DOCTORAL THESIS

Novel Information and Data Exchange within Power Systems Using Enhanced Blockchain Technologies

A thesis submitted for the degree of

Doctor of Philosophy

by

Mubashar Amjad

Department of Electronic and Electrical Engineering

Brunel University London

February 2023

Declaration of Authorship

I, Mubashar Amjad, hereby affirm that the material contained within this thesis has not been previously submitted for any other academic award and is in compliance with the University's guidelines and regulations for research. Additionally, I certify that all the work presented in this thesis is my own original work and that any assistance received in conducting the research and composing the thesis has been appropriately acknowledged and cited.

Signature

Mubashar Amjad

March 2023

Abstract

Current energy systems are primarily designed for centralized power generation and supplying bulk electricity to users with stable and predictable usage patterns. However, with the increasing penetration of renewable energy sources (RES), future energy systems will require greater flexibility and wider distribution of both demand and supply. Integrating RES on a large scale poses challenges to the hosting capacity of distribution systems. To address these challenges, the digitalization of energy systems through novel Information and Communication Technologies (ICT) infrastructure is essential. The shift from centralized to highly distributed systems necessitates increased coordination and communication efforts. This is because a distributed system is composed of multiple independent entities that need to communicate and collaborate effectively to accomplish a shared objective. Coordination and communication are necessary to ensure that the system is operating efficiently and effectively.

Traditional centralized cloud-based data exchange schemes depend on a single trusted third party, this may lead to single-point failure and lack of data privacy and access control. To overcome these issues, a novel approach is proposed for exchanging data within power systems using blockchain technology. This approach enables users to securely exchange data while maintaining ownership. The experiments conducted demonstrate that the proposed approach can handle more users and enables information and data exchange within power systems.

Secondly, this thesis proposes an Artificial Neural Network (ANN) based prediction model to optimize the performance of the blockchain-enabled data exchange approach. A use case for exchanging data within the power system is implemented on the proposed platform using various performance metrics. The results of the proposed approach are compared to two other schemes: the baseline scheme and an optimized scheme. The evaluation results indicate that the proposed approach can enhance network performance when compared to the baseline and optimized schemes.

In summary, the proposed novel approach to ICT infrastructure for successfully exchanging information and data within power systems entities. The performance of the novel approach is evaluated based on the ability to handle multiple users, scalability, reliability, and security.

Acknowledgment

I am deeply grateful to all those who have supported me throughout my PhD journey. First and foremost, all praise and gratitude to God, without whose guidance and blessings this achievement would not have been possible.

I am deeply indebted to my primary supervisor, Prof. Gareth Taylor, for his constant guidance, support, and invaluable feedback throughout my research. His expertise and encouragement have been an invaluable asset to my growth as a researcher. I would also like to extend my appreciation to Prof. Maozhen Li, Rd. Zhengwen Huang, and Dr. Chun Sing Lai for their valuable feedback and advice during my PhD.

I would also like to express my deepest gratitude to my family, especially my parents, for their unwavering love, support, and encouragement throughout my studies. Their sacrifice and dedication have been the foundation of my success. Additionally, I would like to thank my sisters for their support and motivation throughout my journey.

Lastly, I would like to thank my friends for their constant support and encouragement. This thesis would not have been possible without the love and support of my loved ones.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

BUC	Business Use Case
DRES	Distributed Renewable Energy Sources
DSO	Distribution System Operator
HDFS	Hadoop Distributed File System
HTTP	Hypertext Transfer Protocol
ICT	Information and Communications Technology
IDS	Intrusion Detection System
IIoT	Industrial Internet of Things
IT	Information Technology
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
PBFT	Practical Byzantine Fault Tolerance
PoW	Proof of Work
RES	Renewable Energy Sources
REST	Representational State Transfer
UC	Use Case
VM	Virtual Machine
TPS	Transaction Per Second
TSO	Transmission System Operator

Chapter 1 Introduction

1.1 Background

Traditional energy systems are primarily designed for centralized power generation, providing bulk electricity to users with stable and predictable usage patterns. However, as RES becomes more prevalent, future energy systems will require greater flexibility and wider distribution of both demand and supply. Gartner defines digitization as the implementation of digital technologies to change a business model and create new revenue and value-generating opportunities. It is the journey towards becoming a digital business [1]. The energy system has been moving towards digitalization for the past 10 years. The primary focus to date has been on the operation of infrastructure and the implementation of the Smart Grid concept. The present concept of digitalization is very broad and it includes social aspects as well. With the idea of Smart Grid, digitization involves new factors which are:

- New business models for customer involvement
- Integration of different power sector entities

The hosting capacity of distribution grids is currently challenged by the large-scale integration of RES. The digitalization of energy systems using ICT is considered a crucial element in addressing these challenges [2]. Numerous changes brought about by the shift from centralized to distributed management have necessitated more coordination and communication efforts. Interoperability is needed on various levels, from connectivity to regulatory policy, among system operators and market participants for improved cooperation [3]. The European Mandate M/490 has outlined an approach for interoperability within the smart grid [4] The complexity of the power system has made it challenging to specify the requirements between actors in the Smart Grid. ICT will play a crucial role in the transition of the energy sector.

The integration of RES like solar and wind into distribution networks has presented TSOs and DSOs with new difficulties. Due to their decentralized nature, it can be difficult for TSOs and DSOs to have complete visibility and control over the entire system. This can result in problems such as power imbalances, voltage fluctuations, and difficulties in forecasting energy demand [5,6]. RES, such as solar and wind, can be distributed and unpredictable, creating challenges for TSOs and DSOs in terms of managing power flows and ensuring the stability of the grid. Congestion management, voltage regulation, and frequency are key aspects of supply security, as they ensure that the power system can meet the demand for electricity while maintaining stability and reliability. To ensure supply security in these areas, it is essential to maintain overall system balance. TSOs and DSOs need to enhance visibility on each other's systems. As the concern increases that one operator's actions may have a significant impact on other system operators, overall system control has decreased as a result of the replacement of conventional generation techniques with RES at the transmission and distribution level [7]. It is common practice for TSOs and DSOs to exchange information and data, but to enhance overall system interoperability, exchanging data within power systems must be considerably improved [8]. This will not only make the system operate more smoothly overall, but it will also enable other power system entities to participate. As the level of information and data exchange increases, businesses will have more options to operate more flexibly when controlling demand and generation [8].

Interoperability is considered a key facilitator in Smart Grids and is defined as "the ability of two or more devices from the same vendor, or different vendors, to exchange information and use that information for correct cooperation" [9]. Interoperability in the context of Smart Grids refers to the ability of different devices, systems, and technologies to communicate and work together seamlessly, regardless of their manufacturer or protocol. This means that devices from different vendors should be able to share information and work together in a coordinated way to achieve common goals. Without interoperability, different devices and systems in a Smart Grid will not be able to effectively communicate with each other, which would make it difficult to manage and control the grid. According to this definition, two or more systems are considered interoperable if they can perform a specific function together by exchanging information. Figure 1-1 illustrates the concept of interoperability. This definition is considered valid for the entire Smart Grid.

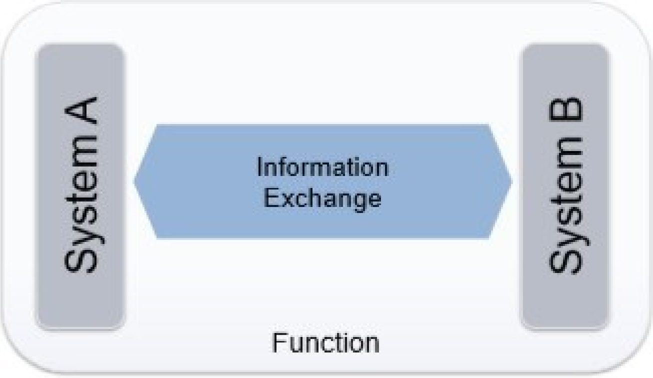


Figure 1-1 Concept of Interoperability [9]

GridWise Architecture Council [GWAC2008] introduced interoperability categories that describe the requirements to attain interoperability between the systems [10]. From the definition mentioned previously, interoperability is classified into three main categories: technical, informational, and organizational. Figure 1-2 below illustrates the Interoperability classification by GWAC. All of these categories are covered by the standards and specifications.

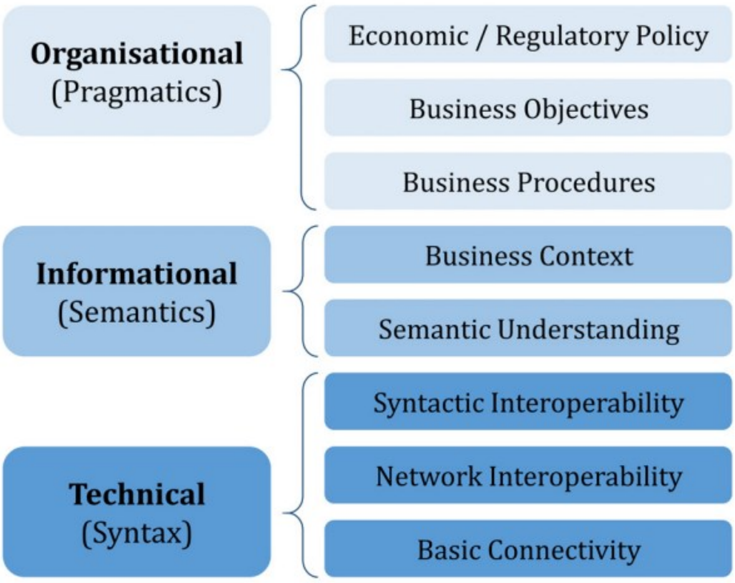


Figure 1-2 Interoperability classification by GWAC [10]

Cross-cutting issues were defined by GWAC (Global Grid Forum) in 2008. These cross-cutting issues are defined as issues that affect several or all categories of a system. Some of the issues are energy efficiency, cybersecurity, resource identification, reliability/scalability, and system evolution [11].

1.2 Research Motivation

The majority of today's energy systems are built for centralized power generation, which delivers large amounts of electricity to consumers with steady and predictable usage patterns. Since RES like solar and wind are extremely weather-dependent and can be dispersed and unpredictable, this creates challenges for the energy system in terms of managing power flows and ensuring the stability of the grid. As the integration of these sources increases, it requires a more flexible and even distribution of demand and supply, The system must be capable of adjusting to the variations of these sources and balancing demand and supply accordingly. The increased demand for integrating RES puts pressure on the hosting capacity of distribution systems. To tackle these problems, new ICT infrastructure and the digitalization of energy systems are necessary. There have been significant changes brought about by the shift from centralized to highly dispersed system management, which has prompted more coordination and communication efforts.

There are some areas in which TSO-DSO data exchange is limited frequently. Closer interaction between grid operators will need these grid areas for power systems operation and planning perspective [12].

- Ancillary Services

  The distribution energy sources such as disturbed energy recourses (DRES) demand-side response (DSR), and flexible thermal generating units could provide ancillary services to the system. TSOs could benefit from these services to avoid the storage of ancillary services [12]. As a result of this, DSOs will need to become more active, as most of these services are connected to the distribution network. This will assist TSOs in providing a variety of system services using the distributed energy sources mentioned above [12]. It has been essential for both TSOs and DSOs to collaborate and exchange data about these flexible energy resources.
- Congestion Management

  Congestion of the distribution and transmission grid is expected to become more consistent because of the variable generation being fed into the network. This causes changes in power flow directions and paths. Active power management is a key tool that will help both the system operators deal with the issue of congestion [12]. Some of the methods that can be used to perform active power management are: managing energy storage devices, active power control RES, and a device like FACTS, or HVDC links are also used [12]. To cover the issue of overloading in the network a more optimal solution is needed. In this regard, TSO-DSO data exchange is of key importance so that actions can be taken by the grid operators.

- Voltage Control

  TSOs are usually responsible for maintaining voltage levels at the desired range using devices like capacitors and tap changing of the reactor. DSOs also utilise these devices to regulate the distribution network's reactive power [12]. The voltage level of the system is an important factor in the efficient and stable operation of the power system. The voltage level must be maintained within a specific range to ensure that the equipment and devices connected to the power system are operating correctly. In this regard, data exchange between TSO-DSO regarding reactive power control resources is necessary to enable both network operators. More benefits can be exploited if devices on the DSOs side are made available for TSO usage and vice-versa [12].

- System Security

  Ensuring the security of the power system is critical for maintaining the reliability and stability of the system. The power system is made up of many different components, including generation, transmission, and distribution, and ensuring the security of each of these components is necessary for the overall security of the system.

  Ensuring the security of the power system is a critical task and TSOs and DSOs play a vital role in this. The transmission system operators (TSOs) manage the planning and operation of the transmission network, while the distribution system operators (DSOs) are in charge of the management and maintenance of the distribution network. Improving collaboration between TSOs and DSOs is necessary to enhance the short-term and long-term security of the system, as they can work together to coordinate the operation and maintenance of the transmission and distribution networks.

However, TSO’s operational security analysis does not consider the distribution network's components due to a lack of information on the DSO side. This is because TSOs typically have less visibility and information about the distribution network, which makes it difficult for them to accurately assess and manage the security of the system. The exchanging of data between TSOs and DSOs will help secure the operation of the system by performing a joint assessment of security [12]. The collaboration of TSOs and DSOs will help utilize the capabilities of DRES which will contribute to restoring the system more quickly and securely.

- System Planning and Development

  System planning and development need significant investment for the expansion of transmission and distribution grids. These activities need to be performed economically and efficiently. For development and integration purposes both transmission and distribution operators should work collaboratively to perform these functions effectively and economically [12]. By improving the collaborations decisions can be made jointly and unnecessary investments can be avoided. This collaboration will encourage intelligent investment on both sides of the grid.

1.3 Blockchain comparison with existing methods

Blockchain data exchange differs from other methods like IEC 61850 and SSL (Secure Sockets Layer) in several key ways. Here's a comparison of blockchain data exchange with these two methods:

Data Integrity and Immutability:

Blockchain: Blockchain ensures data integrity and immutability by using cryptographic techniques and a distributed ledger. Once data is recorded on the blockchain, it is extremely challenging to alter or delete it without consensus from the network participants.

