication with security and traceable measures over a untrusted network.
2) **Provide a reliable incentive scheme based on consensus mechanism**: incentives are what encourage communities of participants to cooperate and create the value that ensure the success of internet services. Another advantage of this integration is the possibility to make incentive trusted decisions since the blockchain can ensure that all participants of a decentralized network share identical contents and get consensus. This

assurance can allow the system to reach an agreement over the whole network and to have global collaboration between the different entities.

3) **Provide scalability to support multiple internet services**: sharing of multiple internet services is related to multiple corporations, and the exchange of value will also involve multiple accounts. Blockchain can support complex transactions by simply using smart contracts which have excellent scriptable programmability, and this would increase the fundamental baseline extensibility needed to support different types of internet services.

C. A CASE STUDY

One of the most popular applications evolving in Internet services is IoT for neuro-informatics, which is used to manage and process various biomedical signals and human health information to support lab-based learning and modeling. Traditional IoT solutions for neuro-informatics is always based on browser-server or client-server architectures, with all functional modules deployed on the central server. Although this model is efficient and easy to maintain, it has some high security risks, such as single point of failure, denial of service attack (DOS), human privacy concerns and so on. In view of this, we can integrate a blockchain-based Internet service architecture into the signal processing and control to achieve a trustworthy neuro-informatics system. The system architecture is shown in Figure 10, which describes the entire blockchain-based architecture.

1) Block data structure

The case data is verified and stored in the blockchain database maintained by each peer in the network. When a web user sends a request for neuro information acquisition or creation into the system, the request signed by user's private key will be considered as unprocessed transaction stored in the transaction pool. After transaction verification, it will be added into a new block. Figure 11 presents the detailed block data structure used for neuro information maintenance.

2) Consensus logic

The consensus algorithm used in data verification should be fast and efficient. In this way, we can verify the new signal information submitted by each IoT device as soon as possible, and also evaluate the contributions of verified information. The whole process flow is shown in Figure 12. The legal transaction will be packed into a new block and broadcast to all selected consensus nodes. Then we need to decide whether the new block has agreement to be be added to the verification chain that is considered to be the longest chain.

During the consensus process, each consensus node needs to follow the criteria as detailed below to verify each unprocessed transaction.

- The data structure of transaction must be correct.

TABLE 4: Key functions defined in smart contract

Function name	Parameters and description
Accept Inference	Receive the user request, and represent input data to json syntax with semantic understanding.
Confirm Inference	Confirm if the input data is satisfied with predefined logical rules.
Drop Inference	Cancel the submitted data, and remove it from transaction pool.

- The input conditions and output inferences fields cannot be empty, and conform with defined size.
- The hash value of inference cannot be 0 or -1.
- The correspondence between conditions and inference should be satisfied with the rules defined in the smart contract.
- If the new transaction already exists in the transaction pool, abandon it.
- The signature of transaction must be legal.
- The size of transaction conforms with the definition.

3) Smart contract

The smart contracts deployed in the blockchain-based IoT for neuro-informatics system include the following interface functions as shown in Table 4, and all the smart contracts will be deployed onto the Ethereum platform. Each web user interconnected into a P2P network must call these functions to implement different operations. Furthermore, various other functional modules (such as account manager, feedback manager, etc.) can be also designed and integrated into a blockchain-based system through smart contracts.

The structure of a smart contract is designed as below:

$$
\begin{aligned}
&struct\ Conclusion\ \{\\
&\quad unit\ typeID\\
&\quad bytes\ signalInfo\\
&\quad bytes\ healthInfo\\
&\quad address\ sender\\
&\quad address\ receiver\\
&\quad unit\ state\\
&\}
\end{aligned}
$$

The structure consists of $typeID$ which is used to label the type of brain signals; $signalInfo$ and $healthInfo$ are input signal information and corresponding health report; $sender$ and $receiver$ are sender address and receiver address respectively; $state$ describes the current state of this transaction.

The main algorithm (Algorithm 1) takes a neuro-informatics application with brain signals through the blockchain-based creation and verification mechanisms.The main loop has initial global variables T, K and C. T is a set of unprocessed transactions stored in a transaction pool. K is a set of public keys, K_i is used to encrypt a corresponding transaction T_i. C is the original blockchain database with a defined genesis block. Each unprocessed transaction with its corresponding public key will be verified by smart contracts and packaged into a new block. This new block will be broadcast to all the peers in the network. Then the consensus nodes will be selected based on the dynamic verification

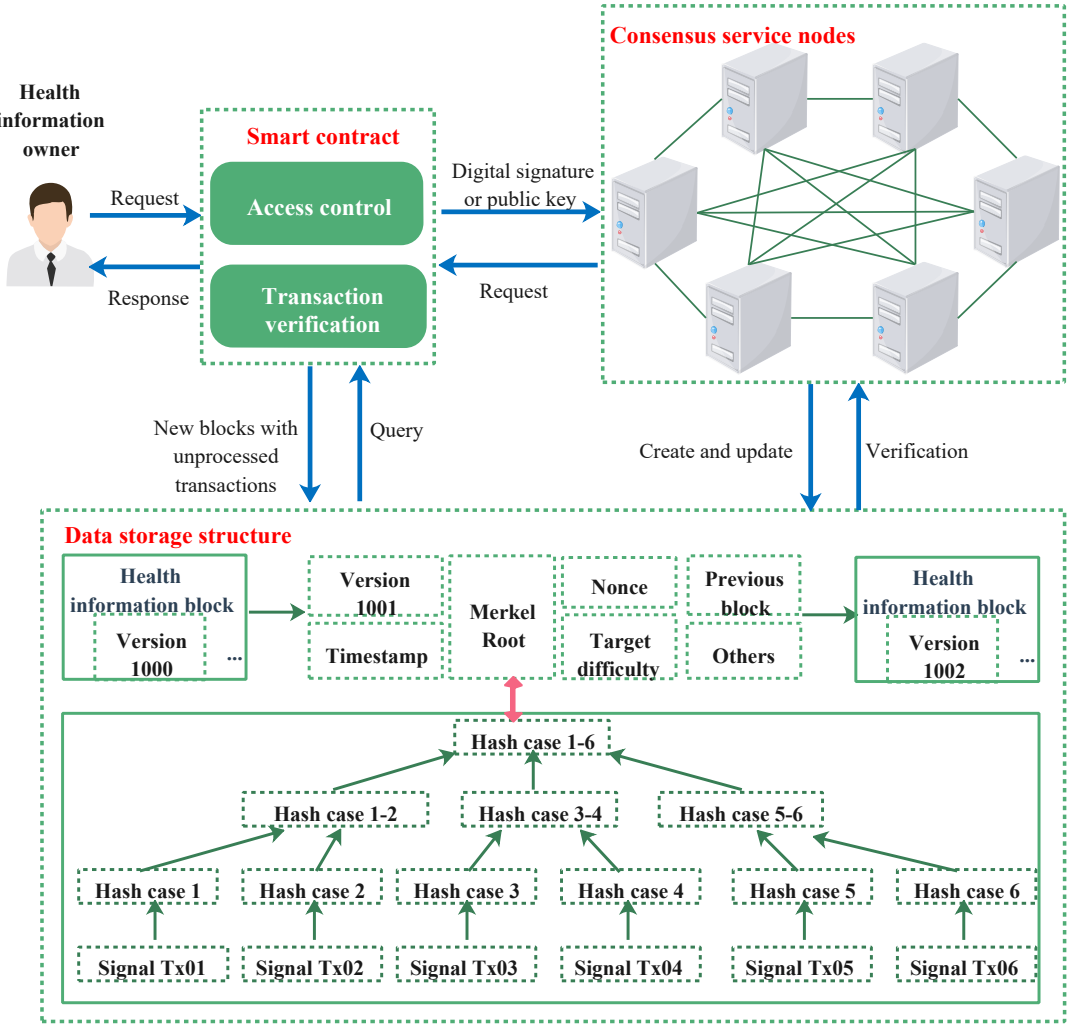


FIGURE 10: System architecture of blockchain-based IoT for neuro-informatics.

scheme to find a solution for the verification job. Finally, the new block is verified and updated in the blockchain database. $assignNewTransaction()$ is used to receive user requests, transform these requests into the defined transaction format and add them into $transactionPool$. $transactionValid()$ is used to verify a new transaction. $addBlock()$ creates a new block and broadcasts it to the whole network. $updateBlock()$ updates the original blockchain database when the new block reaches successful consensus by selected consensus nodes.

VI. KEY BLOCKCHAIN-BASED INTERNET SERVICE REQUIREMENTS

As stated in previous section, a layered security Internet service architecture can be built through blockchain technology. However, from a research viewpoint, there are several key technical requirements that need to be addressed for blockchain-based Internet services to reach their full potential. The key requirements are explored and summarized in the building of blockchain-based Internet service architecture as shown in Figure 13.

A. DATABASE SECURITY

The blockchain database has shown a proven robustness to data security and integrity in cryptocurrencies, which not only supports a single blockchain, but also provides sidechains as well as multichains used by all participants through secured cryptographic protocols [70]–[72]. The advent of decentralized databases built on blockchain technol-

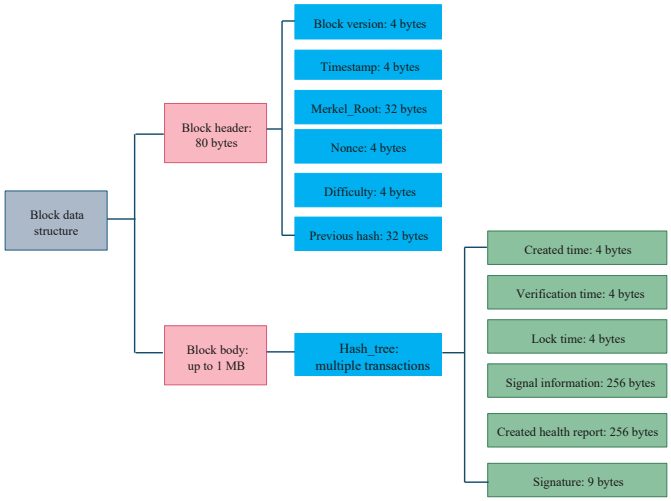


FIGURE 11: Block data structure

Algorithm 1 :BlockchainLoop

Data: The set of transactions: T=$T_1, T_2, \ldots, T_n$; The set of public-key corresponding to each transaction: K=$K_1, K_2, \ldots, K_n$; Blockchain with genesis block: C; verification difficulty level: L;