IEC 61850: IEC 61850, primarily designed for communication in the electrical power system, focuses on data exchange and standardization but does not inherently guarantee data immutability or tamper resistance.

SSL: SSL provides encryption and data security during transmission but does not inherently guarantee the immutability of data once it reaches the destination.

Decentralization and Trust:

Blockchain: Blockchain operates in a decentralized manner, eliminating the need for a central authority. Trust is established through consensus mechanisms, ensuring that data is validated and agreed upon by network participants.

IEC 61850: IEC 61850 typically relies on centralized systems or trusted authorities within the power system, which may introduce single points of failure or vulnerabilities.

SSL: SSL relies on trusted certificate authorities (CAs) to verify the identity of parties involved in data exchange. Trust is centralized around these certificate authorities.

Ownership and Control:

Blockchain: In a blockchain-based system, users retain ownership and control over their data. Smart contracts can automate data exchange while ensuring data ownership remains with the users.

IEC 61850: Control over data may be more centralized within the electrical power system, and data ownership might not be as clearly defined in the standard.

SSL: SSL secures data in transit but does not inherently address data ownership and control. Ownership and control are typically determined by the entities handling the SSL certificates.

Transaction Transparency:

Blockchain: Blockchain provides transparency through a public ledger where all transactions are recorded and visible to participants. This transparency enhances trust and accountability.

IEC 61850: While it facilitates data exchange within the electrical power system, IEC 61850 may not provide the same level of transparency, especially to external parties.

SSL: SSL encryption hides the content of data during transmission, making it secure but less transparent.

Consensus Mechanisms:

Blockchain: Blockchain networks use consensus mechanisms like Proof of Work (PoW) or Proof of Stake (PoS) to validate and agree on the state of the ledger.

IEC 61850 and SSL: These methods do not incorporate consensus mechanisms as they primarily focus on secure data exchange and communication protocols.

In summary, blockchain data exchange stands out for its focus on data integrity, decentralization, trust, user control, transparency, and consensus mechanisms. While IEC 61850 and SSL serve specific purposes in the context of data exchange and security, they may not provide the same level of data immutability and trust that blockchain technology offers, especially in scenarios where data tampering and trust are critical concerns.

1.4 Aim and objectives

This research aims to enable the exchange of information and data within power systems in a scalable and secure manner. The research project focuses on three key aspects of ICT tools and techniques: interoperability, security, and scalability. By focusing on these three key aspects, the research project aims to develop new ICT tools and techniques that can help power system entities to exchange information and data in a more efficient, secure, and reliable manner, providing better integration and coordination between different entities of the power system. To achieve the aim, the following objectives will be addressed:

1. To investigate the research area behind applying different ICT tools for exchanging information and data within the power systems.
2. To review the literature on different coordination schemes to enhance interoperability within the entities of power systems.
3. To address information and data exchange between TSOs and DSOs by developing a platform.
4. The developed platform should be able to exchange information and data in a scalable, reliable, and secure manner.

5. The implementation of the use case for exchanging data within power systems using the proposed platform.
6. To optimize the proposed platform to enhance the performance capabilities in terms of scalability.
7. The comparison of the proposed platform with existing schemes to demonstrate its effectiveness in enhancing network performance

1.5 Contribution to the knowledge

The main contributions to knowledge, as presented in this thesis, can be summarised as follows:

- Comprehensive Literature Review: The thesis starts by providing a thorough review of previous research studies and literature related to the topic. This review serves as a foundation for understanding the current state of the field. It focuses on identifying various Information and Communication Technology (ICT) tools and techniques used in power systems to exchange information and data. This sets the stage for the subsequent developments in the thesis.
- Cloud Computing for data exchange: The thesis introduces the integration of Cloud Computing techniques into the power system. It highlights the scalability of this platform, which is crucial for accommodating an increasing number of users and enabling efficient information and data exchange within the power system. This integration represents a significant advancement in the field.
- Service-Oriented Methodology with Virtual Machines: The thesis outlines the use of a service-oriented methodology, implemented on virtual machines (VMs), to ensure secure and reliable data exchange within power systems. The approach utilizes clusters of VMs, which are software-based simulations of physical servers, to create a cloud computing platform. Multiple use cases are deployed on this platform to validate its suitability and effectiveness for practical applications.
- Blockchain-Centric System Architecture: A major contribution is the design of a system architecture centered around blockchain technology. This architecture ensures the integrity of shared data within power systems. It

also emphasizes user control and ownership of data, which is a critical aspect of data exchange in sensitive environments like power systems. Additionally, the use of Smart Contracts for workflow automation enhances the efficiency and trustworthiness of data exchange processes.

- Optimized Performance with Artificial Neural Networks (ANN): The thesis proposes the use of Artificial Neural Networks (ANN) as a learning-to-prediction model to optimize the performance of the blockchain-based data exchange platform. By estimating optimal latency and throughput, this model enhances the scalability of the platform. This contribution addresses a critical aspect of real-world implementation, ensuring that the system can handle the demands of a dynamic power system effectively.

1.6 Thesis layout

The research is organized into seven chapters. Chapter 1 outlines the motivations of the research presented in this thesis. The research has been conducted to understand how ICT tools can help in exchanging information and data within power systems for operating the overall system efficiently and securely. Therefore, relevant information regarding the background is provided in the section. The main aim and objectives of the research are presented in section 1.3. Following the introductory chapter, the next five include includes more details regarding the theoretical background and the thesis objectives.

In Chapter 2, the importance of power system information exchange is discussed by reviewing the relevant literature. To be specific a brief overview of the challenges faced by TSOs and DSO is presented. Then a detailed review of the importance of enhancing the information exchange in the power system is presented. The chapter also reviews the ICT tools that can be useful in increasing interoperability for power system users. These approaches are examined, and relevant literature is provided in this chapter.

Chapter 3 presents a review of ICT tools that can be useful in increasing interoperability for power system users. These approaches are examined, and relevant literature is provided in this chapter. Cloud computing platforms and blockchain platforms are critically evaluated in this chapter.

Chapter 4 presents the detailed architecture of the proposed cloud platform for the exchange of information and data within power systems. A service-oriented approach for mapping the Business Use Case services on a VM cluster is presented in this chapter. To establish the data exchange cloud platform's suitability for the intended use of priority information and data needs, it is assessed for scalability and reliability.

Chapter 5 introduces our blockchain-based platform for the secure sharing of information and data within the power system. By integrating blockchain and big data technologies like Hadoop, the proposed approach aims to provide a secure, efficient, and scalable solution for exchanging information and data within the power system. Blockchain is a decentralized digital ledger that records transactions across a network of computers. It is often used for secure and transparent record-keeping, and it can provide a solution to the current limitations of a centralized system. Trustworthy transactions of data are ensured by our proposed blockchain platform. The testing results will give an insight into the proposed approach's effectiveness in addressing the issues of data exchange in power systems.

Chapter 6 proposes a novel deep Artificial Neural Optimization (ANN) based scheme for performance optimization for blockchain-based information and data exchange platforms. This framework is used to determine the optimal block size and block interval to achieve high throughput and low latency for our platform. A range of use cases is tested to validate the effectiveness of the proposed scheme.

Chapter 7 summarizes the key findings of the work presented in this thesis, highlighting the major contributions of the thesis. Additionally, this chapter outlines the research's limitations and provides suggestions for further study.

1.7 List of publications

- Amjad, M., Taylor, G., Li, M. and Huang, Z., 2021, December. A Critical Evaluation of Cloud Computing Techniques for TSO and DSO Information and Data Exchange. In *2021 11th International Conference on Power and Energy Systems (ICPES)* (pp. 481-485). IEEE.
- Amjad, M., Taylor, G., Lai, C.S., Huang, Z. and Li, M., 2022, August. A Novel Blockchain-Based Approach to Exchanging Information and Data in Power

Systems. In *2022 57th International Universities Power Engineering Conference (UPEC)* (pp. 1-6). IEEE.

- Amjad, M., Taylor, G., Lai, C.S., Huang, Z. and Li, M., 2022. Scalability and Reliability Analysis of a Novel Cloud Platform for TSO-DSO Information and Data Exchange. In *2022 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe).* IEEE.
- Amjad, M., Taylor, G., Lai, C.S., Huang, Z. and Li, M., 2023. Performance Optimization of a Blockchain-Enabled Information and Data Exchange Platform for Smart Grids (invited paper has been submitted to MDPI Electronics).

Chapter 2 Literature Review

2.1 Introduction

In recent years, the methods used to generate, transmit, and distribute electricity have undergone a shift in the electrical power system. Additionally, the network's power flow has changed from being unidirectional to bidirectional. Congestion on the transmission and distribution networks has increased as a result of the addition of more RES to the system. To ensure the safe and reliable operation of active power systems, managing these changes and the growing interactions across the networks will continue to be a significant challenge. As a result, there is a greater need and demand for increased engagement between TSOs and DSOs. This chapter presents the main issues with the interaction between TSOs and DSOs and discusses the literature review of different possible implementations of ICT tools such as cloud computing and blockchain to enhance the interaction between different power system entities.

2.2 Challenges faced by TSOs and DSOs

The common challenges that are faced by TSOs and DSOs include:

- Renewable energy integration
- Enhance controllability and observability of grid
- Flexible electricity market

2.2.1 Renewable energy integration

In recent years, traditional methods of electricity generation through fossil fuels have decreased, while the use of RES, such as solar and wind power, has increased [13]. The integration of RES into the grid is happening at an unprecedented pace, and TSOs and DSOs are playing an important role in facilitating this transition. They are responsible for integrating these renewable sources into the grid and ensuring that the power system remains stable and reliable. Some European countries have successfully made this renewable integration. TSOs and DSOs are dealing with the problems that are being faced in integrating these sources. Even though RES are connected to the grid, they have not yet been fully integrated into grid operations. This means that TSOs and DSOs are still working on developing the necessary infrastructure and technology to enable full integration of RES into the grid operations. [14]. In the future, this integration of non-consistent energy sources will increase and it will be unsustainable. To support this, these traditional generating units play an important role. Network constraints often happen at distributions and transmission networks because of the abundance of variable RES. This issue needed to be handled more efficiently and economically [15].

2.2.2 Enhance controllability and observability of grid

TSOs and DSOs are facing new challenges as a result of the growing integration of unstable energy resources. Because of this integration, there is limited or no controllability and observability [16]. TSOs are lacking observability as large generation facilities are connected to lower voltage levels. On the other hand, DSOs have no observability over the transmission network. A greater need for TSOs and DSOs cooperation is required over the observability issue to adequately support each other in system operation [17]. There is a significant reduction in generation controllability as countries are moving towards more non-traditional ways of generating electricity. These common issues are faced by both TSOs and DSOs to improve the observability and controllability of the system.

2.2.3 Facilitate flexible electricity market

With consumers showing interest in managing and producing their electricity, there is a new opportunity to make the electrical system more adaptable. Consumers can benefit from this non-discriminatory access to the electricity market [18]. The flexible service provided distribution side can further be exploited. TSOs and DSOs are responsible for coordinating these decentralized assets for dependable and secure system operation. To keep the network running smoothly and safely while keeping costs down, these resources can offer ancillary services [19,20,21]. A non-discriminatory use of these resources can benefit the whole electrical system [22]. With other stakeholders, TSOs and DSOs need to improve existing barriers to exploit these flexible resources transparently and competitively.

2.3 Common challenges among TSOs and DSOs

Common challenges that are posed by TSOs and DSOs are described in the above section. Stronger collaboration between TSOs and DSOs is needed to address these challenges. Potential areas that need improvement concerning TSO-DSO collaboration are presented in Table 2.1.

Table 2-1 Areas required improvement in TSO-DSO interaction

Areas to improve TSO-DSO Interaction	**Solution**
Voltage control	Coordination between TSO-DSO using flexible resources connected to the distribution network (reactive power management)
System Security	Joint assessment of TSO-DSO to mitigate anticipated risk. TSO-DSO both contribute to system restoration by exploiting resources available at the distribution side.
System planning	Collaborative planning of transmission and distribution network
State estimation	To improve state estimation, data exchange by both TSOs and DSOs is required on each other’s network
Congestion management	Collaborative use of flexible resources mainly connected to the distribution side will help in active power management in real-time
Ancillary services	TSO and DSO coordination is needed to harness flexible resources connected to the distribution network.
Optimal work planning	Wider operation planning through shared coordination between TSOs and DSOs

To enhance the visibility of each other's networks, TSOs and DSOs must collaborate more effectively. This includes sharing information and data about the status and operation of their respective networks. However, the information exchanged between TSOs and DSOs needs to be handled with great care as it may contain sensitive and confidential information. To improve observability, TSOs and DSOs need to classify the data they exchange and ensure that only the necessary information is shared. Additionally, TSOs and DSOs need to adopt suitable protocols and standards to ensure that the data is exchanged in a format that can be easily understood and used by the other participants [16,23]. TSOs and DSOs should work together to define common principles to improve their coordination.

2.4 Data exchange practice in TSO-DSO interaction

It is necessary to define the data based on the roles that are played by the various system operators to make certain that TSOs and DSOs have access to the information that is necessary for them to effectively fulfill their responsibilities [23,22].

2.4.1 Principles of data exchange

The fundamental principles for data exchange between TSOs and DSOs include:

- Data security and privacy
- Ensure competition in the electricity market and enable new business partners in the market
- Non-discriminatory access to data
- Data preservation guarantee
- Cost-efficient data exchange
- Exchange of data using standards

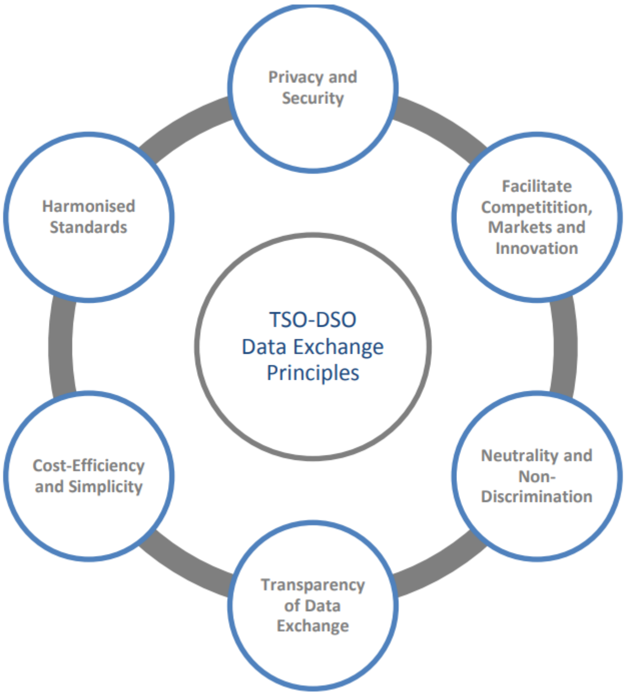

Privacy and Security
Facilitate Competitition, Markets and Innovation
Neutrality and Non-Discrimination
Transparency of Data Exchange
Cost-Efficiency and Simplicity
Harmonised Standards
TSO-DSO Data Exchange Principles

Figure 2-1 TSO-DSO data exchange principles [23]

2.4.2 Requirements for data exchange

The data exchange requirements are based on the roles that are performed by TSOs and DSOs. This data exchange can be classified differently. The data exchange requirement that TSOs and DSOs must work jointly is [24]:

- Frequency of data exchange
- Type of data exchange
- The granularity of data exchange

The frequency of data exchange is an important factor to consider when exchanging information and data between TSOs and DSOs. For real-time data exchange, such as monitoring the status of the power system and responding to potential issues, a higher frequency is needed [24]. The type of data exchange between TSO and DSO is divided into different types:

- Real-time data refers to data that is collected and made available at the exact time it is generated. It is used for real-time analysis of the transmission and distribution system. This type of data is important for monitoring the status of the power system and responding to potential issues, such as power outages or equipment failures. It is also useful for load forecasting, system balancing, and for providing accurate data for market operations and settlement [24].

- Scheduled data refers to data that is collected and made available at a scheduled time interval. This type of data is used for the analysis of operational security for transmission or distribution systems during the time interval of operational planning. It is used for short-term forecasting, and also for long-term planning, such as to evaluate the future power flows, identify the potential congestion, and balance the system accordingly [24].
- Structural data refers to general information from the network and its relevant grid. It is used for distribution and transmission system operation and security analysis at any time interval and also used for system planning. It includes information such as the physical layout of the transmission and distribution networks, the location and specifications of power generators, transmission lines, and transformers, and the characteristics of the loads connected to the system [24].

The granularity of data refers to the degree of detail and specific information that is included in the data. The granularity of data needed by TSOs and DSOs depends on the tasks that they perform [25]. For example, TSOs may require more detailed data for real-time monitoring and control of the transmission network, while DSOs may require more detailed data for distribution network management [25].

2.4.3 Concept of observability area

The observability area is defined as the area of the power system that a system operator observes to properly operate its grid [25]. Each system operator has a responsibility area, which is the area that it is responsible for managing and maintaining. The observability area includes the responsibility area, as well as the surrounding areas that can affect the responsibility area.

The concept of observability area is important for data exchange between TSOs and DSOs because it helps to ensure that the necessary data is collected, stored, and shared in a format that can be easily understood and used by the other party. By jointly defining the observability area, TSOs and DSOs can agree on the specific data needs and the level of detail that is required for each task [25]. As these Small Unbundled

Generators (SUGs) are connected to the distribution network, it means that they have a great impact on the observability area of the TSOs and DSOs. The grids that are electrically close to the TSO-DSO interface are the ones that affect the observability of TSOs and DSOs [25]. The following figure 2-2 depicts the concept of observability area.

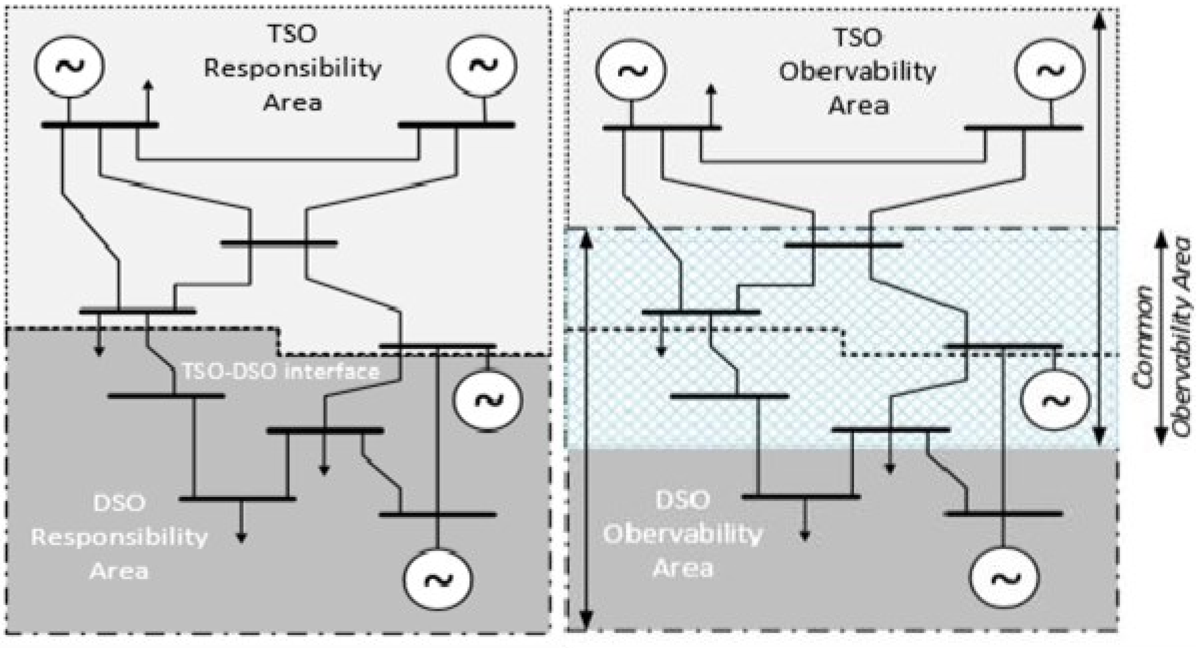
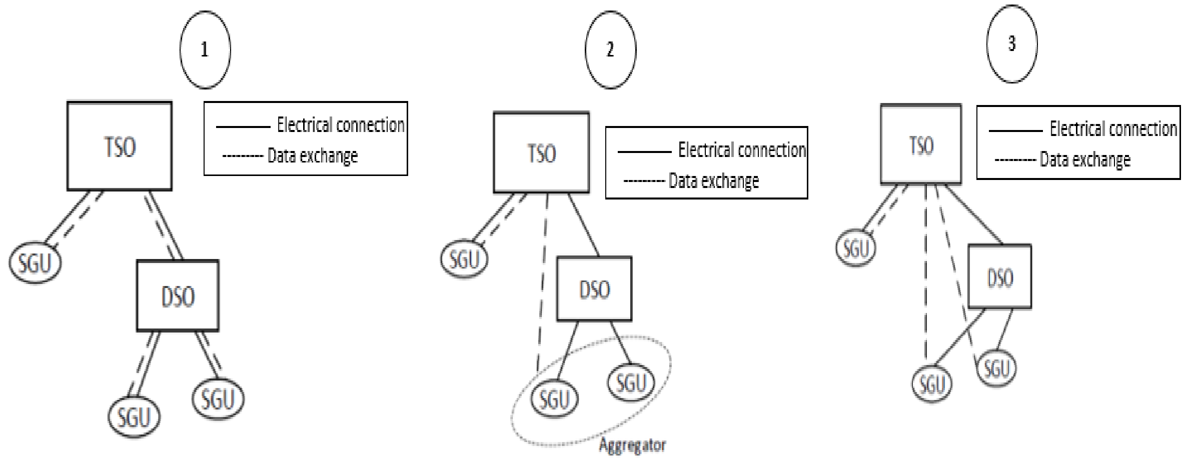


Figure 2-2 Observability area concept [25]

Currently, there is no established scheme for data exchange between TSOs and DSOs with SUGs. Figure 2-3 illustrates different possibilities of exchange schemes for data exchange between TSOs and DSOs with SUGs. These different possibilities of exchange schemes are designed to improve the observability and coordination of the power system, enabling TSOs and DSOs to effectively handle the integration of RES and ensure continuity of power supply to the end-users. The data exchange scheme must be agreed upon by TSOs, DSOs, and SUGs and it should be supported by the definition of the observability area of each other's network [25].



TSO access distribution-connected SGU data through the DSO

TSO access distribution-connected SGU data through an aggregator

TSO access distribution-connected SGU data directly through distribution-connected SGU

Figure 2-3 TSOs data exchange with the distribution connected SGUs [25]

2.5 System operation and flexibility market roles and responsibilities evolution

To increase cooperation among system operators, several position papers are being published to address the changes to the roles and responsibilities [26,27]. The following sections explain the important elements of these evolutions.

2.5.1 System operations

Optimal system planning and operation are supported by cooperation between system operators. For this relevant information is needed to be exchanged between system operators. This calls for a regulatory framework that will set the following criteria: i) reliability, ii) data privacy, iii) transparency, and iv) cost efficiency. Regarding different stages of system operation, such as long-term, short-term, and real-time, TSOs and DSOs need to improve data sharing to have better visibility into each other's network. To build a common understanding of each other's network, sharing a power system overview among all system operators is crucial [28]. This will help to ensure that all system operators are aware of the current state of the power system and any potential issues that may arise. By having a clear and detailed overview of the system, it will be easier for all system operators to manage the integration of new market participants and to ensure the continuity of power supply to the end-users [27]. Both network operators should work together in defining the requirements for observability [29].

2.5.2 Flexible market design

All entities connected to the transmission and distribution grid should participate in the energy market and offer their services. TSOs and DSOs should make sure that these services are available to all market consumers. A market framework defined by both TSOs and DSOs is required to make the market more flexible and accessible [54]. These flexible recourses can create a conflict between TSOs and DSOs depending on who will use these resources and for what purpose. Allocation of these resources should be based on economic and technical optimization [54]. A regulatory framework is needed to resolve the issue of allocating flexible resources between TSOs and DSOs [26].

The current balancing market is evolving and will keep evolving in the future. Some key point needs to be considered in this evolving market: i) trading for balancing purpose, ii) operational grid constraint to be included in the market for short time intervals, iii) economic way of using flexible resources [30]. More suitable flexible resources should be integrated into the market to support system operators [31].

2.5.3 Roles and responsibilities

Both TSOs and DSOs are responsible for the secure operation of their networks. This means that they must ensure that the power system is operating stably and securely and that the continuity of power supply to the end-users is maintained. One of the key responsibilities of TSOs and DSOs is to manage the grid, including the management of voltages and congestion on the grid [32]. The situation is evolving on the distribution side and more distributed energy resources are connected to the distribution network. New roles might emerge for DSOs in this regard. With the integration of RES and the increasing adoption of distributed energy resources, DSOs will play an important role in grid operations [31]. The roles of DSOs in the future will be of neutral market facilitators and providing security to the system operation. As market facilitators, DSOs should make sure that resources are available on the distribution side to the flexible market. DSOs will also be contributing to system security by providing support to the TSOs by providing a solution to the system-wide problems [32]. Similar rules are defined for TSOs as well. Both market participants cannot act as commercial service providers [32,33].

The current development in the market has not only changed roles of TSOs and DSOs but also of other market participants. The role of third-party aggregators is also evolving with time. In some countries, these third-party aggregators allow customers to take part in demand response. There is a lot of work going on regarding the standardization of the framework that will define the operational structure between third-party aggregators and balance response parties [31]. Transparency of these rules is important for the participants taking part in the flexible market [34]. Considering the heterogeneity of the roles and responsibilities, a one-fit-all solution cannot be provided.

2.6 Regulation and network code concerning TSO-DSO data exchange

The need for greater TSO-DSO collaboration is recognized by the regulators. Greater collaboration is required by both TSOs and DSOs to support the power system as a whole [35]. Considering this greater need, progress has been made in recent years to create an appropriate structure to support future TSO-DSO collaboration. At the European level Network code (NCs) provide the basic structure for TSO-DSO future collaboration. However, TSO-DSO collaboration is not directly explained in NCs. There are several topics discussed in NCs that are directly or indirectly relevant to TSO-DSO cooperation. The network codes developed are, i.e., system operation codes, network connection, and market-related codes. These network codes are drafted with guidance from the Agency for the Cooperation of Energy Regulation (ACER). The following is the details grid code that focuses on TSO-DSO data exchange [36].

2.6.1 Connection-related network codes

Connection codes and regulations are in place to ensure that generators, demand, and high voltage direct current are connected to the power system safely and securely.

2.6.1.1 Connection requirements for generators

The network code related to generators will increase competitiveness across the market. New generators connected to the grid must respect these harmonizing standards.

2.6.1.2 Connection requirements for demand code connection

Increasing the integration of renewable generation units is one of the main aims [37]. These connection codes set requirements for individual or third-party aggregation for demand facilities.

2.6.1.3 Connection requirements for high voltage direct current

These connection codes specify the requirements for High voltage direct current (HVDC) [38]. These codes are used for long-distance direct current (DC) connections i.e., interconnectors in Europe, HVDC submarine cable connections, and NorNed.

2.6.2 Operation-related network codes

Network codes are guidelines that have been developed by the European Union (EU) to ensure the proper functioning of the electricity market and the secure and efficient operation of the power system. These codes are relevant in the context of TSO-DSO data exchange as they define the rules and regulations for exchanging data among system operators. The main network codes that are relevant to TSO-DSO data exchange include Operation Planning and Scheduling (OPS), Operation Security (OS), Frequency Containment Reserves (FCR), and Frequency Restoration Reserves (FRR) [39].