Result: blockchain-based neuro-informatics case:B

$transactionPool$ = assignNewTransaction(T_i, K_i)
if $transactionPool \neq$ null **then**
  for each $T_i \in transactionPool$ **do**
    broadcast T_i to each node in the network;
    calculate Merkle-tree with the corresponding T_i and K_i;
    P_j =transactionValid(T_i, K_i,C,L): get the peer ID who competed to be the first;
    $block_{new}$=addBlock(T_i, K_i):add new block and broadcast to the whole network;
    while True **do**
      B=UpdateBlock($block_{new}$,B);
    end while
  end for
  return B
end if

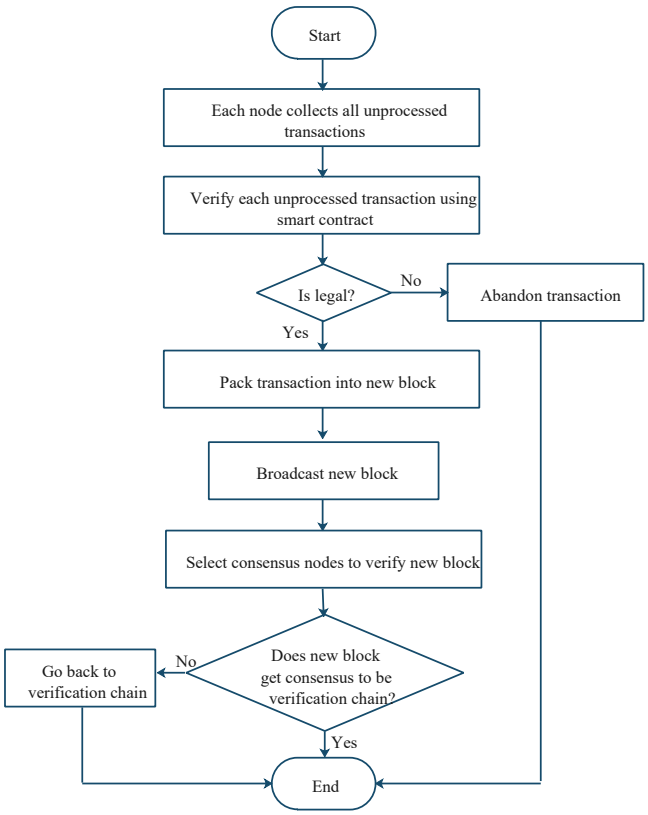


FIGURE 12: The flow diagram of general consensus process

ogy creates new requirements, as they will exchange massive volumes of data that need to be stored and managed. The following requirements are investigated:

Storage security: decentralized storage needs to meet the demands of storing high volumes of data across the Internet. Blockchain's linked storage structure allows for one chain on the whole network. All coincident transactions are kept in the same block based on a consensus algorithm, and in the case of Bitcoin, a block is created every few minutes. However with the exponential increase of technology usage, from the point of technical implementation, there are three main methods: sidechains, sharding and Directed Acyclic Graph (DAG). Rootstock [73], Alpha [74] and Liquid [75] are typical examples of using sidechains, which allow tokens and other digital assets from one blockchain to be securely used in another separate blockchain and then be moved back to the original blockchain if needed. Zilliqa [76], Rchain [77] and Quarkchain [78] use the sharding mechanism to scale up, which divides the super blockchain network into several sub-chain networks (each sub-chain network we call a shard) consisting of part of peers. IoT Chain (ITC) [79], Byteball [80] and IOTA [81] are the most applicable examples of the DAG structure. These new implementations are scalable, light-weight and decentralized, making them more suitable for large-scale networks and they also support different types of transactions being recorded on different chains simultaneously. The storage potential of enhancing the single-chain blockchain storage into sidechains, sharding and DAG [81] structures can be seen in Figure 14.

Data management: in a traditional database, a client can perform four basic functions on data: Create, Read, Update, and Delete (CRUD commands). Since blockchain

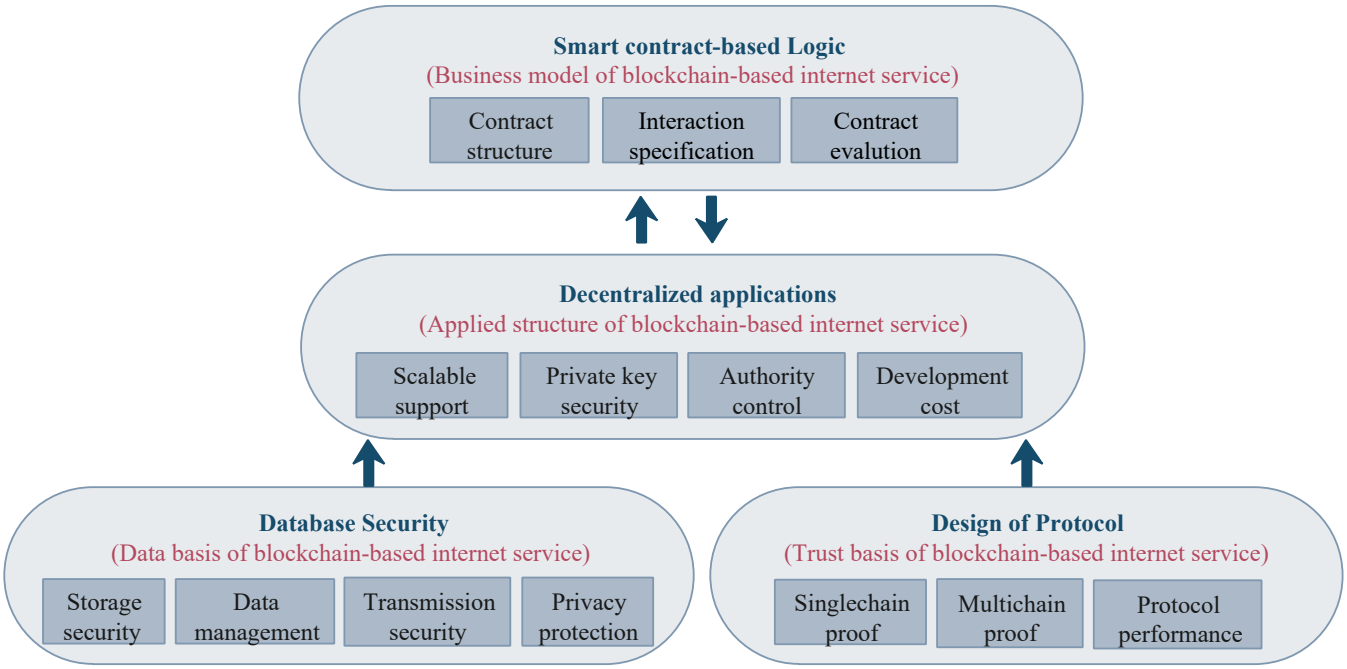


FIGURE 13: Key technical requirements in Blockchain-based Internet service.

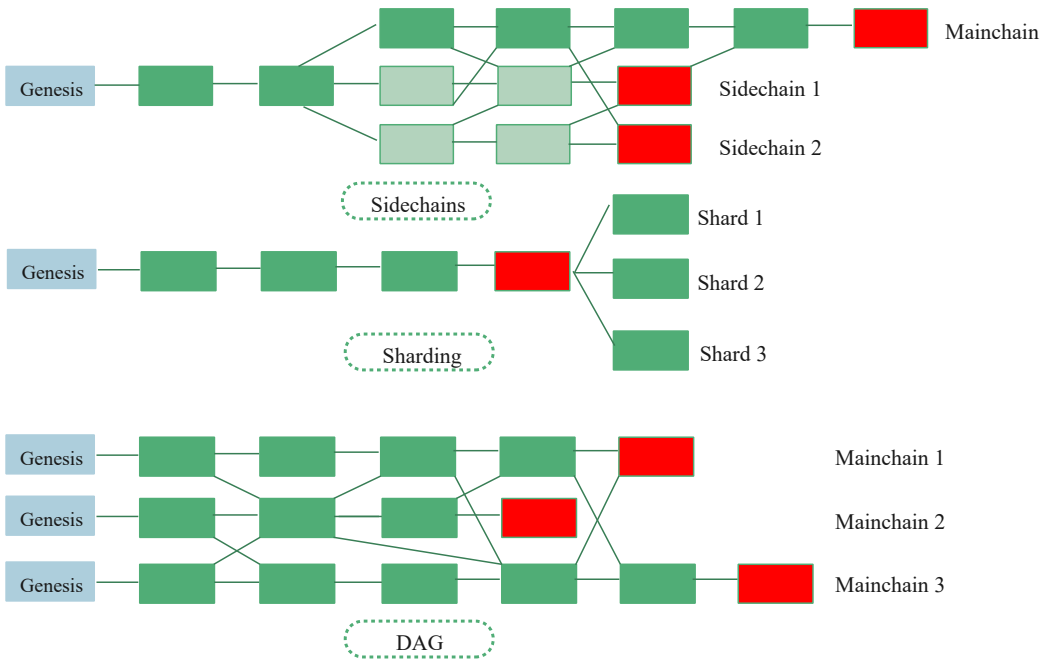


FIGURE 14: Storage structure between sidechains, sharding and DAG.

TABLE 5: Key features of Typical Hash functions used in blockchain database

Algorithm	security	Arithmetic speed	Output length
MD5	Low	Fast	128
SHA1	Medium	Medium	160
SHA2	High	Lower than SHA1	256
SHA3	High	Lower than SHA1	256

data is permanently stored and is immutable, the operations associated with blockchain are creating and reading, which means that there is no native deletion or update. Reading can query and retrieve existing data from blocks indexed by their hash value with some other attributes. Writing is delayed by waiting for block creation, and an additional mechanism is required to implement the concept of deletion and update (for example, flagging transactions as stale). A public blockchain database is a read-uncontrolled as well as write-uncontrolled database, which means any client can read a block in the existing chains, and write a new block into the chains (subject to consensus) [64]. However, with the existing technology, write operations are slow due to the transaction confirmation mechanisms which take several minutes to complete. Therefore, faster and more intelligent methods are required to maintain data with blockchain-based Internet service databases.

Transmission security: blockchain databases use advanced cryptographic techniques to ensure data transmission security. It involves at least two levels of security protection. Firstly, the global states are protected by a Merkel tree where the root hash is stored in the block header. Furthermore, the block history is also protected through a chain of cryptographic hash pointers [41], [82]. Hashing is also used in encryption of transactions. There are several typical cryptographic algorithms, such as MD5, SHA1/SHA2 and SM3 [41]. As indicated by Table 5, hash functions like MD5 and SHA1 are officially insecure, and SHA2 and SHA3 are the most popular hash functions used in blockchain databases. The Merkel tree helps achieve rapid and secured transaction verification, while the hashing and time-stamp enable integrity and traceability during transmission between peers in the network.

Privacy protection: as previously discussed, blockchain-based data storage and transmission is transparent, and users can use a digital signature to protect their privacy through the use of a public and private key pair. Public keys can be shared with everyone while private keys are kept secret. Either of the keys can be used for message encryption while the other key is used for message decryption. RSA (Rivest Shamir Adleman), ECC (Elliptic-curve cryptography) and SM2 (SuperMemo) are among the most common asymmetric encryption methods used in blockchain systems [22]. Table 6 lists the key features of these digital signature methods. These asymmetric encryption methods should be further strengthened in a huge number of Internet services with multichain interaction.

TABLE 6: Key features of typical digital signature methods used in blockchain database

Algorithm	security	Maturity	Arithmetic speed	Resource consumption
RSA	Low	High	Slow	High
ECC	High	High	Medium	Medium
SM2	High	High	Medium	Medium

B. PROTOCOL DESIGN

Since a blockchain database supports both single chain and multichain structures, there is a need to design and apply different protocols to ensure trust is inherent. The following requirements are discussed for consensus protocols used for intrachain and interchain communication.

Intrachain proof protocol: The consensus protocol for single blockchains is used to achieve agreement on a single data value among distributed processes or systems. The most common consensus protocols used for single blockchains include: PoW, PoS, DPoS, Paxos, PBFT and DBFT.

- PoW: Proof of Work is one of the first utilized consensus protocols that is computationally based, requiring miners to find the solution to a puzzle. Several cryptocurrencies utilize a variant of this protocol [4], [83], [84]. It is a data item that is time-consuming to produce but easy to verify by others which satisfies specific necessities [85], [86].
- PoS: Proof of Stake is a proposal that determines who will add the next block into the blockchain based on how much stake a miner has in the network [87] - in other words, mining is done by stakeholders in the ecosystem who have the resilient motivations to be decent stewards of the system [88] [89].
- DPos: Delegated Proof of Stake is a newer consensus structure where users select some delegate nodes which confirm the validity of a block [90]. The network performance, resource consumption and fault tolerance of DPoS are similar with PoS [91].
- Paxos: Paxos is a consensus protocol based on a leader role. A leader node has absolute authority and it allows other nodes to participate in supervision. All the nodes in the network have a general access mechanism. However, during the process of selection, malignant nodes cannot be allowed. Hence, fault tolerance is not available in Paxos [92], [93].
- PBFT: similar to Paxos, Practical Byzantine Fault Tolerance (PBFT) uses permissive voting, with the principle that the minority is subordinate to the majority [94]. In contrast, the consensus algorithm allows a 33.3% fault tolerance [95].
- DBFT: Delegated Byzantine Fault Tolerance (DBFT) is similar to PBFT where the main difference is based on including a leader driver with delegation to improve the efficiency of data processing [96].

Interchain proof protocol: an efficient and secure communication protocol over the Internet and is the most in-demand technology for blockchain-based Internet services. As multichain blockchains can allow for large storage capacities, together with higher data integrity and transparency, multichain is a suitable solution [97]. Among multichain technologies, cross-chain communication is one of the key issues. Key types of cross-chain technologies are outlined below.

- Notary schemes: these are the most common schemes used for routing payments across diverse digital ledgers through the separation of receivers and senders from the risk of intermediary failures. This protocol is invoked by hosts over higher-level protocol modules in an interledger environment [98]. Figure 15 shows the basic layered structure for a notary scheme.
  Referring to the Figure 15, connectors act as a notary to build the communication between interledgers deployed in the different blockchain platforms. For instance, in Ripple, the Notary module would call on a local ledger module which would create a Ripple transaction with the interledger packet attached to the Ripple Consensus algorithm [99]. Then, the Ripple address would be derived from the interledger address that might be connected to other ledgers via the local ledger interface.
- Relays: this technology uses building blocks that allow contracts to securely verify blockchain transactions without any intermediaries. They can also act as a smart contract that stores block headers. Then, these block headers are being used to build a mini-version of the blockchain [100] (refer to figure 16). Bitcoin also uses this method to achieve Simplified Payment Verification (SPV) light wallets. The work flow is divided into three steps:
  i) relayers constantly submit blockchain headers;
  ii) transactions are submitted to be verified;
  iii) verified transactions will be replayed to the smart contract.
  Relays belong to the early stage of cross-chain communication technology. They combine two different blockchains with a defined smart contract. The applied trust model is similar to the single blockchain and chains do not fail or suffer from 51% consensus attacks. A typical example implemented by relays is BTC (Bitcoin) relays that connects Ethereum and Bitcoin using a smart contract [101], where clients can pay for Ethereum usage via Bitcoin payment. Another example is RootStock (RSK) [102] which is a smart-contract platform that incorporates a Turing Complete Virtual Machine (TCVM) with Bitcoin. Relays also provide some network enhancements such as better scalability and faster transaction features which will also enable new usage scenarios.
- Hash-locking: this is a key technology of the lightning network. Single blockchain has limitations such as the transaction rate (of the order of a few transactions per second in the whole network), and the verification of new blocks which require relatively long time durations by consensus nodes [103], [104]. Both these problems bring difficulties when extending the application capabilities of blockchain-based Internet services. Hash locking provides an extended channel that restricts the spending of an output until a specified piece of data is publicly revealed. Hash-locking has the useful property that once any hash-lock is opened publicly, any other hash-lock secured using the same key can also be opened [105] (Figure 17).
  For instance, if two users (Alice and Bob) make a Hashed Timelock Contract (HTLC) protocol before communication, the blockchain system will lock the lightning network between them (Alice and Bob), until Bob can return a hash value within 3 days. If this hash value is correct, Alice can transfer money to Bob immediately. Therefore, if two peers pre-set a hash-lock contract, then they can achieve instant and multiple transactions between each other. However, although hash-locking can realize the exchange of digital assets, it cannot support cross-chain contracts. Hence, hash-locking applications are limited.
- Distributed private key control: this is a hybrid protocol that combines some single blockchain protocols together. Private assets can be mapped to a public blockchain through a distributed private key and control technology, which can realize lock-in and unlocked modes. A lock-in model is the process focusing on retaining the control and mapping of assets, while unlocked is the reverse operation of the lock-in process, allowing control power to be returned to the owner. Figure 18 shows the basic function of distributed private key control [106].
  As an example, fusion [104] is a popular distributed private key control platform. Fusion ensures that nobody can access the complete private key, making sure that no single node can obtain the control of the completely digital ledger. In addition, Fusion is based on the Hierarchical Hybrid Consensus Mechanism (HHCM) combining the PoS and PoW blockchain protocols, and it utilizes parallel computing to group nodes, thereby achieving a favourable balance of efficiency and safety.
- Notary schemes + Relays: this key type combines both technologies. Relays are first used to build an efficient communication channel and Notary schemes aim to achieve instant transactions between peers in the network. One typical instance is Ether Universe [107] (as shown in Figure 19). Ether Universe connects different blockchain networks such as Ethereum, Bitcoin, EOS and others via 'connectors' used in Notary schemes and 'verification' used in Relays.
  Ether Universe inherits the advantages of EOS, which can process millions of transactions per second and generate corresponding transaction snapshots at the same

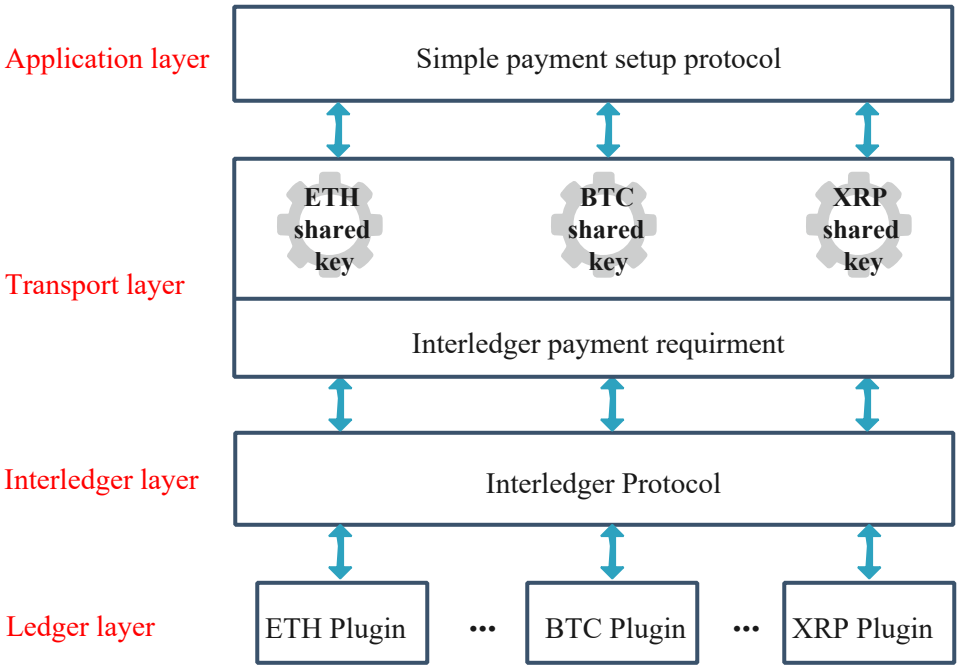


FIGURE 15: The notary scheme layered structure.

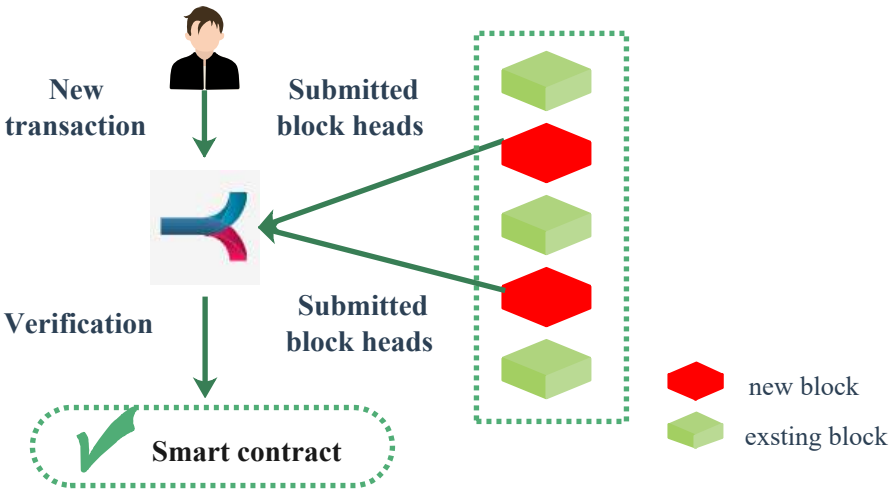


FIGURE 16: Relays design and construction.

time. Ether Universe is a very recent addition to the cross-chain platform which requires further evaluation.