- The guidelines on Operation Security (OS) set out rules and requirements for a transmission system that are applied to system operators including TSOs, DSOs, and other market participants. Both TSOs and DSOs should define and agree on the process of managing and providing real-time data exchange. These codes encourage real-time data exchange and provide a framework of the structure for this exchange of data.
- The guidelines on FCR and FRR encourage closer cooperation between TSOs and DSOs in the integration of RES. It provides guidelines for DSOs to act as an aggregator and Dos should collaborate. These codes emphasize increasing the observability of system operators on each other’s networks. DSOs should set limits on time before the reserve action happens. These limitations should happen transparently. System operators should collaborate on these arrangements.

- The guidelines on operation planning and scheduling (OPS) explain the minimum requirements for system operators to ensure operational planning is done in a coherent and coordinated way. This operational planning should apply to all relevant stakeholders i.e. TSOs, DSOs, and market participants.

There is a need for stronger cooperation among system operators as emphasized in network codes related to the operation. This greater cooperation will increase the frequency of data shared among system operators. The available data will enhance the system's observability which will provide a more accurate picture of the state of the system. In summary, data exchange between TSOs, DSOs, and other market participants is crucial for the efficient and secure operation of the power system. Therefore, all parties need to agree on the format and frequency of data to be exchanged for effective information sharing.

2.6.3 Market-related network codes

In the power system, the market-related network codes are divided into three categories: Forward Capacity Allocation (FCA), Electricity balancing (EB), and Congestion Allocation and Congestion Management (CACM). The FCA codes are related to the long-term market and forward markets, where participants can trade and secure capacity for a long time in advance [40]. These codes also set methodologies for defining bidding zones' capacity and criteria and processes to review bidding zones. FCA and CACM are not related to the relevant analysis and are not been further explored.

EB guidelines [41] are available for TSOs to share the resources that are used by them for balance generation and demand. These codes allow new participants to take an active part in this market. Electricity balancing guidelines will help provide system security, lower costs to consumers, and fewer emissions. These codes encourage data exchange between TSOs and DSOs concerning imbalance settlements. These codes allow the stage for the balance server providers and the requirements need to be defined by DSOs in terms of balance service providers.

2.7 TSO and DSO collaboration in the context of smart grid

For exchanging data between TSOs and DSOs, communication infrastructure tools are important nowadays. Such infrastructures are connected to the power systems devices and components to meet the requirements of communication. There are lots of communication technologies already implemented in existing power systems. Communication infrastructure that is developed in a power system must have ‘’wide bandwidth and low latency to support massive real-time data collection and data-driven large-scale grid optimization” [42]. An ICT system in the power system is a vital tool for monitoring, controlling, and optimizing the grid. To be effective, the system must meet certain requirements. One of the most important requirements is that the system must be able to provide reliable and secure transmission of wide-area field measurements and control commands. This is important for managing voltage and frequency throughout the entire network [43]. Fibber optical and wireless cellular technologies are considered to be appropriate for TSO-DSO communication. The data flow in the smart grid is shown in Figure 2-4.

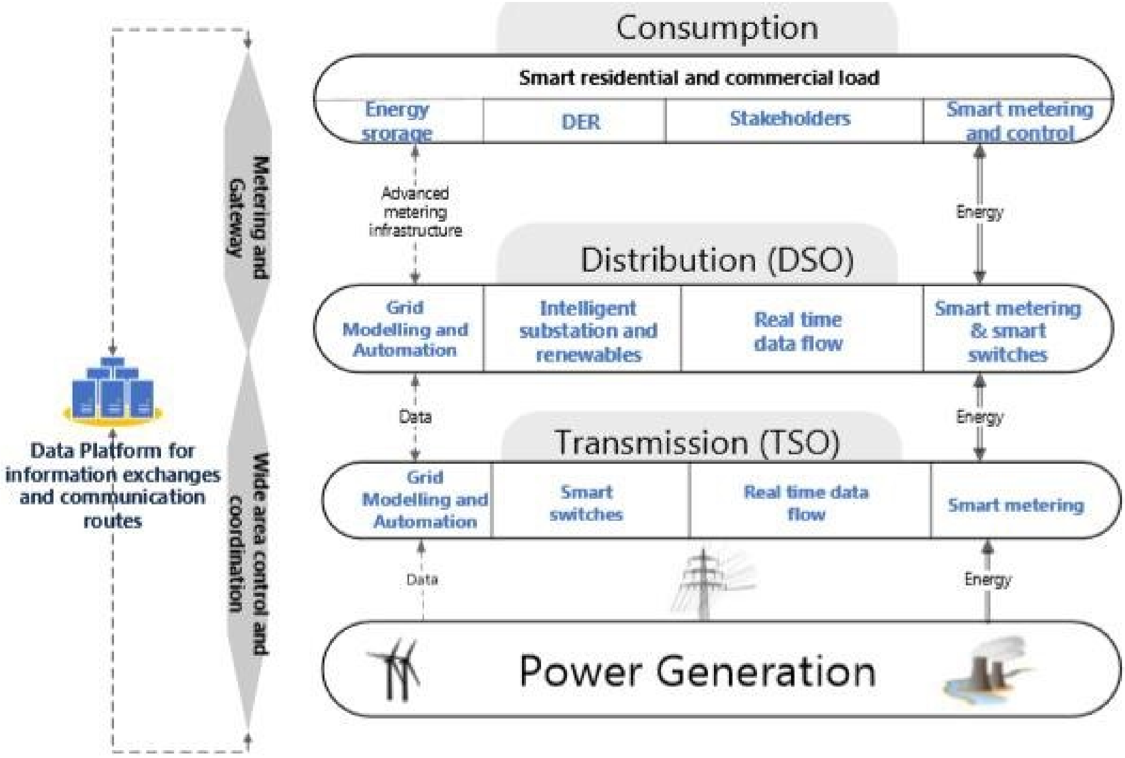

Consumption
Smart residential and commercial load
Energy srorage
DER
Stakeholders
Smart metering and control
Advanced metering infrastructure
Energy
Distribution (DSO)
Grid Modelling and Automation
Intelligent substation and renewables
Real time data flow
Smart metering & smart switches
Data
Energy
Transmission (TSO)
Grid Modelling and Automation
Smart switches
Real time data flow
Smart metering
Data
Energy
Power Generation
Metering and Gateway
Wide area control and coordination
Data Platform for information exchanges and communication routes

Figure 2-4 Bidirectional flow of data within smart gird [43]

The concept of the smart grid enables the free movement of data between different parts of grids. Different layers interact with each other in a smart grid that ensures a more secure and reliable operation through the network [43].

2.8 TSO-DSO data exchange ICT requirements

For future smart grids communication network is of great importance, it is considered the central nervous system. Communication infrastructure is important for collaboration between TSOs and DSOs [44]. This infrastructure will meet the challenges of connecting various power system devices and components. Communication standards are defined for communication technologies to perform tasks associated with them. The standards available for communication devices provide flexibility, scalability, and interoperability between the devices but raise security concerns. Inside the power system and electricity markets, different communication protocols are implemented to apply different levels of security.

Different communication technologies are currently applied in the power system. The reference architecture given by IEC TR 62357-1 explains the complexity of data exchange within the power system. These communication technologies must have the diversity to connect to the system devices. low bandwidth and high ability to support large real-time data are required by communication infrastructure [44]. This section reviews the current ICT infrastructure and models used for data exchange between TSOs and DSOs.

2.8.1Current ICT infrastructure

This section explains the current efforts being made to develop an infrastructure to enhance collaboration within power systems.

2.8.1.2 Explanation of reference architecture

TSO-DSO reference architecture explains the services and communication protocols that are used for power system management [45]. It emphasizes the boundaries between different standards where harmonization is necessary. The reference architecture is layered with minimal dependencies between each layer. It consists of

TC57 standards and the interface between them. TSO or DSO Reference Architecture with standards to be used is shown in Figure 2-5.

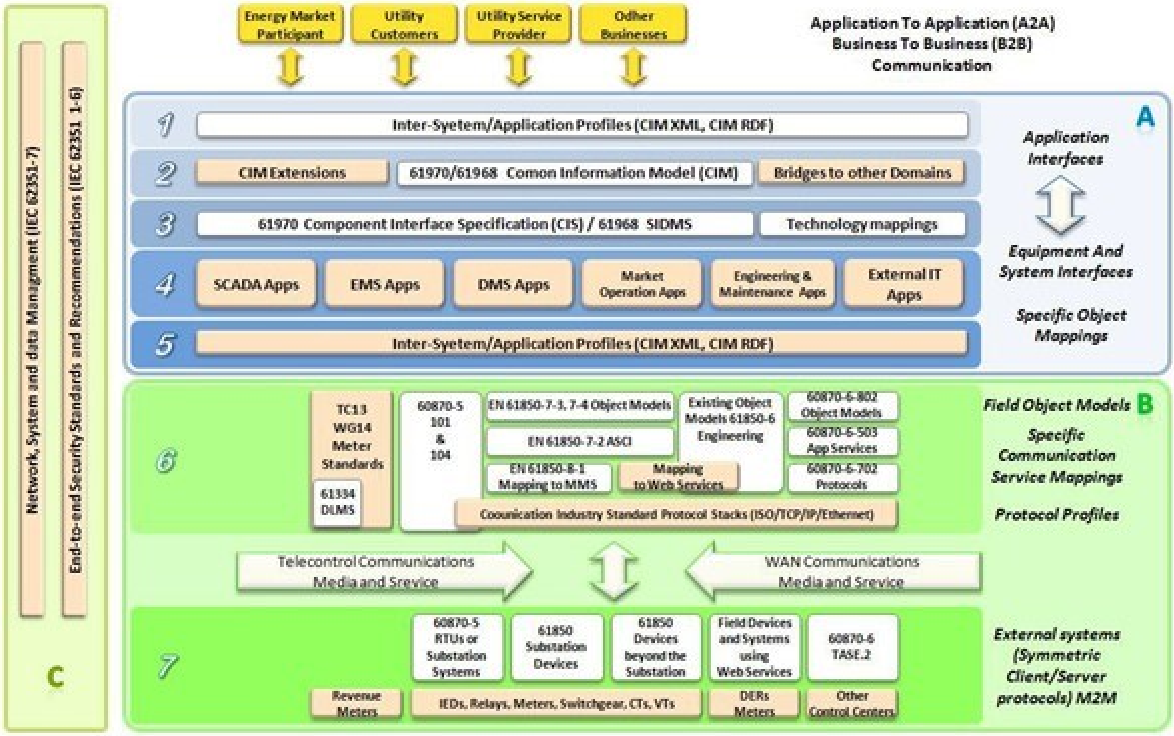

Energy Market Participant
Utility Customers
Utility Service Provider
Odher Businesses
Application To Application (A2A)
Business To Business (B2B)
Communication
A
1
Inter-Syetem/Application Profiles (CIM XML, CIM RDF)
2
CIM Extensions
61970/61968 Comon Information Model (CIM)
Bridges to other Domains
3
61970 Component Interface Specification (CIS) / 61968 SIDMS
Technology mappings
4
SCADA Apps
EMS Apps
DMS Apps
Market Operation Apps
Engineering & Maintenance Apps
External IT Apps
5
Inter-Syetem/Application Profiles (CIM XML, CIM RDF)
Application Interfaces
Equipment And System Interfaces
Specific Object Mappings
Network, System and data Managment (IEC 62351-7)
End-to-end Security Standards and Recommendations (IEC 62351 1-6)
C
B
6
TC13 WG14 Meter Standards
61334 DLMS
60870-5 101 & 104
EN 61850-7-3, 7-4 Object Models
EN 61850-7-2 ASCI
EN 61850-8-1 Mapping to MMS
Mapping to Web Services
Existing Object Models 61850-6 Engineering
60870-6-802 Object Models
60870-6-503 App Services
60870-6-702 Protocols
Coounication Industry Standard Protocol Stacks (ISO/TCP/IP/Ethernet)
Field Object Models
Specific Communication Service Mappings
Protocol Profiles
Telecontrol Communications Media and Srevice
WAN Communications Media and Srevice
7
60870-5 RTUs or Substation Systems
61850 Substation Devices
61850 Devices beyond the Substation
Field Devices and Systems using Web Services
60870-6 TASE.2
Revenue Meters
IEDs, Relays, Meters, Switchgear, CTs, VTs
DERs Meters
Other Control Centers
External systems (Symmetric Client/Server protocols) M2M

Figure 2-5 Reference Architecture example for TSO or DSO information exchange [45]

Part A of the Reference Architecture explains business integration between different stakeholders, data applications, and representation for distribution and transmission systems. In this section, data is represented using CIM-defined standards. The syntax and semantics for data exchange are based on CIM standards.

Part B of the Reference Architecture specifies the communication with external devices. The objects in the upper part of the models communicate using WAN (high-speed data links). Communication between field devices is more heterogeneous and it uses communication network topologies. SCADA is an example that uses both types of interface.

2.8.1.3 ICT data models

Data transfer is a crucial aspect of TSO-DSO collaboration for the efficient and reliable operation of the power system. When it comes to data representation, it is important to consider the application layer protocols and software interfaces. Data models can be classified into two categories: abstract models and concrete models.

Abstract models are technology-independent and are typically created using Unified Modelling Language (UML). They provide a high-level view of the data and its relationships, making it easier to understand the data's purpose and usage.

Concrete models, on the other hand, are technology-specific and are typically implemented using languages such as Extensible Markup Language (XML) or JavaScript Object Notation (JSON). XML (Extensible Markup Language) is a widely used markup language that is designed to store and transport data in a structured and human-readable format. JSON is a lightweight and widely adopted data interchange format. It is based on a subset of the JavaScript language and is easy for both humans and machines to read and write. JSON is commonly used in web APIs, configuration files, and various data exchange scenarios in modern web development. They provide a detailed view of the data and its implementation, making it easier to understand how the data is used in the system.