Protocol performance: Consensus algorithms are designed to establish reliability in a network involving multiple unreliable nodes. For the consensus algorithms used in single blockchain, the protocol performance is mainly analyzed based on the average confirmed efficiency, resource consumption and tolerance power [108] presented in Table 7. From Table 7, it can be seen that PoW, PoS and other consensus algorithms are inefficient with associated issues of serious energy consumption. Hence, these algorithms cannot meet the performance requirements of blockchain-based Internet services.

Considering the poor consensus and energy performance of most current intrachain protocols, the design of new intrachain protocol should be satisfied with the following requirements:

1) Dynamic verification: is able to perceive and adjust the mining structure for different networking environments. In addition, dynamic verification also reflects the more efficient usage of computing resources such as CPU load, memory, bandwidth and so on. Hence, the performance of an intrachain protocol should have a stable longer-term decrease use of resources.
2) High-throughput and low delay: high-throughput means the intrachain protocol can process more ver-

TABLE 7: Comparison with different single-chain consensus protocol.

Consensus	PoW	PoS	DPoS	Paxos	PBFT	DBFT
Year	2008	2012	2014	2015	2015	2016
Average confirmation time	about 10 minutes	about 60 seconds	about 3 seconds	$\geq$ 1 second	$\geq$ 1 second	$\geq$ 1 second
CPU usage	High	Medium	Medium	Low	Low	Low
Tolerance power	$\leq 25\%$	$\leq 51\%$	$\leq 51\%$	$\leq 51\%$	$\leq 33.3\%$	$\leq 33.3\%$

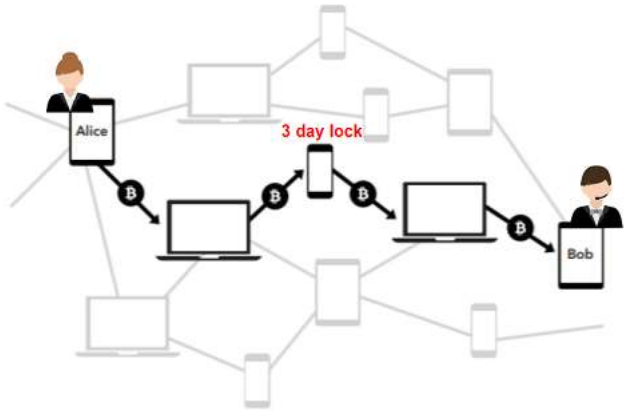


FIGURE 17: Hash-locking transaction example.

ification requests per unit time and the low delay is related to the transaction cost. The intrachain protocol should optimize the user experience and reduce waiting times.

3) Low power consumption: to support large-scale internet services, the design of node selection strategy, grouping verification and node management can be used to reduce power consumption.

Here, the key features between the different cross-chain communication technologies in Table 8 are compared and discussed by the following presented criteria.

- **Trust model:** proof principles used between separated chains.
- **Usable for cross contract:** the difficulty level of smart contracts deployed into multichain structure.
- **Transaction speed:** the transaction processing performance during mining.

Considering the above criteria for an interchain protocol, Notary schemes and Hash-locking have difficulty in support cross-chain smart contracts, and thus, they have poor scalability. Relays have low transaction speed and high delay, which is also not suitable for various internet services. It is necessary to design a hybrid interchain protocol to support concurrent processing of diversified services that satisfy the following requirements.

1) Anchoring between multi-chains to guarantee non-tampering: the transactions between chains should be linked by two-way peg[1] or other similar strategies to ensure reliable transmission and avoid double cost.
2) Efficient verification of cross-chain transactions: a shorter block interval can make transaction verification more efficient, but it may cause increased chain forking that reduces the network availability. Thus, the design of an interchain protocol should consider the trade-off between verification time and the number of forks.
3) Cooperative consensus based on dynamic construction strategy: the consensus nodes selected from different chains are used to build a set of verification nodes. The dynamic construction strategy should be based on the computing power, the credibility of the node and other factors to make sure that the selection of verification nodes is uncontrollable.

[1]two-way peg enables interchangeability of digital assets at a predetermined rate between the two chains.

C. APPLICATION REQUIREMENTS

In this subsection, several key requirements of applications for various Internet services are listed.

Scalable (massive) user support: At present, basic Internet services such as web-based shopping sites like Amazon and Internet email hosts like Gmail have a massive number of user accounts. Therefore, in order to deploy a new Internet service architecture on the basis of the blockchain platform, the architecture has to support massive numbers of users, and avoid the resulting problems related to network performance, while also giving consideration to expandability storage.

Security of private keys: the user experience is an important indicator of Internet services. It is inefficient and possibly insecure to use a haphazard, guessable string as an account or password identifier for each user. In addition, if a user loses their authentication details, there is a need for authentication mechanisms to re-establish the identity. Contemporary systems apply two-factor authentication. However, it is desirable to have a set of security mechanisms to store private keys combining the blockchain platform and application layer [22].

Authority control: data sharing and transparency are very sensitive topics for business services. Simultaneously, as a mutual trust between peers is being built, there is a need to guarantee the privacy of commercially-sensitive information as well as individual privacy, as they are included in the basic philosophy of the blockchain-based service architecture.

Development cost: the convenience and reliability of application development determines the success of blockchain deployment. During the development phase, there is a need to put into consideration the costs of development including the technical, time and labour costs [109].

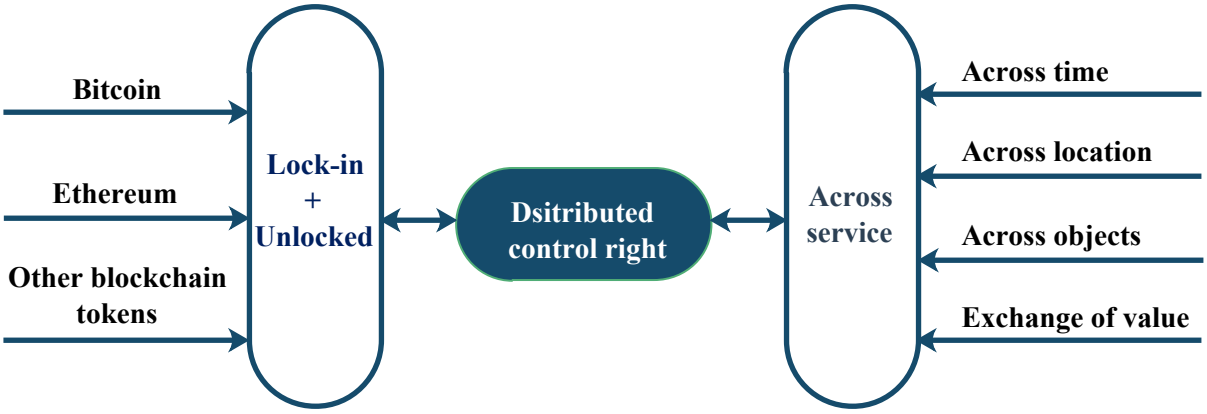


FIGURE 18: Basic structure of distributed private key control.

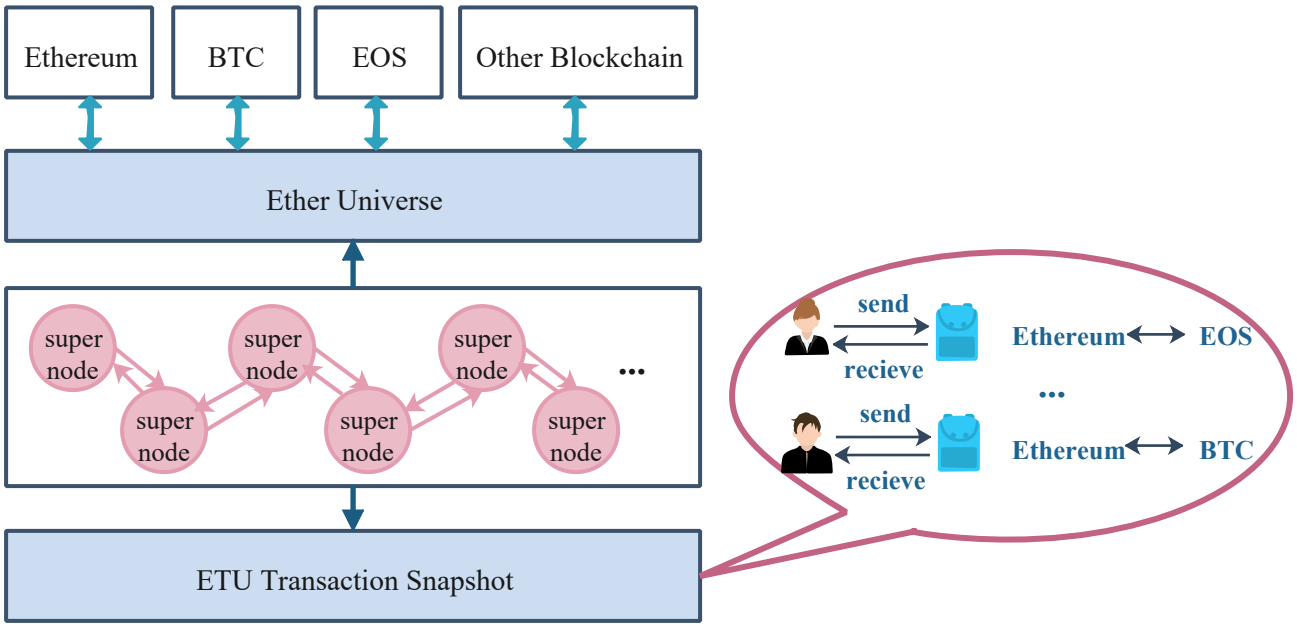


FIGURE 19: Basic structure of Ether Universe combined with notary schemes and relays.

D. CONTRACT REQUIREMENTS

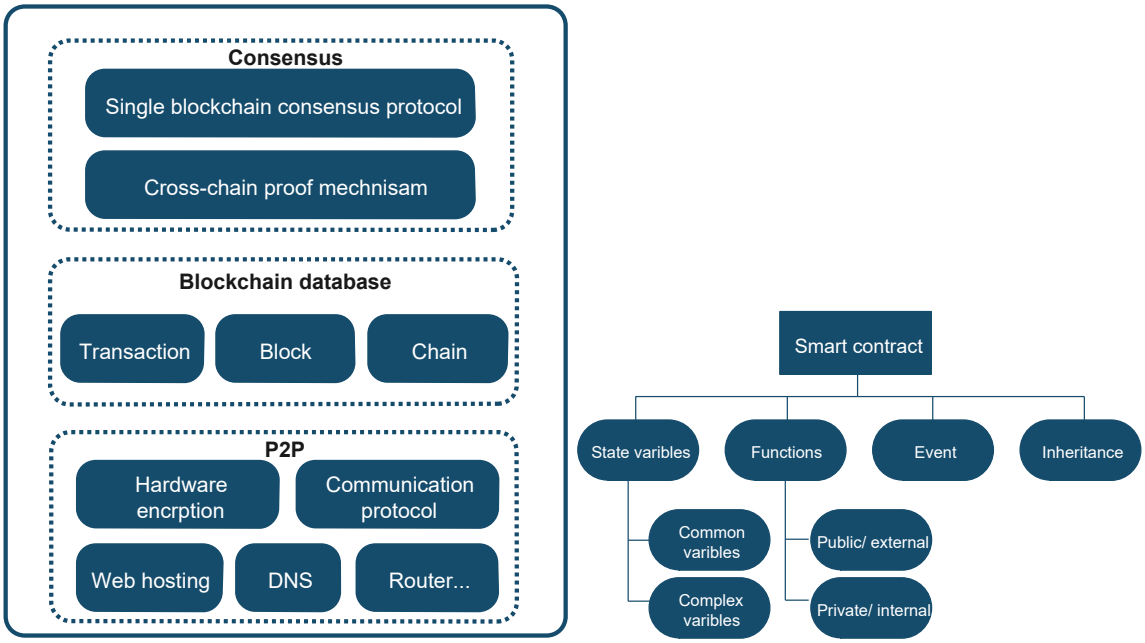
A blockchain-based Internet service architecture provides two levels of contracts: standard and smart contracts. A standard contract is suitable for simple scenarios and is always deployed or encapsulated when the blockchain is initially created (only simple commands are supported). A smart contract aims to solve more complex scenarios and it exposes many API interfaces for arbitrary programming that developers can use to make complex agreements between different nodes [110]. The key requirements of standard and smart contracts have been outlined in Figure 20.

Contract structure: the standard contract can be considered as the cryptography mechanism used inside the blockchain platform as described in previous sections. The standard contract cannot be updated and deleted after being deployed in the blockchain system. On the other hand, the smart contract includes fully-featured scripted programming, made up of a set of rules running on top of a blockchain-based system. The smart contract is also proposed to reduce transactional costs and guarantee a greater degree of security [110]. The main structures used in standard and smart contracts are shown in Figure 20.

Interface specifications: the contract interface should be designed according to the blockchain database model. Operations related to contracts can be classified into two levels: static and dynamic. The static level aims to define the relationship between users and objects. For instance, a standard contract can be used to create an account and declare the owned assets. The dynamic level focuses on operations between users and users or users and objects. For instance, a smart contract can be used to define restrictions with regard to asset transfers, updating account information, access control

TABLE 8: Comparison with different cross-chain consensus protocol.

Types	Notary schemes	Relays	Hash-locking	Distributed private key control	Notary schemes + Relays
Typical Applications	Interledger	BTC, RootStock, Polkadot	HTLC	Fusion	Ether Universe
Trust model	Majority of notaries honest	Chains do not fail or get"51% attacked"	Chains do not fail or get "51% attacked"	Hybrid consensus protocol	Majority of notaries honest + Chains do not fail or get "51% attacked"
Usable for cross-contract	Difficult	Easy	Difficult	Easy	Easy
Transaction speed	Low	Low	High	High	High
Popularity	Launched in 2012, well-known	Launched in 2015, well-known	Launched in 2016, not well-known	Launched in 2017, not well-known	Launched in 2018



(a) Standard contract in blockchain

(b) Smart contract in blockchain

FIGURE 20: Basic components of standard contract and smart contract

and so on. The following specifications should satisfy the contract interface:

i) Embody the principles of interface isolation.

ii) Interface definition and encapsulation should be related to different service fields.

iii) Support a stateless interface call, which is independent of previous operations or previous relationships.

Contract evaluation: although smart contracts are used in many blockchain platforms and are driven by many different types of services, it is necessary to determine the evaluation measures of smart contracts [39]. After understanding the ways to apply smart contracts with detailed insights, there is a need to measure the performances and challenges when they are deployed, such as formal descriptions, contract model verification, consistency tests and so on [111], [112] (Figure 21).

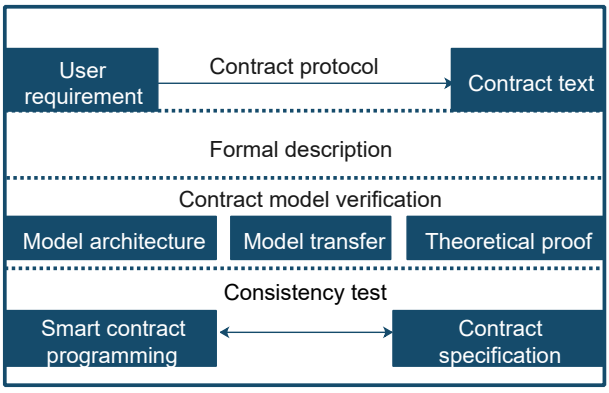


FIGURE 21: Contract evaluation requirements

VII. BLOCKCHAIN-BASED INTERNET TRENDS AND CHALLENGES

In the section VI, different technical requirements for each plane in the internet service architecture were discussed. With these requirements in mind, the researchers envision that blockchain technology will remarkably advance and become the basis for building completely trustworthy Internet service architecture that can tackle privacy and trust issues that are currently encountered with today's Internet services. In this section, the key trends and challenges which blockchain-based Internet development tries to address with respect to the proposed requirements are noted.

A. BLOCKCHAIN TRENDS FOR FUTURE INTERNET SERVICES

Over the past few years, along with the rapid development of the Internet, there are five main Internet technologies which have mainly influenced the early development of blockchain [113] (shown in figure 22), including TCP/IP [114], Routers [115], Web applications [116], P2P [117] and information security technology [118]. Based on these five main influences, blockchain attempts to build a decentralized structure to achieve new applications using cryptographic methods.

It can be seen that the TCP/IP protocol is the basic technology and de facto standard for the transport and networks layers of a layer-based approach for internetworking, but now blockchain technology is one of the new technologies in the associated application layer. Blockchain is the technological imitation of router technology from the network layer to the application layer, that performs traffic routing decision functions required on the Internet. With the development of web applications, two main application structures have emerged: browser/server and client/server model. However, both models are based on centralized or locally centralized controllers to concentrate on Internet services or applications, while blockchain attempts to change them to a decentralized structure. In 2000, a P2P network was proposed to partition tasks or workloads among peers which is the foundation of blockchain technology. Then, following the many information security technologies used for Internet services, blockchain used several cryptographic methods to build a transparency and trustworthy mechanism to support transactions between different peers. Thus there is an inextricable connection between the development of the Internet and blockchain technology.

B. TRENDS IN BLOCKCHAIN- SYSTEMS

Up to now, blockchain technology has been steadily developing from the original Bitcoin protocol for digital currency to the second generation Ethereum platform integrated with smart contracts [119]. Today, we are in the process of building what is unofficially termed blockchain 3.0 and future-generational blockchain 4.0 [120], [121]. In this section, we provide a simple description about how the technology is evolving from its initial form, to become a fully-edged globally distributed system as shown in Figure 23.

Blockchain 1.0 is completely dedicated to the digital currency. The typical platforms that are supported are the mining of Bitcoin and other crypto-currencies such as Litecoin [122], Dogecoin [123] and so on. The consensus algorithm utilizes Proof of Work (PoW) which is only used in the public chain. Blockchain 1.0 guarantees distributed storage, allows data sharing between nodes, and enables transparency in transactional processing [108].

In Blockchain 2.0, some new cryptographic methods such as the Merkle tree were added into the data plane to more efficiently manage the transactions. In addition, apart from PoW and PoS, other consensus algorithms used in public chains, private chains or consortium chains, such as DPos and PBFT, were proposed to reduce the volume of transactions [124] [125]. The most important improvement was the utilization of the smart contract, which automatically executes small computer programs when certain conditions are met [108]. Smart contracts aimed to reduce the cost of verification and execution, while aiding fraud prevention. The most prominent system in this version of blockchain was Ethereum, proposed in 2013. This version allows the formation and transfer of digital assets and other financial applications. The main limitations of Blockchain 1.0 and 2.0 are the energy consumption,volume of transactions and cost [108].

In order to tackle the limitations in blockchain 1.0 and 2.0, a third generation of blockchain platforms was proposed to support different blockchain data structures, proof protocols and the development of various areas rather than financial applications. However, it still has some limitations such as the efficiency of consensus, security of smart contract and interoperability of multichain [126], [127].

With the rise of new industrial technology, known as Industry 4.0, a fourth generation of blockchain platforms is being presented to provide ideal solutions to satisfy business demands. Blockchain 4.0 aims to improve the consensus efficiency, the scalability of blockchain networks, the energy requirements of computation and so on, thereby tailoring blockchain to real, contemporary and future environments.