In TSO-DSO collaboration, both types of data models play an important role in understanding and utilizing the data effectively. It is important to have a clear understanding of the data representation and the different ways it can be classified for successful data transfer.

When an abstract model is implemented using technology, a concrete model is obtained. The standards for abstract models are IEC 61970/61968/62325, Common Information model (CIM), and IEC61850 [45]. For major power system components, CIM is the standard used for abstract modeling. CIM provides syntax and semantics for data exchange.

The TCP/IP (Transmission Control Protocol/Internet Protocol) model is another networking model that is widely used and is often compared to the OSI model. It is a more practical and commonly implemented model that corresponds closely to the functionality of the Internet. Let's discuss both models in terms of system interoperability:

OSI Model and Interoperability:

The OSI model is a theoretical and comprehensive framework that divides the networking process into seven distinct layers, each with its specific functions. While the OSI model provides a clear and structured approach to understanding networking protocols, it is not directly implemented in practice. Instead, it serves as a reference model for creating and understanding real-world networking protocols and technologies.

Interoperability in the context of the OSI model can be challenging because different vendors and systems may implement networking protocols differently. As a result, ensuring seamless communication and compatibility between devices from different manufacturers can be complex and may require extensive testing and standardization efforts.

TCP/IP Model and Interoperability:

The TCP/IP model, on the other hand, is a more pragmatic and widely used networking model, especially in the context of the Internet. It consists of four layers:

Application Layer: Corresponds to the OSI Application Layer. It includes protocols like HTTP, FTP, SMTP, and DNS, facilitating end-user interactions with network services.

Transport Layer: Combines the functions of both the OSI Transport and Session Layers. The two most important protocols in this layer are TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). TCP provides reliable, connection-oriented communication, while UDP offers unreliable, connectionless communication.

Internet Layer: Corresponds to the OSI Network Layer. This layer is responsible for routing packets across different networks and is where the IP (Internet Protocol) operates.

Link Layer: Corresponds to both the OSI Data Link and Physical Layers. It includes protocols and technologies related to the physical network medium and data link establishment, such as Ethernet and Wi-Fi.

Interoperability in TCP/IP Model:

The TCP/IP model has facilitated interoperability to a significant extent, particularly because it is the foundational model of the Internet. Most networking equipment and devices are built to be compatible with TCP/IP standards, leading to a high level of interoperability on the Internet.

TCP/IP's success in achieving interoperability is due to its practical design and widespread adoption. The simplicity of its four-layer structure and the versatility of IP as a routing protocol have made it the de facto standard for networking in a global context.

However, it's essential to note that even though TCP/IP facilitates broad interoperability at the network and transport layers, the application layer can still face challenges. Different applications may use proprietary or non-standard protocols, leading to potential issues in communication between specific applications or services.

In summary, while both the OSI model and TCP/IP model provide frameworks for understanding networking protocols and technologies, the TCP/IP model's practical implementation has contributed to greater system interoperability, particularly in the context of the Internet. The TCP/IP model's standardized approach at the network and transport layers has been instrumental in achieving seamless communication and connectivity across a vast array of devices and networks worldwide. However, achieving complete interoperability still requires adherence to common standards and protocols across all layers, especially at the application layer, to ensure smooth communication between various systems and services.

2.8.2 Overview of the current communication infrastructure

There is a growing interest in studying TSO-DSO cooperation, and as a result, there is a lot of ongoing research in this area. However, it is a relatively new topic and most of the research currently available is at a conceptual level. This section reviews the most recent conceptual models for TSO-DSO cooperation.

2.8.2.1 Coordination schemes present in Smart NET

The Smart Net project presents five coordination schemes for optimized interaction between TSOs and DSOs in managing data exchange for the better use of ancillary services. These coordination schemes provide the architecture for this interaction, and specific roles are defined for each scheme, which are taken up by system operators. These roles include the management of data exchange, the coordination of ancillary services, and the optimization of system performance. These schemes aim to enhance TSO-DSO interoperability, resulting in improved supply reliability and congestion control. By implementing these coordination schemes, the Smart Net project aims to improve the efficiency and effectiveness of data exchange between TSOs and DSOs in the power sector [46]. Details of these schemes are presented below.

Centralized AS market- In this scheme, TSO without the involvement of DSO can use the resources connected to both transmission and distribution networks. This scheme limits the involvement of DSOs, but still, has to provide observability over its network.

Local market- In this scheme DSOs will solve the issues on the distribution network and after solving will offer the remaining resources that are available to TSOs. DSOs will operate the local market where TSOs can indirectly contract with DER via that market.

Common TSO-DSO market- In this scheme both TSOs and DSOs have a common object to lower the cost of the resources needed by both system operators. Both system operators should coordinate together which resources should be used to deal with local constraints to maintain flexibility on both networks.

Integrated flexible market- In this scheme the market is open for all market operators i.e. both regulated and non-regulated. To guarantee neutrality an independent market operation is required.

Shared balancing responsibility- In this scheme shared responsibilities regarding balancing are predefined. TSOs have no access to these resources on the distribution network, where DSOs will arrange a local market for balancing purposes as agreed with TSOs.

2.8.2.2 Service platform by ENTSO-E

A service platform is developed by ENTSO-E that is used for data exchange between TSO and regional security coordinators [47]. The operational planning data environment (OPDE) supports the implementation of the Common information model (CIM). The Common Information Model is an international standard for data exchange in the energy field. OPDE supports Advance Message Queuing protocols for data exchange. An OPDE service platform can be used for TSO-DSO cooperation in the future. OPDE is divided into three categories, providing specific services:

Energy communication platform- It is a communication platform that is for message delivery. This platform has additional features like security, transparency, reliability, and portability.

EDX network- The EDX network is composed of two modules: toolbox and Service Catalogue. Both of these modules have their responsibility. Toolbox deals with the messaging interface and message delivery on the other hand Service catalog deals with the management of the network.

An operational planning data management network- OPDM is an application that is used for providing operational data. Both the client and service provider can interact using specific messages.

2.8.2.3 TDX-ASSIST project

In this project novel ICT techniques are developed for secure and scalable data exchange between TSOs and DSOs. The three main aspects of data exchange using ICT tools developed in this project are Security, Scalability, and Interoperability [48]. In this project, substantial work has been done regarding developing Business Use cases (BUCs) details are in Chapter 4. 11 business use cases were developed, and 13 System Used Cases (SUCs) were extracted regarding performing the field test. All the BUCs and SUCs were developed using the Modsarus© tool. Different demonstrations were done by different in this project for Security, Scalability, and interoperability. The defined BUCs and services are presented in Figure 2-6.

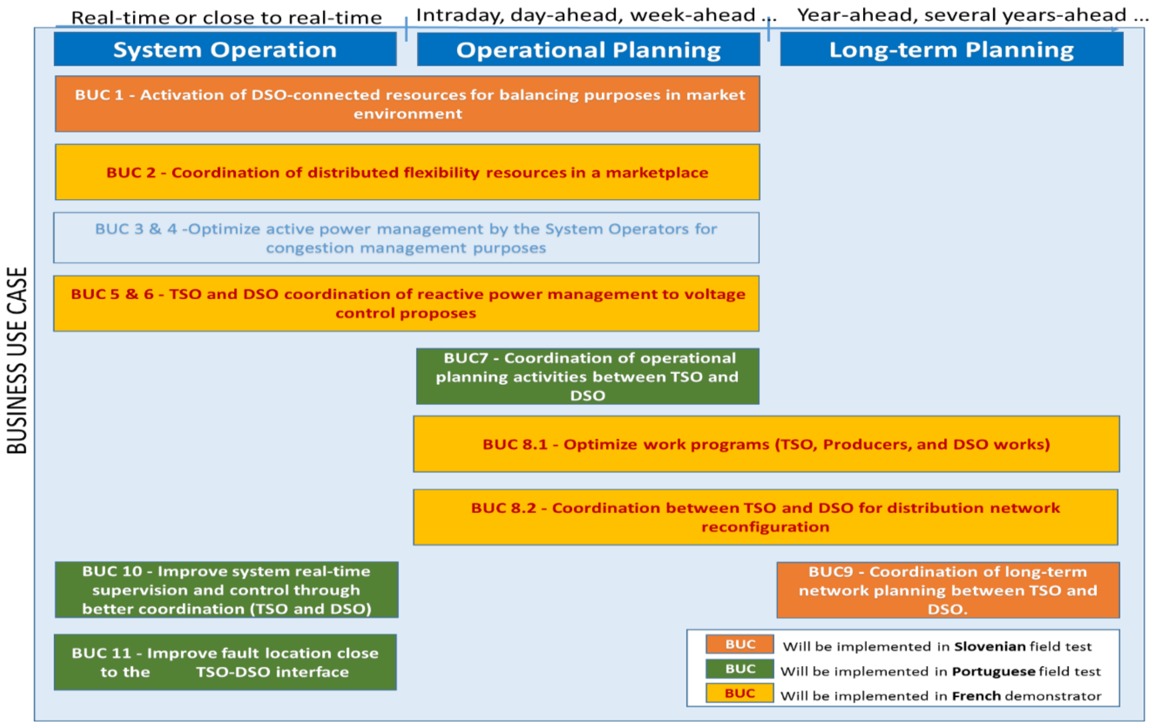

Real-time or close to real-time
Intraday, day-ahead, week-ahead ...
Year-ahead, several years-ahead ...
System Operation
Operational Planning
Long-term Planning
BUSINESS USE CASE
BUC 1 - Activation of DSO-connected resources for balancing purposes in market environment
BUC 2 - Coordination of distributed flexibility resources in a marketplace
BUC 3 & 4 -Optimize active power management by the System Operators for congestion management purposes
BUC 5 & 6 - TSO and DSO coordination of reactive power management to voltage control proposes
BUC7 - Coordination of operational planning activities between TSO and DSO
BUC 8.1 - Optimize work programs (TSO, Producers, and DSO works)
BUC 8.2 - Coordination between TSO and DSO for distribution network reconfiguration
BUC 10 - Improve system real-time supervision and control through better coordination (TSO and DSO)
BUC9 - Coordination of long-term network planning between TSO and DSO.
BUC 11 - Improve fault location close to the TSO-DSO interface
BUC Will be implemented in Slovenian field test
BUC Will be implemented in Portuguese field test
BUC Will be implemented in French demonstrator

Figure 2-6 Business use cases defined in the TDX-Assist project [48]

2.8.2.4 TSO-DSO Coordination Models

Transmission System Operators (TSOs) and Distribution System Operators (DSOs) play crucial roles in the operation of the electric power grid. Coordination between these two entities is essential to ensure the reliability and efficiency of the power system. Various coordination models and approaches have been developed to facilitate this collaboration. Here are some common TSO-DSO coordination models:

Vertical Coordination Model:

Overview: In the vertical coordination model, the TSO and DSO operate as distinct entities, each responsible for its part of the power grid. The coordination primarily occurs through hierarchical communication channels.

Use Cases: This model is often used when the TSO and DSO have well-defined responsibilities and do not frequently need to interact.

Information Sharing and Data Exchange:

Overview: TSOs and DSOs can establish information-sharing protocols and data exchange standards to improve coordination. This model focuses on the seamless exchange of operational data and information.

Use Cases: Information sharing can include real-time grid data, load forecasts, outage information, and more. It is essential for both planning and real-time grid management.

Market-Based Coordination:

Overview: Market-based coordination involves creating markets or platforms where TSOs and DSOs can buy and sell various grid services, such as flexibility, capacity, or demand response, to balance supply and demand.

Use Cases: This model encourages market participants to provide grid support services and fosters cooperation between TSOs and DSOs to ensure grid stability.

Integrated Control and Coordination:

Overview: Integrated control and coordination models aim to merge the operational functions of TSOs and DSOs to some extent. This can include joint dispatch of resources, shared control rooms, and combined planning efforts.

Use Cases: This model is particularly relevant in situations where the boundaries between transmission and distribution systems are blurred, such as in the integration of distributed energy resources (DERs).

Active Network Management (ANM):

Overview: ANM systems provide real-time visibility and control of distribution networks. They enable automatic or manual interventions in response to changing grid conditions and can be coordinated with TSO operations.

Use Cases: ANM systems help manage congestion, voltage control, and the integration of renewable energy sources at the distribution level, which has implications for the overall grid.

Multi-Agent Systems (MAS):

Overview: Multi-agent systems involve the use of software agents that represent TSOs, DSOs, and other entities in a decentralized manner. These agents negotiate and make decisions to optimize grid operations.

Use Cases: MAS can be used for real-time coordination, market transactions, and demand-side management, enabling efficient interactions between TSOs and DSOs.

Regulatory Frameworks and Standards:

Overview: Regulatory authorities often play a role in shaping TSO-DSO coordination models. They establish rules, guidelines, and standards to ensure fair and effective collaboration.

Use Cases: Regulatory frameworks can mandate data sharing, define roles and responsibilities, and encourage the adoption of coordination technologies.

These TSO-DSO coordination models aim to address the evolving challenges posed by increasing grid complexity, the integration of renewable energy sources, and the need for more efficient and reliable grid operations. The choice of coordination model may vary depending on the specific regulatory environment, market structures, and grid characteristics in a given region.

2.9 Chapter Summary

This chapter reviewed the main principles and challenges for exchanging information data within the power system. It also included the role and responsibilities of power system entities in the power system for exchanging information. Afterward, the overview of the current ICT infrastructure is reviewed. The concept of observability area both from TSOs and DSOs points of view is explained in the survey conducted. TSOs and DSOs must have adequate observability over each other's network concerning data exchange. The current TSO-DSO data exchange platforms were also discussed based on the available literature. The five most relevant challenges identified in the literature are:

- TSOs and DSOs need to enhance visibility over each other's networks.
- Need to coordinate operational optimization in different time scales.
- Facilitate consumers to take part in the market.
- The need for a well-organized and coordinated network design.
- Integration of variable renewable energies for better management of the networks.

However, there are still a series of challenges and issues to be addressed to improve the current level of information exchange within the power system.