C. CHALLENGES AND FUTURE

Based on the above discussions, the evolving key requirements which enable blockchain to be able to communicate and interoperate over the Internet, maintain a global and reliable repository of information [126] can be found. However, blockchain is also faced with multiple challenges and research problems that need to be resolved. Generally, three criteria are always used to assess the blockchain technology: decentralization, scalability and consistency. There is a tradeoff among these three characteristics, for example, the applications based on Bitcoin and Ethereum platforms are decentralized and consistent, where every full node stores all the data without centralized control, but they suffer in the lack of true scalability (which is exhibited by the duration of

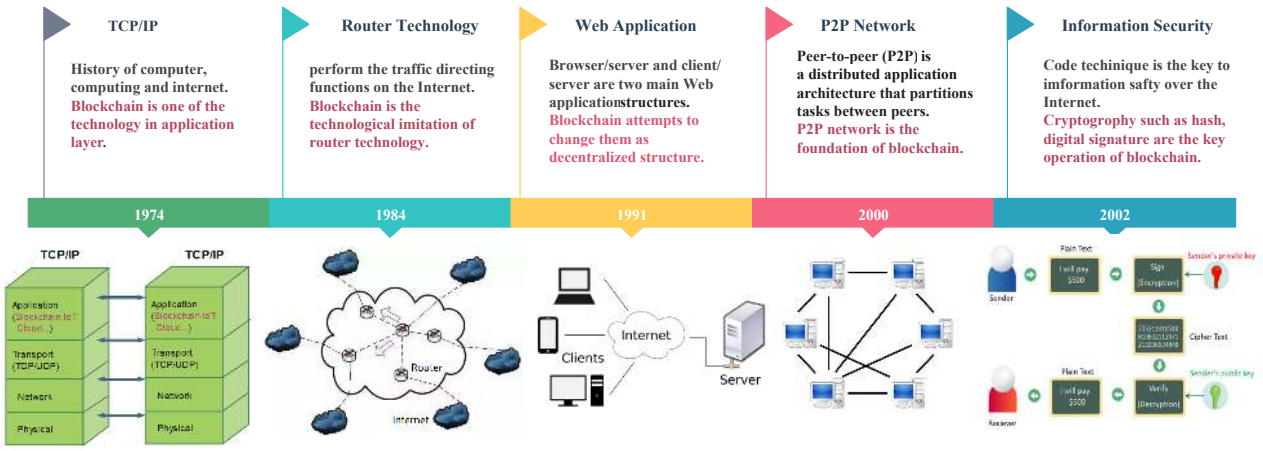


FIGURE 22: 'Development of Internet' VS 'Development of Blockchain'.

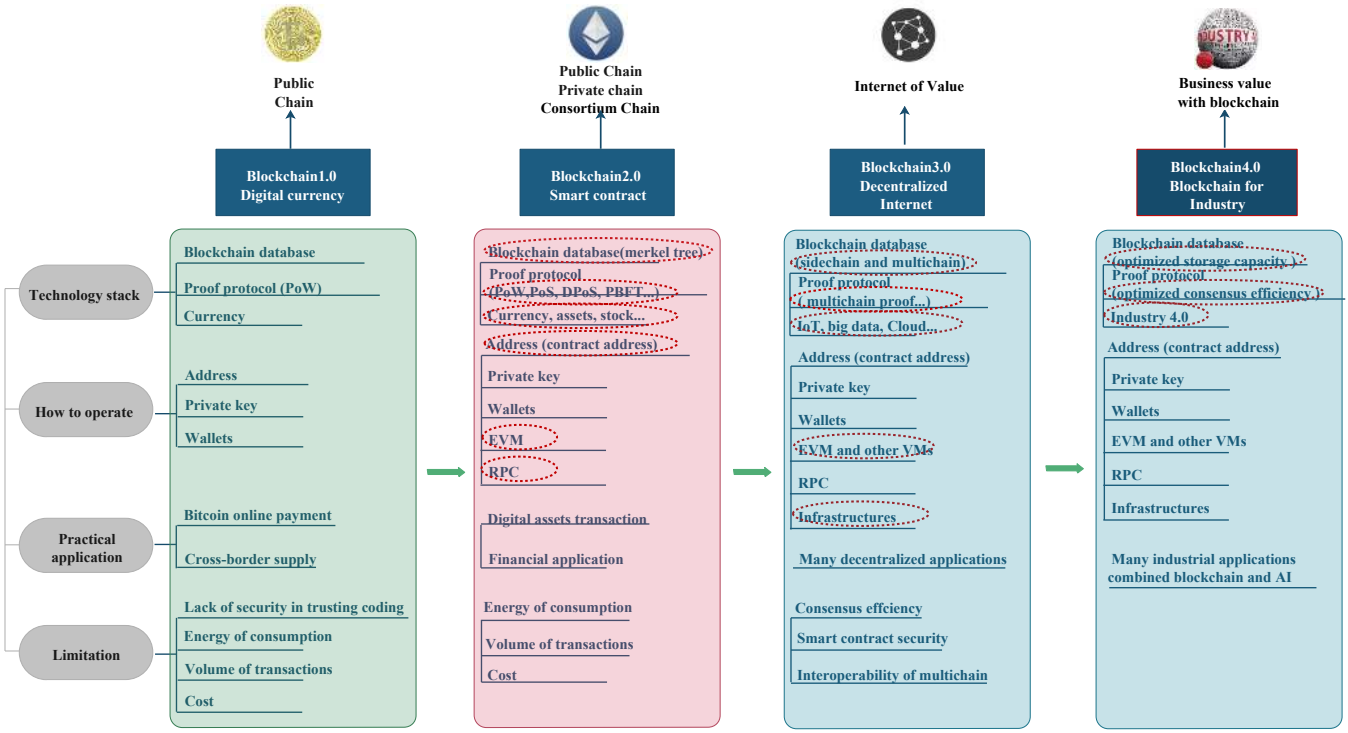


FIGURE 23: The evolution of Blockchain system.

several minutes needed for one block confirmation). To apply the blockchain-based internet service, we summarize future challenges mapped to proposed key requirements as shown in Table 9 .

From table 9, it can be seen that there are a few challenges that need to be addressed before the current blockchain technologies can concurrently assure decentralization, scalability and consistency with billions of transactions in each second. Here, we outline the main challenges to six areas:

1) Storage capacity: in blockchain, there is a requirement for all transactions to be stored in each node and this record is immutable, ensuring data integrity and continuity. However, this introduces the problem of excessive system storage due to the characteristics of non-erasable and distributed storage. Therefore, there is a need to design and develop an optimized model of decentralized but robust, reliable and load-balanced storage to allocate data based on the performance of individual nodes.
2) Consensus performance and efficiency: the consensus protocol plays a key role in the scalability of blockchain networks. However, the current consensus methods always require long verification times for transactions, even when there is a relatively small

TABLE 9: Key requirements of blockchain-based internet mapped to existing blockchain elements

Type	Key requirement	Existing blockchain elements					
		Data structure	Consensus protocol	Multichain proof protocol	Decentralized control	Standard contract	Smart contract
Database	Data security	✓	✓	✓	✓	✓	✓
	Data storage capacity	×	×	N/A	N/A	N/A	N/A
	Data management	✓	✓	✓	✓	✓	✓
Protocol	Confirmation time	N/A	✓	✓	N/A	✓	✓
	Performance and efficiency	N/A	×	×	N/A	N/A	N/A
	Resource consumption	N/A	×	×	N/A	N/A	N/A
	Tolerance power	N/A	✓	✓	N/A	N/A	N/A
Application	Scalability	×	×	×	N/A	N/A	N/A
	Privacy and security	✓	✓	✓	✓	✓	✓
	Data management	×	×	×	×	N/A	✓
Contract	Transaction fee	N/A	N/A	N/A	N/A	N/A	✓
	Programming performance	N/A	N/A	N/A	N/A	×	×

number of nodes. It can be seen that the performance and efficiency of current consensus protocols needs to be improved.

3) Protocol scalability: current blockchain protocols are effective in securing and managing the data stored within the network. However, newer systems fail to scale after some threshold of record and network size [108]. In order to maintain a coherent and synchronized state of information, a blockchain data structure, in particular for multichain data should be provided to enable communication in a secure and efficient manner without affecting security. This also involves the challenge of both identifying and determining the number of nodes that should have a transaction validation role in order to ensure the best protocol efficiency.
4) Resource consumption: since a small fee is required as an incentive to pay miners for maintaining the distributed ledger (by solving a computationally-expensive problem), this scheme is not satisfactory for massive volumes of transactions due to the prohibitive power (and fiscal) cost. As a consequence, there is a need to seek diminished global power consumption.
5) Personalization mining: providing methods to personalize blockchain for specific Internet services is another important challenge. Artificial intelligence (AI) algorithms can help to solve this by making different parts of the blockchain 'smarter'. For example, node behaviour can be learned via their history of actions to make intelligent decisions. In another example, deciding whether a node should be used in transaction verification or determining the weighting/contribution level of different nodes in the whole network is challenging.
6) Contract performance: the contracts used in blockchain-based systems are computer programs intended to facilitate, verify, or enforce the negotiation or performance of a prior agreement. Unfortunately, current smart contracts do not use the full potential of arbitrary programs, which would allow for a much more semantically-rich environment and a lack of associated contract evaluation. When an arbitrary contract code is enabled, the code requires a rigorous and robust compilation and evaluation system to determine contract pre and post-conditions. Otherwise, the fulfillment of a contract may be vague and subject to unwanted side effects or errors. Another limitation is that contracts cannot change what should in essence be stored due to the current immutability of blocks (or the underlying immutable database metaphor). A layer enabling mutable objects to be stored (distributed and decentralized) is also required but not at the expense of the trust and integrity of the data.

VIII. CONCLUSION

In this paper, we have conducted a comprehensive survey on current informative Internet service architectures together with blockchain technologies used to understand the challenges of the internet service architectures and the benefits of blockchain compared with traditional centralized-based mechanism. We presented the vision of building a blockchain-based Internet service architecture which was designed to achieve a trustworthy Internet in a decentralized manner, then discussed its key technical requirements from

different aspects related to the proposed architecture, and analyzed the trends and challenges mapping to these key requirements. The main purpose of this study is to guide more detailed and innovative solutions to implement the future trustworthy Internet service. This style of service architecture will not only meet the massive information requirements of contemporary and emerging systems, but also coupled with the secure, fair and scalable environments such systems are currently lacking.

ACKNOWLEDGMENT

This research was supported by Smart Services and Systems research group which has carried out fundamental and applied research in research areas including blockchain, expert systems, Web Services and so on. We would like to thank all group members for providing references and suggestions related to our research field.