Chapter 3 ICT Tools for Exchanging Information and Data

3.1 Introduction

Chapter 2 addressed the necessity of data exchange and provided an overview of the current infrastructure dedicated to this purpose. In this chapter discusses the technologies used for exchanging information and data in the context of the thesis. It emphasizes the importance of digitalization of energy systems using new ICT infrastructure and addressing the challenges of interoperability within power system entities. The chapter goes into detail about the specific technologies and protocols used for data exchange and how they can be applied to improve communication and coordination within the power system. It also discusses the benefits and limitations of these technologies and the potential impact they can have on the overall efficiency and reliability of the power system. Overall, this chapter aims to provide an in-depth understanding of the role of ICT in power systems and how it can be leveraged to solve interoperability issues.

3.2 Cloud computing for data exchange

Cloud computing is a pivotal technology for enhancing data exchange in power systems. It enables utilities and grid operators to efficiently store, process, and share vast amounts of data critical for managing electrical grids and ensuring reliability. Cloud platforms offer scalable storage solutions, data analytics tools, and secure communication channels, facilitating real-time data sharing among various stakeholders. This fosters improved grid monitoring, predictive maintenance, and faster response to outages or demand fluctuations, ultimately leading to a more resilient and efficient power system. Additionally, cloud-based solutions enhance cybersecurity measures, safeguarding critical power infrastructure against emerging threats. Exchanging information between TSOs and DSOs in the cloud has various advantages. Cloud computing refers to the delivery of various computing services, including servers, storage, databases, networking, software, analytics, and intelligence, over the Internet [49]. It allows individuals and organizations to access and use these services without the need to own or manage the underlying physical infrastructure [50].

The following are the key features of Cloud computing:

- Resource pooling: Cloud providers can dynamically allocate and reallocate resources based on demand, which helps to optimize resource usage.

- Rapid elasticity: Cloud resources can be quickly scaled up or down as needed.
- Measured service: Cloud providers can track and report on the usage of resources, which allows for billing based on usage.
- Scalability: Cloud computing allows for near-unlimited scalability, making it easy to handle large amounts of data and a large number of users.
- Availability: Cloud providers offer a high availability of resources, which allows for minimal downtime.
- Security: Cloud providers use various technologies and best practices to help ensure the security of data and systems in the cloud.
- Cost-effective: Cloud computing can help reduce costs by allowing users to pay only for the resources they use, and by eliminating the need for expensive hardware and IT staff.

Cloud computing provides three services which are infrastructure as a service (IaaS), platform as service (PaaS), and software as a service (SaaS) [49]. These services are explained further below.

- Infrastructure as a Service (IaaS)

  This type of cloud computing, known as Infrastructure as a Service (IaaS), provides virtualized computing resources such as servers, storage, and networking over the internet [51]. IaaS providers allow customers to rent computing resources on-demand, on a pay-as-you-go basis, providing flexibility and cost efficiency. Examples of IaaS providers include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).
- Platform as a Service (PaaS)

  This type of cloud computing, known as Platform as a Service (PaaS), allows for the creation, testing, and deployment of applications without the need to handle the underlying infrastructure. PaaS providers offer various services such as databases, web servers, and development environments, which can be utilized to develop and run applications. PaaS providers are particularly useful for developers as they abstract away the need to manage infrastructure, allowing them to focus on building and deploying applications. Examples of PaaS providers include AWS Elastic Beanstalk, Microsoft Azure App Service, and Google App Engine.
- Software as a Service (SaaS)

This type of cloud computing delivers software applications over the internet. SaaS providers offer a variety of applications such as email, customer relationship management, and accounting that can be accessed and used by customers over the internet. Examples of SaaS providers include Salesforce, Microsoft Office 365, Cloudera, and Google G Suite. Figure 3-1 below shows the services of Cloud Computing.

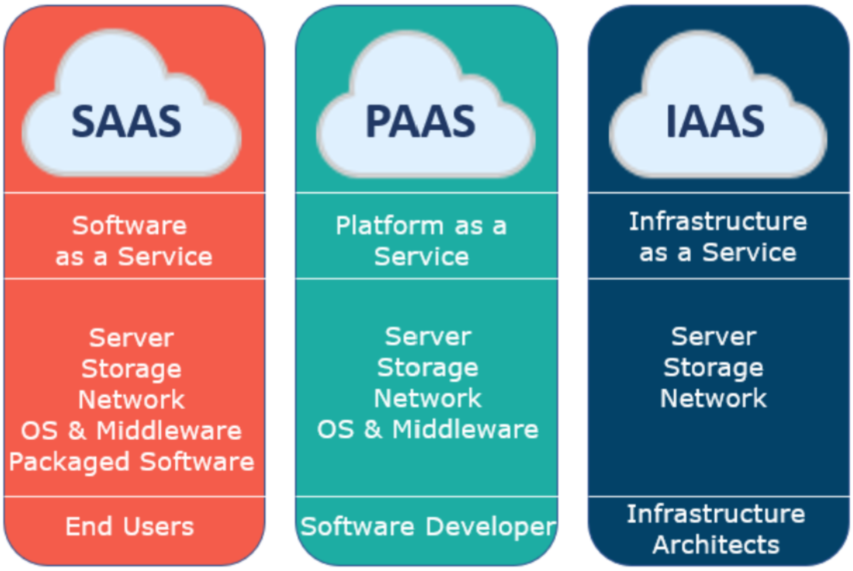


Figure 3-1 Cloud Computing Services [49]

3.2.1 Cloud computing types

There are three types of Cloud Computing. Types of Cloud computing are discussed in this section.

- Private Cloud

Private cloud computing is a type of cloud service that is provided through a private network and is only accessible to a select group of users [50]. It offers organizations greater control, security, and compliance over their data and applications. This type of cloud service allows organizations to keep sensitive data and critical applications on-premises and to tailor the infrastructure to their specific needs. It also allows organizations to manage and maintain the infrastructure in-house, which can be beneficial for organizations with specific security or regulatory requirements. There are several ways to implement a private cloud, such as using virtualization technologies to create a virtualized infrastructure within an organization's data center, using dedicated hardware to build a dedicated infrastructure, or using a combination of on-premises and third-party resources, which is known as a hybrid cloud.

- Public Cloud
  Public cloud computing is a service that is available to the general public and is provided by a third party over the Internet. With public cloud computing, customers can access the scalability, flexibility, and cost-effectiveness of cloud computing without having to invest in and manage their infrastructure. Public cloud providers, such as AWS, Azure, and Google Cloud, offer a wide range of services including computing power, storage, databases, and network services that can be used to build and run applications. The public cloud also allows for easy integration with other services and data sources and can be used for various types of workloads such as testing and development, running production applications, and big data analytics [50].
- Hybrid Cloud
  A hybrid cloud is a combination of Public and Private Clouds. This cloud is managed independently, but the data and applications can be shared among the clouds [49,50]. A hybrid cloud allows organizations to keep sensitive data and critical applications on-premises in a private cloud while leveraging the scalability and cost-effectiveness of public cloud services for non-sensitive workloads. This enables organizations to benefit from the increased security and control of private clouds for sensitive data, while also enjoying cost savings and scalability from public clouds for non-sensitive workloads. A hybrid cloud also allows organizations to move workloads between private and public clouds as their needs change, providing increased flexibility and agility.

3.3 Cloud computing advantages

Following are some advantages of applying cloud computing for exchanging information and data within power systems.

- Scalability and Flexibility:

Cloud services allow users to scale their computing resources up or down as needed. This elasticity is especially beneficial for businesses with fluctuating workloads. You can easily increase resources during peak demand and scale down during quieter periods.

- Cost-Efficiency:

Cloud computing operates on a pay-as-you-go model, where users only pay for the resources they consume. This eliminates the need for upfront capital investments in hardware and software. It also reduces costs associated with maintaining and upgrading infrastructure.

- Accessibility and Ubiquity:

Cloud services are accessible from anywhere with an internet connection. This facilitates remote work, collaboration among geographically dispersed teams, and access to data and applications on various devices (e.g., smartphones, tablets, laptops).

- High Availability and Reliability:

Leading cloud providers, such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), offer robust infrastructure with redundant data centers. This results in high availability and reliability, reducing the risk of downtime and data loss.

- Automatic Updates and Maintenance:

Cloud providers handle server maintenance, software updates, and security patches. This relieves users from the burden of managing these tasks, ensuring that systems are up-to-date and secure.

- Data Security and Compliance:

Cloud providers invest heavily in security measures, including encryption, access controls, and monitoring. Many of them also offer compliance certifications, making it easier for businesses to meet regulatory requirements in various industries.

- Disaster Recovery and Backup:

Cloud services offer robust disaster recovery solutions. Data is often replicated across multiple data centers, ensuring that it can be recovered even in the event of a catastrophic failure. Automated backup options further enhance data protection.

- Resource Pooling:

Cloud computing allows multiple users to share resources, benefiting from economies of scale. This leads to cost savings and efficient resource utilization.

- Innovation and Rapid Development:

Cloud platforms provide access to a wide range of development tools, databases, and other services that can accelerate application development and innovation. Developers can easily experiment and deploy new features without the need to procure hardware.

- Environmental Sustainability:

Cloud providers often have data centers that are more energy-efficient and environmentally friendly than traditional on-premises data centers. By using cloud services, organizations can reduce their carbon footprint.

- Global Reach:

Cloud providers have data centers distributed across the globe. This allows organizations to deploy applications and services closer to their target audience, reducing latency and improving user experience.

- Analytics and Big Data:

Cloud platforms offer powerful analytics and big data processing capabilities. Organizations can harness the scalability and storage options of the cloud to analyze vast amounts of data and gain valuable insights.

- Collaboration and Integration:

Cloud services facilitate collaboration among teams and integration with other cloud-based or on-premises systems. This interoperability can streamline business processes and improve productivity.

3.4 Virtual Machines advantages

3.4.1 Security

VMs have become increasingly popular due to their ability to emulate computer environments, provide isolation for different users, revert to previous states, and support remote launch [52]. These features enhance security by implementing hardware abstraction and isolation, making it more difficult for attackers to gain unauthorized access to data and resources on the physical machine. The ability to restore a VM to a previous state before an attack or data loss occurs improves malware removal and data preservation. Additionally, the capability to remotely start and stop VMs reduces the opportunities for attackers to plan and execute their attacks. VM infrastructure has the potential to be more secure than physical server infrastructure [52]. Hypervisors run outside of the VMs, allowing them to detect malware, which is why VMs are considered a more secure option.

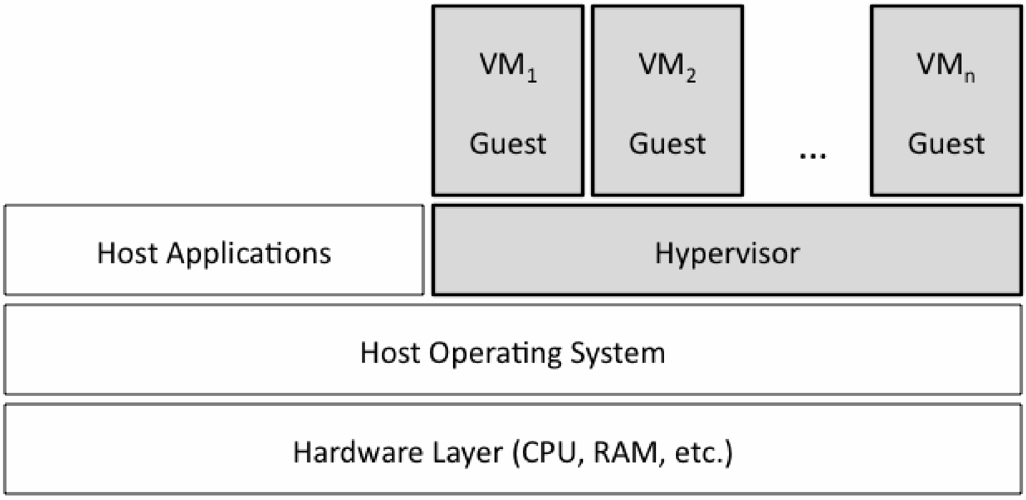


Figure 3-2 Virtual machine architecture [52]

3.4.1.1 Abstraction

Each VM is given its own strictly restricted resources, and VMs abstract the hardware layer. Additional security is offered by this abstraction layer [52]. An attacker has complete control over the computer once they get access to the hardware layer. Operating systems limit hardware access by abstracting away hardware specifics, which explains how the same operating system can run on two separate computers with various hardware setups [53]. In other words, programmers and hackers cannot interact with the hardware since the operating system does. Complete hardware and operating system abstraction are produced by VMs. An application running locally on a physical device is aware of the operating system it is using. However, it's important to note that the host operating system and any running processes within it are not visible to the guest operating system as shown in Figure 3-3. Even though it is operating in a virtualized environment, the guest is unaware of it. This makes it harder for attackers to manipulate and gain control over the machine, as they are not aware of the specific details of the host environment.