REFERENCES

[1] R. Braden, D. Clark, and S. Shenker, "Integrated services in the internet architecture: an overview," Tech. Rep., 1994.
[2] A. Ljungqvist and W. J. Wilhelm Jr, "Ipo pricing in the dot-com bubble," The Journal of Finance, vol. 58, no. 2, pp. 723–752, 2003.
[3] B. Van Schewick, Internet architecture and innovation. Mit Press, 2012.
[4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[5] B. Zhou, J. Pei, and W. Luk, "A brief survey on anonymization techniques for privacy preserving publishing of social network data," ACM Sigkdd Explorations Newsletter, vol. 10, no. 2, pp. 12–22, 2008.
[6] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis, "A framework for efficient data anonymization under privacy and accuracy constraints," ACM Transactions on Database Systems (TODS), vol. 34, no. 2, p. 9, 2009.
[7] X. Xiao, G. Wang, and J. Gehrke, "Differential privacy via wavelet transforms," IEEE Transactions on knowledge and data engineering, vol. 23, no. 8, pp. 1200–1214, 2011.
[8] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," in Advances in neural information processing systems, 2014, pp. 2879–2887.
[9] W. He, D. Akhawe, S. Jain, E. Shi, and D. Song, "Shadowcrypt: Encrypted web applications for everyone," in Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2014, pp. 1028–1039.
[10] C. Peikert, "Lattice cryptography for the internet," in International Workshop on Post-Quantum Cryptography. Springer, 2014, pp. 197–219.
[11] F. Heuer, T. Jager, E. Kiltz, and S. Schäge, "On the selective opening security of practical public-key encryption schemes," in IACR International Workshop on Public Key Cryptography. Springer, 2015, pp. 27–51.
[12] J. Yli-Huumo, D. Ko, S. Choi, S. Park, and K. Smolander, "Where is current research on blockchain technology?âĂŤa systematic review," PloS one, vol. 11, no. 10, p. e0163477, 2016.
[13] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in internet of things: challenges and solutions," arXiv preprint arXiv:1608.05187, 2016.
[14] M. Conoscenti, A. Vetro, and J. C. De Martin, "Blockchain for the internet of things: A systematic literature review," in Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of. IEEE, 2016, pp. 1–6.
[15] A. Hari and T. Lakshman, "The internet blockchain: A distributed, tamper-resistant transaction framework for the internet," in Proceedings of the 15th ACM Workshop on Hot Topics in Networks. ACM, 2016, pp. 204–210.
[16] A. Chakravorty and C. Rong, "Ushare: user controlled social media based on blockchain," in Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication. ACM, 2017, p. 99.
[17] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, "Medshare: Trust-less medical data sharing among cloud service providers via blockchain," IEEE Access, vol. 5, pp. 14 757–14 767, 2017.
[18] R. Dennis and G. Owen, "Rep on the block: A next generation reputation system based on the blockchain," in 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST). IEEE, 2015, pp. 131–138.
[19] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, L. Jia-Nan, Y. Xiang, and R. Deng, "Crowdbc: A blockchain-based decentralized framework for crowdsourcing," IEEE Transactions on Parallel and Distributed Systems, 2018.
[20] V. Gramoli, "From blockchain consensus back to byzantine consensus," Future Generation Computer Systems, 2017.
[21] P. Ridyard, "Tempo, our ledger and consensus tech," 2018.
[22] G. Zyskind, O. Nathan et al., "Decentralizing privacy: Using blockchain to protect personal data," in Security and Privacy Workshops (SPW), 2015 IEEE. IEEE, 2015, pp. 180–184.
[23] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in 2016 IEEE symposium on security and privacy (SP). IEEE, 2016, pp. 839–858.
[24] E. Duffield and K. Hagan, "Darkcoin: Peertopeer cryptocurrency with anonymous blockchain transactions and an improved proofofwork system," bitpaper. info, 2014.
[25] M. Milutinovic, W. He, H. Wu, and M. Kanwal, "Proof of luck: An efficient blockchain consensus protocol," in Proceedings of the 1st Workshop on System Software for Trusted Execution. ACM, 2016, p. 2.
[26] R. Pass and E. Shi, "Hybrid consensus: Efficient consensus in the permissionless model," in LIPIcs-Leibniz International Proceedings in Informatics, vol. 91. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.
[27] Y. Guo and C. Liang, "Blockchain application and outlook in the banking industry," Financial Innovation, vol. 2, no. 1, p. 24, 2016.
[28] Q. K. Nguyen, "Blockchain-a financial technology for future sustainable development," in Green Technology and Sustainable Development (GTSD), International Conference on. IEEE, 2016, pp. 51–54.
[29] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on. IEEE, 2017, pp. 618–623.
[30] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot," in Proceedings of the Second International Conference on Internet-of-Things Design and Implementation. ACM, 2017, pp. 173–178.
[31] J. C. Song, M. A. Demir, J. J. Prevost, and P. Rad, "Blockchain design for trusted decentralized iot networks," in 2018 13th Annual Conference on System of Systems Engineering (SoSE). IEEE, 2018, pp. 169–174.
[32] D. Larimer, N. Scott, V. Zavgorodnev, B. Johnson, J. Calfee, and M. Vandeberg, "Steem: An incentivized, blockchain-based social media platform," March. Self-published, 2016.
[33] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," in Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. IEEE Press, 2017, pp. 468–477.
[34] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Blockchain-based database to ensure data integrity in cloud computing environments," 2017.
[35] Y. Zhang, R. Deng, X. Liu, and D. Zheng, "Outsourcing service fair payment based on blockchain and its applications in cloud computing," IEEE Transactions on Services Computing, 2018.
[36] Z. Lu, Q. Wang, G. Qu, and Z. Liu, "Bars: a blockchain-based anonymous reputation system for trust management in vanets," in 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, 2018, pp. 98–103.
[37] W. Viryasitavat, L. Da Xu, Z. Bi, and A. Sapsomboon, "Blockchain-based business process management (bpm) framework for service composition in industry 4.0," Journal of Intelligent Manufacturing, pp. 1–12, 2018.
[38] H. Su, F. Li, Q. Liang, C. X. Miao, and L. Xin, "Trustable web searching verification in a blockchain," May 17 2018, uS Patent App. 15/349,299.
[39] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, "Evaluation of logic-based smart contracts for blockchain systems," in International Symposium on Rules and Rule Markup Languages for the Semantic Web. Springer, 2016, pp. 167–183.

[40] L. Aniello, R. Baldoni, E. Gaetani, F. Lombardi, A. Margheri, and V. Sassone, "A prototype evaluation of a tamper-resistant high performance blockchain-based transaction log for a distributed database," in Dependable Computing Conference (EDCC), 2017 13th European. IEEE, 2017, pp. 151–154.

[41] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 7, pp. 1366–1385, 2018.

[42] E. Münsing, J. Mather, and S. Moura, "Blockchains for decentralized optimization of energy resources in microgrid networks," in Control Technology and Applications (CCTA), 2017 IEEE Conference on. IEEE, 2017, pp. 2164–2171.

[43] G. Fridgen, S. Radszuwill, N. Urbach, and L. Utz, "Cross-organizational workflow management using blockchain technology-towards applicability, auditability, and automation," 2018.

[44] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016, pp. 254–269.

[45] N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on ethereum smart contracts (sok)," in Principles of Security and Trust. Springer, 2017, pp. 164–186.

[46] O. Barais, A. F. Le Meur, L. Duchien, and J. Lawall, "Software architecture evolution," in Software Evolution. Springer, 2008, pp. 233–262.

[47] A. Melis, S. Mirri, C. Prandi, M. Prandini, P. Salomoni, and F. Callegati, "A microservice architecture use case for persons with disabilities," in International Conference on Smart Objects and Technologies for Social Good. Springer, 2016, pp. 41–50.

[48] A. Terzis, L. Wang, J. Ogawa, and L. Zhang, "A two-tier resource management model for the internet," GLOBECOM-NEW YORK-, vol. 3, pp. 1779–1791, 1999.

[49] B. Urgaonkar, G. Pacifici, P. Shenoy, M. Spreitzer, and A. Tantawi, "An analytical model for multi-tier internet services and its applications," in ACM SIGMETRICS Performance Evaluation Review, vol. 33, no. 1. ACM, 2005, pp. 291–302.

[50] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network and openflow: From concept to implementation," IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 2181–2206, 2014.

[51] A. Perrig, P. Szalachowski, R. M. Reischuk, and L. Chuat, SCION: a secure Internet architecture. Springer, 2017.

[52] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," Journal of Economic Perspectives, vol. 31, no. 2, pp. 211–36, 2017.

[53] E. N. P. Law and the Future of the Global Data Economy, "The g.d.p.r," https://www.newyorker.com/, 2018.

[54] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini, "Blockchain based decentralized management of demand response programs in smart energy grids," Sensors, vol. 18, no. 1, p. 162, 2018.

[55] S. Wurster, M. Böhmecke-Schwafert, F. Hofmann, and K. Blind, "Born global market dominators and implications for the blockchain avant-garde," in Corporate and Global Standardization Initiatives in Contemporary Society. IGI Global, 2018, pp. 86–115.

[56] W. Viriyasitavat and D. Hoonsopon, "Blockchain characteristics and consensus in modern business processes," Journal of Industrial Information Integration, 2018.

[57] M. Walport, "Distributed ledger technology: Beyond blockchain," UK Government Office for Science, 2016.

[58] M. Ali, J. C. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A global naming and storage system secured by blockchains." in USENIX Annual Technical Conference, 2016, pp. 181–194.

[59] G. Smaragdakis, N. Laoutaris, K. Oikonomou, I. Stavrakakis, and A. Bestavros, "Distributed server migration for scalable internet service deployment," IEEE/ACM Transactions on Networking (TON), vol. 22, no. 3, pp. 917–930, 2014.

[60] S. D. Gribble, E. A. Brewer, J. M. Hellerstein, and D. Culler, "Scalable, distributed data structures for internet service construction," in Proceedings of the 4th conference on Symposium on Operating System Design & Implementation-Volume 4. USENIX Association, 2000, p. 22.

[61] I. Bashir, "Mastering blockchain: Distributed ledgers, decentralization and smart contracts explained, book mastering blockchain: Distributed ledgers, decentralization and smart contracts explained," 2017.

[62] D. Zhaoyang, L. Fengji, and G. Liang, "Blockchain: a secure, decentralized, trusted cyber infrastructure solution for future energy systems," Journal of Modern Power Systems and Clean Energy, vol. 6, no. 5, pp. 958–967, 2018.

[63] P. Tasatanattakool and C. Techapanupreeda, "Blockchain: Challenges and applications," in Information Networking (ICOIN), 2018 International Conference on. IEEE, 2018, pp. 473–475.

[64] C. S. Ali Taghikhani and D. M. Kazemi, "Automating hiring with blockchain and artificial intelligence technologies," Tech. Rep., 2018.

[65] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer et al., "On scaling decentralized blockchains," in International Conference on Financial Cryptography and Data Security. Springer, 2016, pp. 106–125.

[66] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng, and V. C. Leung, "Decentralized applications: The blockchain-empowered software system," IEEE Access, vol. 6, pp. 53 019–53 033, 2018.

[67] A. McAfee, E. Brynjolfsson, T. H. Davenport, D. Patil, and D. Barton, "Big data: the management revolution," Harvard business review, vol. 90, no. 10, pp. 60–68, 2012.

[68] S. Siewert, "Why software engineers and developers should care about blockchain technology," white paper, April, 2018.