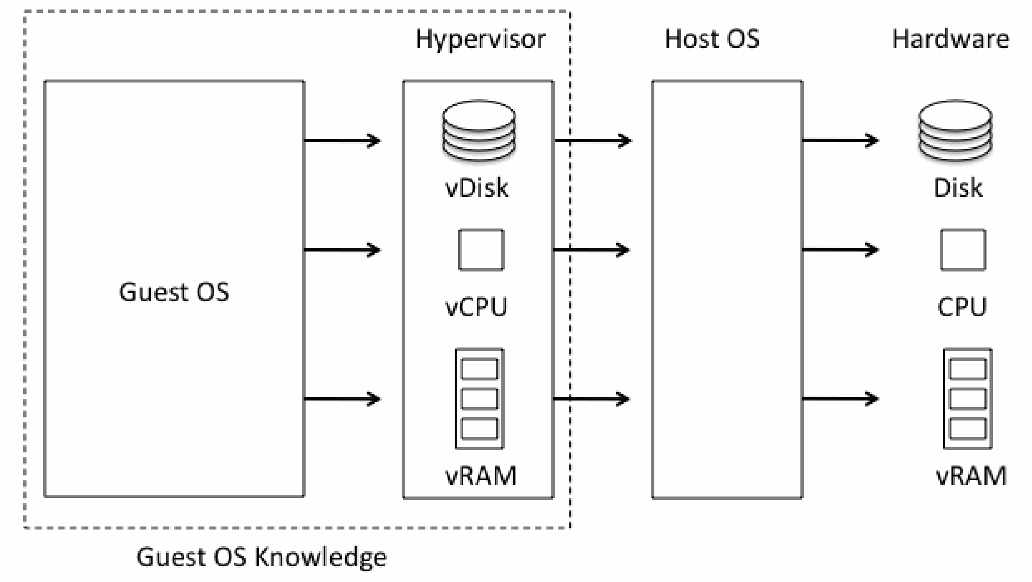


Figure 3-3 Abstraction of physical resources [53]

Because they are so much more straightforward than conventional operating systems, hypervisors are much simpler to secure [53]. In comparison to a microkernel, a hypervisor has a smaller code base and a hardware compatibility layer, which makes it simpler for programmers to reduce flaws and vulnerabilities. The hypervisor's main responsibility is to assign and connect physical resources to each VM. Each virtual computer isolates the visitor and blocks unauthorized guests from accessing resources. This implies that only one virtual computer can be compromised at a time by attackers, not the entire real machine.

3.4.1.2 Isolation

Each guest operating system is given the freedom to function independently thanks to the hypervisors' division of physical resources into separate entities. Any other VMs on the server or the host operating system shouldn't be impacted by an assault on this particular VM [54]. This contrasts with multi-user operating systems, where an attack might potentially impact every user. Each user in a VM infrastructure has access to a particular VM either directly or indirectly. Access to files, including the ability to read, write, or execute them, is determined by the file system within a multi-user operating system. VMs offer more security than regular multi-user computers due to their isolation and abstraction [55]. The hypervisor can delete hacked VMs or restore them to their original condition.

3.4.1.3 State Restore

The capacity of VMs to roll back to a previous state is often praised. Typically, a file on the host is where the contents of each VM's virtual disc are kept. When modifications are made or at regular intervals, the majority of VMs take a snapshot of the virtual disk's contents [56]. State restore is not only convenient, but it also perfectly removes viruses and contributes to data integrity. Services like backup and Windows' System. While physical computers also try to provide the feature of restoring to a previous state, they often fall short as they are unable to combine all system settings and data into a comprehensive state [57]. State restore, which might return unpatched states and cause inconsistencies in the server infrastructure, unfortunately, makes security procedures more difficult.

3.4.1.4 Remote Control

One of the advantages of VMs from a security perspective is the ability to remotely start them, which allows for the machines to be active and accessible when needed. This is different from physical servers which are often left on all the time, even when not in use. Limiting the amount of time, a computer is online is an effective defense against potential attacks [58]. For example, if a virus affects one computer on a network, only online VMs will be affected. Keeping VMs in use and closely monitored, as they can be called up as needed, increases the likelihood of detecting an intrusion. This is because a person is more likely to notice an intrusion when actively using a computer than when it is not in use.

3.4.1.5 External Monitoring

Because VMs only use a portion of the hardware resources available, it is easy to monitor resource utilization and identify malicious software from outside the VM. Operating systems are typically physically installed with virus protection. However, advanced attackers have found various ways to circumvent virus protection, giving them access to an unprotected operating system. However, the hypervisor or a specially approved VM that can see software activities can keep an eye on VMs. The latter approach is recommended since it restricts the hypervisor's function, keeping it as straightforward and secure as feasible. The dedicated VM is only given access to the resources allotted to the monitored VM by the hypervisor. A single dedicated VM that is only used to keep an eye on other running VMs is seen in Figure 3-4. These monitors are employed, among other things, in forensic analysis, integrity checking, honeypot systems, and intrusion detection systems (IDSs) [58]. To prevent malicious code from gaining access beyond the VM, the VM can be turned off or shut down if an attack is detected by the monitoring system and hypervisor [59] if it is being watched by an external process. The paradox with physical machines is that they cannot accurately tell if they contain a virus if they do. VMs, however, do not experience this issue.

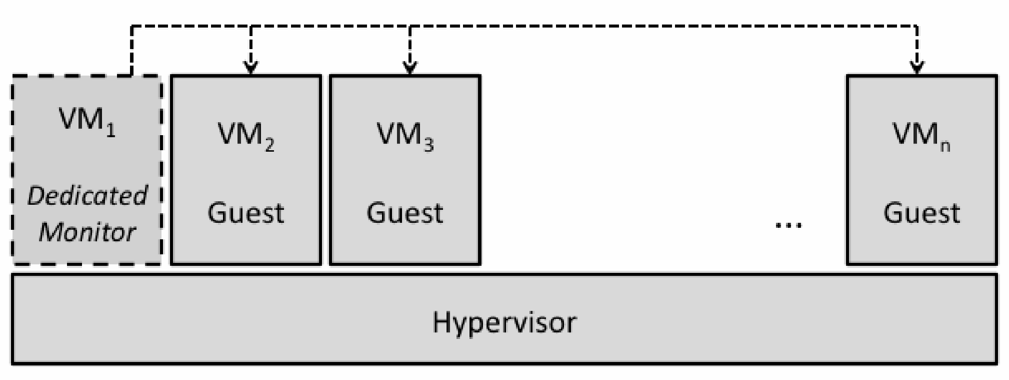


Figure 3-4 Dedicated external monitor [59]

3.4.2 Scalability

When it comes to scalability in the area of cloud computing, there are generally two types of scalability that are presented below:

Vertical scalability: It is the ability to add more resources to the same server or hardware to make it bigger or do more. For example, you could add more processing power to a server to make it go faster [60]. It can be done by adding more hardware to the same thing, like hard drives, servers, CUs, and so on. It gives the operating system and applications more resources that they can use together [61]. This type of scalability may also be referred to as scaling up or scaling in. Vertical scaling is thought to be the easier way to grow because servers on cloud platforms like AWS are already virtualized, making it easy to add new hardware. Figure 3-5 shows how the number of servers in a cluster can grow vertically.

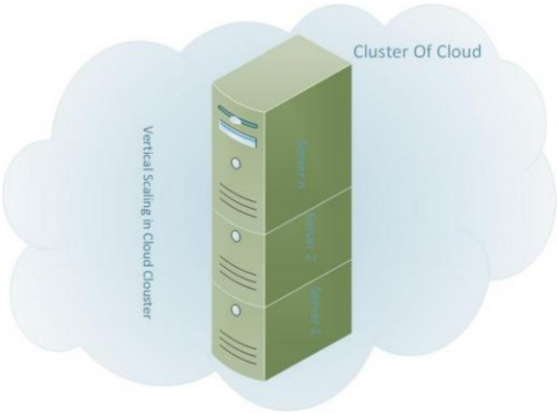


Figure 3-5 Vertical scaling of servers in a cluster [61]

Horizontal scalability: Multiple pieces of hardware or software, like servers or networks, can be connected to the system or resources so that they work as one logical unit [62]. It means adding more resources that all do the same job. In the case of servers, for example, it could add more servers as needed to make the logical unit faster or more available. One can have two, ten, or more of the same server doing the same work instead of just one. It is also called scaling up or scaling down. However horizontal scaling makes the system harder to understand. Not only do updates, security, and monitoring need to be done on multiple servers, but applications, data, and backups also need to be in sync across many instances [63]. Figure 3-6 shows how servers in a cluster can grow horizontally.

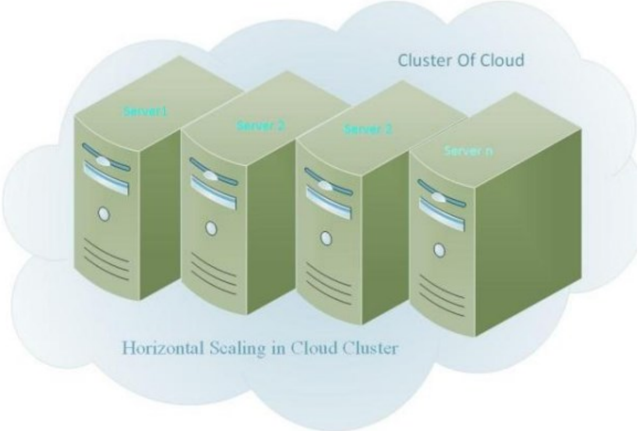


Figure 3-6 Horizontal scaling of servers in a cluster [63]

In general, a cluster of VMs, employed to develop a data exchange cloud platform, has the following scalability advantages, which are data storage, computing power, versatility, time saving, and cost-saving respectively.

3.4.2.3 Versatility

Cloud computing enables dynamic reconfiguration of infrastructure and workloads to fit current needs, without being tied to past equipment and assets, as they no longer need to be maintained. This makes it possible to create private networks that function as hybrid clouds or multi-cloud deployments, addressing existing issues and concerns. With cloud computing, it is possible to make real-time adjustments to ensure that the IT infrastructure meets the current requirements. This flexibility and scalability provided by cloud computing make it a valuable solution for organizations looking to manage and optimize their IT infrastructure [66].

3.4.2.1 Data Storage

Cloud services provide a variety of services, storage is one of the most important. Having sufficient storage space is crucial for storing important files, hosting applications, and safeguarding valuable customer data. Cloud computing allows for scalable data storage to meet availability needs without incurring the capital costs associated with expanding physical infrastructure. This eliminates the need to manage a constantly growing collection of hard drives and allows organizations to easily adapt to changing storage needs.

3.4.2.2 Computing Power

Cloud computing has transformed the way IT infrastructure is managed by providing easy access to high-performance computing resources. This technology allows for utilizing powerful software development tools, analyzing data through advanced analytics programs, and creating products and services that drive significant business outcomes. The cloud environment enables organizations to scale computing power as per their needs, whether it be handling temporary traffic spikes or increasing capacity for permanent increases in workloads. This flexibility and scalability make cloud computing a cost-effective and efficient way to manage IT resources and adapt to changing business requirements. It is a crucial solution for organizations looking to optimize their IT infrastructure and stay competitive in the market [65].

3.4.2.4 Time Saving

Managing modern infrastructure can be challenging and time-consuming, often diverting resources away from innovation. Utilizing cloud computing and data centers for computing solutions instead of on-premises alternatives can free up technical resources for more innovative projects, rather than dedicating them to troubleshooting and management [67].

3.4.2.5 Cost Saving

The use of cloud computing can result in cost savings for both equipment and power and cooling. The ability to adjust resources as needed also reduces risk. Additionally, many applications may run more efficiently in the cloud and can be easily migrated. These cost savings can then be used to support further business growth [67].

3.5 Blockchain in Information and Data Exchange

Before the emergence of blockchain technology, several key innovations paved the way for its development. One of the earliest precursors was cryptographic hashing, which allowed for the creation of a unique digital fingerprint for data. This enabled secure authentication and verification of information. Additionally, concepts like public-key cryptography, introduced by Whitfield Diffie and Martin Hellman in the 1970s, provided a foundation for secure digital communication. The idea of distributed computing and consensus algorithms, exemplified by the Byzantine Generals' Problem and solutions proposed by researchers like Leslie Lamport, also contributed to the development of blockchain. Furthermore, in the late 1990s and early 2000s, projects like Hashcash and b-money introduced elements of proof-of-work and decentralized digital currency, foreshadowing the core principles of blockchain. These precursors collectively laid the groundwork for the eventual creation of blockchain technology and its revolutionary impact on various industries, including power systems and data exchange. Blockchain is a digital ledger technology that is distributed, immutable, and secured using cryptography. It is used to record transactions across a network of computers. A blockchain is a digital record that is made up of a sequence of blocks, each containing a set of transactions. These blocks are linked together using cryptographic hash functions, forming a chain. This creates an immutable and unchangeable record of transactions, which is why it's called a "blockchain". The most popular blockchain platforms are Bitcoin and Ethereum. Transactions on a blockchain are grouped, recorded permanently in blocks, and linked in a chronological and linear order to form a blockchain. Every block in the blockchain is identified by a hash of its block header. The method of creating blocks and appending them to the blockchain is utilized to monitor the entire chain of network activity, commencing with the first block in the chain [68]. Figure 3-7 illustrates the basic blockchain structure.

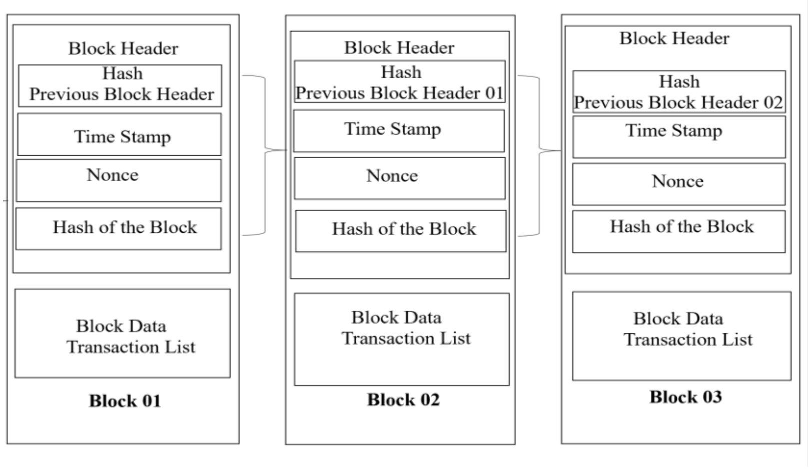


Figure 3-7 A basic blockchain structure [68]

3.5.1 Types of Blockchain

There are different types of blockchains based on the data they manage, the accessibility of that data, and the actions that users can perform. Bitcoin introduced public blockchains, where nodes are not trusted. There are two types of blockchains: public or permissionless and private or permissioned. Public blockchains are accessible to anyone and have an anonymous identity, and private blockchains are closed networks with a verified identity and restricted access. Public blockchains can also increase computational power as the block size and data size grow, making them highly decentralized and scalable [69]. Most public blockchains, like Bitcoin and Ethereum, currently use a type of Proof of Work (PoW) consensus mechanism. PoW is a consensus mechanism that relies on computational power to validate transactions and add them to the blockchain. It is a decentralized process where a group of users called "miners" compete to validate transactions by solving complex mathematical problems. Once a miner solves the problem, they broadcast the solution to the network, and if other miners agree that the solution is valid, the block is added to the blockchain, and the miner is rewarded with cryptocurrency. PoW is a secure and robust mechanism, but it's also energy-intensive, which can make it less efficient and more costly. Although they function similarly to public blockchains, permissioned or private blockchains require user authentication before they allow them to join. For submitting transactions, reading transactions, and taking part in the consensus mechanism, users in permissioned blockchains may have varying levels of access. Permissioned blockchains can use lighter consensus methods due to the initial user filtering which accelerates transaction processing [70]. One well-known permissioned blockchain example is Hyperledger Fabric [71].