[69] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," URL: http://www. opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains, 2014.

[70] N. Bozic, G. Pujolle, and S. Secci, "A tutorial on blockchain and applications to secure network control-planes," in Smart Cloud Networks & Systems (SCNS). IEEE, 2016, pp. 1–8.

[71] S. Raval, Decentralized Applications: Harnessing Bitcoin's Blockchain Technology. " O'Reilly Media, Inc.", 2016.

[72] G. Greenspan, "Multichain private blockchainâĂŤwhite paper," URl: http://www. multichain. com/download/MultiChain-White-Paper. pdf, 2015.

[73] "Rootstock," https://www.rsk.co/, 2015.

[74] "Alpha," https://www.alpha-blockchain.co/, 2018.

[75] "Liquid," https://blockstream.com/liquid/, 2018.

[76] "Zilliqa," https://zilliqa.com/, 2016.

[77] "Rchain," https://www.rchain.coop/, 2017.

[78] "Rchain," https://quarkchain.io/, 2017.

[79] Z. T. L. D. Xie Zhuopeng, Ding Ying, "Iot chain: A high-security lite iot os," https://iotchain.io/, 2015.

[80] S. Popov, "Byteball wiki," https://byteball.org/, 2017.

[81] ——, "The tangle," https://iotchain.io/, 2018.

[82] Z.-d. CHEN, Y. Zhuo, Z.-b. DUAN, and H. Kai, "Inter-blockchain communication," DEStech Transactions on Computer Science and Engineering, no. cst, 2017.

[83] S. King, "Primecoin: Cryptocurrency with prime number proof-of-work," July 7th, 2013.

[84] A. Chepurnoy, T. Duong, L. Fan, and H.-S. Zhou, "Twinscoin: A cryptocurrency via proof-of-work and proof-of-stake." IACR Cryptology ePrint Archive, vol. 2017, p. 232, 2017.

[85] H. B. Alla, S. B. Alla, A. Touhafi, and A. Ezzati, "A novel task scheduling approach based on dynamic queues and hybrid meta-heuristic algorithms for cloud computing environment," Cluster Computing, vol. 21, no. 4, pp. 1797–1820, 2018.

[86] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016, pp. 3–16.

[87] S. King and S. Nadal, "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake," self-published paper, August, vol. 19, 2012.

[88] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. bft replication," in International Workshop on Open Problems in Network Security. Springer, 2015, pp. 112–125.

[89] A. Kiayias, A. Russell, B. David, and R. Oliynykov, "Ouroboros: A provably secure proof-of-stake blockchain protocol," in Annual International Cryptology Conference. Springer, 2017, pp. 357–388.

[90] D. Larimer, "Delegated proof-of-stake (dpos)," Bitshare whitepaper, 2014.

[91] ——, "Transactions as proof-of-stake," Nov-2013, 2013.

[92] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed, "Zookeeper: Wait-free coordination for internet-scale systems." in USENIX annual technical conference, vol. 8, no. 9. Boston, MA, USA, 2010.

[93] K. Birman, D. Malkhi, and R. Van Renesse, "Virtually synchronous methodology for dynamic service replication," Appears as Appendix A in [4], 2010.
[94] L. S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in Advanced Computing and Communication Systems (ICACCS), 2017 4th International Conference on. IEEE, 2017, pp. 1–5.
[95] D. J. Madigan, Z. Baumann, and N. S. Fisher, "Pacific bluefin tuna transport fukushima-derived radionuclides from japan to california," Proceedings of the National Academy of Sciences, vol. 109, no. 24, pp. 9483–9486, 2012.
[96] S. Jeon, I. Doh, and K. Chae, "Rmbc: Randomized mesh blockchain using dbft consensus algorithm," in Information Networking (ICOIN), 2018 International Conference on. IEEE, 2018, pp. 712–717.
[97] C. Cachin and M. Vukolić, "Blockchains consensus protocols in the wild," arXiv preprint arXiv:1707.01873, 2017.
[98] "Interledger architecture overview," https://interledger.org/, 2016.
[99] L. Lee, "New kids on the blockchain: How bitcoin's technology could reinvent the stock market," Hastings Bus. LJ, vol. 12, p. 81, 2015.
[100] D. Wang, J. Zhou, A. Wang, and M. Finestone, "Loopring: A decentralized token exchange protocol," URL https://github. com/Loopring/whitepaper/blob/master/en_whitepaper. pdf, 2018.
[101] "Btc relay," http://btcrelay.org/, 2016.
[102] https://www.rsk.co/, 2018.
[103] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," See https://lightning. network/lightning-network-paper. pdf, 2016.
[104] https://www.fusion.xyz/blockchain/, 2018.
[105] V. Buterin, "Chain interoperability," R3 Research Paper, 2016.
[106] L. Deng, H. Chen, J. Zeng, and L.-J. Zhang, "Research on cross-chain technology based on sidechain and hash-locking," in International Conference on Edge Computing. Springer, 2018, pp. 144–151.
[107] https://etu.link/files/etu-whitepaper.pdf., 2018.
[108] W. Yang, S. Garg, A. Raza, D. Herbert, and B. Kang, "Blockchain: Trends and future," in Pacific Rim Knowledge Acquisition Workshop. Springer, 2018, pp. 201–210.
[109] E. Zmaznev et al., "Bitcoin and ethereum evolution," 2018.
[110] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," Ieee Access, vol. 4, pp. 2292–2303, 2016.
[111] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Gollamudi, G. Gonthier, N. Kobeissi, N. Kulatova, A. Rastogi, T. Sibut-Pinote, N. Swamy et al., "Formal verification of smart contracts: Short paper," in Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security. ACM, 2016, pp. 91–96.
[112] R. M. Parizi, A. Dehghantanha et al., "Smart contract programming languages on blockchains: An empirical evaluation of usability and security," in International Conference on Blockchain. Springer, 2018, pp. 75–91.
[113] G. Peters, E. Panayi, and A. Chapelle, "Trends in cryptocurrencies and blockchain technologies: a monetary theory and regulation perspective," Journal of Financial Perspectives, vol. 3, no. 3, 2015.
[114] V. Cerf, Y. Dalal, and C. Sunshine, "Specification of internet transmission control program," Tech. Rep., 1974.
[115] M. Burstein and R. Pelavin, "Hierarchical channel router," Computer-Aided Design, vol. 16, no. 4, pp. 216–224, 1984.
[116] L. Shuchun, L. Mengyang, W. Shixian, and X. Zhenqin, "The design and implementation of the browser/server mode mis [j]," Computer Engineering and Applications, vol. 6, p. 038, 2000.
[117] D. Barkai, "An introduction to peer-to-peer computing," Intel Developer update magazine, pp. 1–7, 2000.
[118] S. Halevi and H. Krawczyk, "Strengthening digital signatures via randomized hashing," in Annual International Cryptology Conference. Springer, 2006, pp. 41–59.
[119] M. Swan, Blockchain: Blueprint for a new economy. " O'Reilly Media, Inc.", 2015.
[120] J. L. Zhao, S. Fan, and J. Yan, "Overview of business innovations and research opportunities in blockchain and introduction to the special issue," 2016.
[121] N. Joshi, "Blockchain meets industry 4.0-what happened next?" 2017.
[122] "LitecoinïijŽthe cryptocurrency for payments," https://litecoin.org/, 2013.
[123] "Dogecoin," https://dogecoin.com/, 2013.
[124] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in 2017 IEEE international congress on big data (BigData congress). IEEE, 2017, pp. 557–564.
[125] A. Larmuseau and D. M. Shila, "Private blockchain configurations for improved iot security," Blockchain for Distributed Systems Security, p. 255, 2019.
[126] Z. Zheng, S. Xie, H.-N. Dai, and H. Wang, "Blockchain challenges and opportunities: A survey," International Journal of Web and Grid Services, vol. 1, pp. 1–25, 2016.
[127] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges." IJ Network Security, vol. 19, no. 5, pp. 653–659, 2017.

WENLI YANG received the B.E.degrees in Communication Engineering from Wuhan University, Wuhan, China, in 2007, and got the Ph.D. degree in System Analysis and Integration from Huazhong University of Science and Technology, Wuhan, China, in 2012, during her Ph.D, she also studied in University of California, Davis as a visiting scholar. She was an Associate Professor in China, and now she is a PhD student in the School of Technology, Environments and Design, University of Tasmania. Her research area includes blockchain, machine learning and expert systems.

ERFAN AGHASIAN He received B.Eng degree in Information Technology at Qazvin Azad University (Barajin), MSc. degree in Information Technology Management at University Technology of Malaysia (UTM) and the PhD degree in Information Technology at University of Tasmania. His research interests include Computer Systems and Network Security, Bloackchain, Data Security and Data Anonymization.

SAURABH GARG is a lecturer at the University of Tasmania, Australia. He is one of the few Ph.D. students who completed in less than three years from the University of Melbourne. He has published more than 40 papers in highly cited journals and conferences. During his Ph.D., he has been awarded various special scholarships for his Ph.D. candidature. His research interests include resource management, scheduling, utility and grid computing, Cloud computing, green computing, wireless networks, and ad hoc networks.

DAVID HERBERT is currently a PhD student in the School of Technology, Environments and Design, University of Tasmania. After nearly two decades as a senior systems support officer and software developer he decided to apply his technical skills in pursuing a long-delayed academic career. His research interests include expert systems, machine learning and robotics in the School of Technology, Environments and Design, University of Tasmania.

LEANDRO DISIUTA received his B.S. degree in Mechatronics Engineering (Automation and Control) from Pontifical Catholic University of Rio Grande do Sul, Brazil, in 2011. He received his M.Sc. degree in Electrical Engineering with emphasis in aerospace Biomedical Engineering from Pontifical Catholic University of Rio Grande do Sul, Brazil, in 2014. He is currently pursuing the Ph.D. degree in computer engineering at University of Tasmania, Hobart, Australia. His research interests are smart systems and remote laboratories.

BYEONG KANG is a Professor in School of Technology, Environments and Design, University of Tasmania, Australia. Prof. Kang received his PhD from the University of New South Wales, Sydney, in 1996, and has worked as a visiting researcher in the Advanced Research Lab HITACHI situated in Japan. His research interests include basic knowledge acquisition methods and applied research in Internet systems as well as medical expert systems. Pro. Kang leads the Smart Services and Systems research group of postdoctoral scientists, which has carried out fundamental and applied research in research areas, expert systems, SNS analysis and smart industry areas. He has served as a chair and steering committee member in many international organizations and during conferences.