3.5.2 Blockchain platforms

The first use of the blockchain was in Bitcoin [72]. It was primarily created to control and send Bitcoin (its cryptocurrency). The advent of Ethereum [73] brought with it the notion of smart contracts, which greatly expanded the potential of blockchain technology and opened up a range of new applications and opportunities for development. This led to the emergence of a new generation of blockchain platforms and an abundance of use cases. Decentralized applications and smart contracts are being built on various blockchain platforms today. These platforms have different features and capabilities, but they all use the core blockchain technology. to anyone in the world. They support integrated cryptocurrencies and allow for a trustless and decentralized system. However, they can have scalability issues and may not provide the same level of privacy as private blockchains. Permissioned blockchains are designed for specific use cases and have a defined set of participants. They may not have integrated cryptocurrencies and may require permission to join and participate in the network. They are more suitable for use cases where a high level of privacy and security is required such as Hyperledger Fabric, Corda, and Quorum.

In terms of tools and developer community, Ethereum and Hyperledger Fabric have well-developed tools and strong developer communities, while others may provide minimal support for users and developers. It's also important to consider the network's performance, privacy, cost, and maturity when choosing a blockchain platform.

Table 3-1 compares various platforms. In this thesis, the Hyperledger Fabric platform is used for exchanging information and data. The most developed permissioned blockchain that is available is Hyperledger Fabric [71], and it is a perfect fit for exchanging information and data within the power system. The fundamental ideas behind these two systems are discussed in the paragraphs that follow.

Table 3-1 Blockchain platforms

Platforms	**Public or permission**	**Cryptocurrency**	**Smart contract**	**Consensus algorithm**
Bitcoin	Public	Yes	Ivy	PoW
Ethereum	Public	Yes	Solidity	PoW and PoS
Hyperledger Fabric	Permissioned	No	Java, Go, Node.js	PBFT
Corda [74]	Permissioned	No	Kotlin	Pluggable
Quorum [75]	Permissioned	No	Solidity	Raft

3.5.3 Hyperledger fabric

The Linux Foundation hosts the permissioned blockchain known as Hyperledger Fabric [71]. Hyperledger Fabric's modular architecture allows for the plugging-in of various components such as consensus and databases. The membership layer can authenticate users and grant access based on their level of access and system policy [76]. Hyperledger Fabric is particularly advanced in terms of permissions, as it allows for granular access control. For example, it can be configured to allow only certain users to submit transactions, execute specific smart contracts, and view the ledger state. This level of granularity allows organizations to control access to sensitive information and ensures that only authorized parties can access and make changes to the network.

Chain code is a concept introduced by Hyperledger Fabric. Chain code is a program that executes throughout the execution phase and implements the application logic

[77]. The phrases "smart contract" and "chain code" are used synonymously in Hyperledger Fabric. The transaction logic is, however, represented by a smart contract, which is subsequently packaged as chain code and uploaded on the Fabric blockchain network [78]. Each smart contract in a chain code has a set of transaction specifications as its foundation, and a chain code may contain one or more smart contracts. All of the smart contracts contained in a chain code are made accessible to applications when it is deployed to the network.

3.5.4 Hyperledger Fabric transaction validation and endorsement

In Hyperledger Fabric, transactions are validated and endorsed by a specific subset of network participants called "endorsers." When a client submits a transaction, the endorsers receive the transaction and validate it according to the smart contract's business logic and the current state of the ledger. If the transaction is valid, the endorsers will "endorse" it by signing it with their private keys. These endorsements are then sent back to the client, who can then broadcast the transaction to the rest of the network for ordering and commit [79].

Once the transaction is committed, it becomes a part of the ledger and can be queried by other participants. The endorsement process plays a crucial role in ensuring the integrity and accuracy of the ledger by validating transactions before they are added to the blockchain. This process certifies that only legitimate transactions are recorded on the ledger. Additionally, it allows for multiple levels of access control, as different endorsers can be assigned to different subsets of the network, and certain transactions can be restricted to certain endorsers [80].

When compared to other blockchain systems, such as Ethereum or Bitcoin, Hyperledger Fabric stands out because it validates transactions with the help of endorsement policies. Any node on the network is capable of submitting genuine transactions in such kinds of systems [81]. On the other hand, transactions in Hyperledger Fabric have to be confirmed by trustworthy entities within the network, which makes it more realistically simulate situations that occur in the real world. Hyperledger Fabric also offers key-level or state-based endorsement policies. This policy enables more granular policies that are recorded on the ledger. Because of this, it is possible to configure a one-of-a-kind endorsement policy for each key-value item

[82]. Endorsing peer nodes must verify the transaction before signing them. This is done to ensure that there have been no intermediate steps. A transaction is considered legitimate if it passes both of these checks successfully.

3.5.5 Smart contracts

Smart contracts are computer programs that run on a blockchain network and are used to automate the execution of certain actions or transactions. With the advancement of blockchain platforms that support programming languages, it has become possible to develop more complex smart contracts that can be applied in a wide range of situations. The blockchain ensures that smart contracts adhere to their terms and conditions. The use of blockchain technology ensures that once a contract is executed, it cannot be altered and all parties have access to the same information. This eliminates the need for intermediaries, such as lawyers or banks, to verify or enforce the contract. Smart contracts have the potential to transform the way we conduct business and interact with each other. They can automate complex processes, reduce the need for intermediaries, and increase transparency and security [83].

Several blockchain platforms use smart contracts, some of the most popular ones include:

Ethereum: Ethereum is the most widely recognized platform for creating and executing smart contracts. It employs its programming language, Solidity, to compose and implement smart contracts.

NEO: NEO is a blockchain platform that uses smart contracts to build decentralized applications. It uses a unique consensus algorithm called Delegated Byzantine Fault Tolerance (dBFT) to process transactions.

TRON: TRON is a blockchain platform that uses smart contracts to build decentralized applications. Delegated Proof of Stake (DPoS) is used to process transactions.

Cardano: Cardano is a blockchain platform that uses smart contracts to build decentralized applications. It uses a unique consensus algorithm called Ouroboros to process transactions.

Tezos: Tezos is a blockchain platform that uses smart contracts to build decentralized applications. It uses a unique consensus algorithm called formal verification to process transactions.

Hyperledger Fabric: Hyperledger Fabric is an open-source blockchain platform that uses smart contracts to build decentralized applications. It is designed for enterprise use cases and supports multiple programming languages.

These are just a few examples of the many blockchain platforms that use smart contracts. Each platform has its unique features and capabilities and is suitable for different types of projects and use cases.

3.5.6 Consensus algorithm

The validation of transactions in blocks is completed by the consensus mechanism, which also agrees with all peers. Several alternative consensus techniques vary in processing power, performance, scalability, and ability to tolerate disruptive behaviors. In their study of blockchain consensus algorithms, Nguyen and Kim [84] carried out a study on consensus methods. The consensus procedures are divided into two groups: proof-based and voting-based. In the proof-based approach, a leader (or leaders) is chosen who is in charge of authenticating and affixing blocks to the ledger.

Proof of Work (PoW) [85] was first proposed by S. Nakamoto as a consensus mechanism for Bitcoin and later adopted by Ethereum. In the incentive-based PoW algorithm, miner nodes must solve a complex mathematical problem to earn rewards. This process is similar to repeatedly guessing a value, called a nonce until the problem is solved. The first miner to solve the problem is the winner and gets to create the next block [86]. This block is then broadcast to other nodes for validation. The validation process is relatively simple, and the group of transactions is added as a new block to the blockchain once the proposed block is confirmed to be valid.

To make PoW less expensive, Peercoin [87] first incorporated proof of stake (PoS). The foundation of PoS is the demonstration of Bitcoin ownership. The network selected a miner for each round based on the stake amounts of the nodes. Wealthier nodes have a greater likelihood of being chosen. After other nodes have validated the block, the miner is rewarded for suggesting the correct block. PoS decreases computation, but it has the potential drawback of making rich nodes richer over time.

It implies that richer nodes would have a higher chance of being chosen every time [87].

Byzantine Fault Tolerance (BFT) is a consensus mechanism used to ensure that a distributed system can function properly even when some of its components fail or act maliciously. The term "Byzantine" refers to the Byzantine Generals Problem, which describes a scenario in which multiple generals, each commanding a portion of the Byzantine army, must reach a consensus on a strategy for attacking a city. In a distributed system, nodes may experience failures or act maliciously, which can lead to conflicting information. The BFT consensus mechanism addresses this problem by allowing nodes to reach a consensus through a process of communication and voting. There are several BFT algorithms such as Practical Byzantine Fault Tolerance (PBFT) [88], Federated Byzantine Agreement (FBA), and Delegated Byzantine Fault Tolerance (DBFT). These algorithms differ in terms of their communication patterns, the number of rounds of communication required, and the number of faulty nodes that the system can tolerate. Figure 3-8 shows the three phases in PBFT [89].

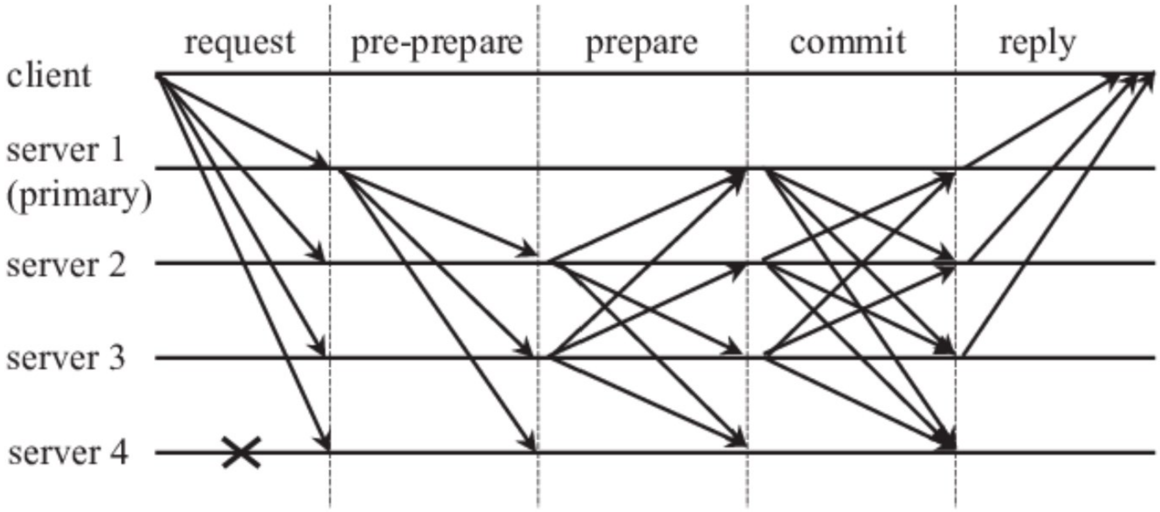


Figure 3-8 PBFT Operation [89]

checksums for file integrity verification, all within the context of application-level interactions and services.

3.5.7 Open Systems Interconnection Model

The OSI (Open Systems Interconnection) model is a conceptual framework used to understand and describe how different networking protocols and technologies interact with each other to enable communication between devices on a network. It consists of seven layers, each responsible for specific functions in the communication process. Let's identify where the technology described earlier (using blockchain and checksums for file validation) sits within the OSI model:

Physical Layer: The Physical Layer deals with the physical transmission of data over the network medium, such as cables, switches, and network interfaces. The technology we described does not directly interact with the Physical Layer, as it operates at higher layers of the OSI model.

Data Link Layer: The Data Link Layer is responsible for establishing and maintaining a link between two directly connected nodes on the same network. It ensures reliable data transfer over the physical link. The technology we described does not directly interact with the Data Link Layer either.

Network Layer: The Network Layer is responsible for routing data packets between different networks, enabling communication between devices on different subnets or networks. The technology we described does not have a direct association with the Network Layer since it operates at a higher layer.

Transport Layer: The Transport Layer is responsible for end-to-end communication, ensuring reliable and error-free data transfer between applications running on different devices. The technology we described does not reside at the Transport Layer either.

Session Layer: The Session Layer is responsible for establishing, managing, and terminating sessions between applications on different devices. The technology we described does not have any specific interaction with the Session Layer.

Presentation Layer: The Presentation Layer is responsible for data representation, ensuring that data from different systems can be interpreted and presented in a compatible format. The technology we described does not operate at the Presentation Layer.

Application Layer: The Application Layer is the highest in the OSI model, and it directly interacts with the end-user applications. This layer includes protocols and services that end-users interact with directly. The technology described, which uses blockchain and checksums for file validation, primarily resides at the Application Layer.

Specifically, the technology uses the blockchain as a decentralized and immutable ledger to record file upload transactions and metadata. This ledger is an application-level construct that ensures transaction transparency and integrity. Additionally, the validation of the file's authenticity through checksums (cryptographic hashes) also occurs at the Application Layer. These checksums are calculated and verified as part of the application's logic and are used to ensure the integrity of the file data.

In summary, the technology utilizing blockchain and checksums for file validation primarily sits at the OSI model's Application Layer. It leverages the decentralized and immutable properties of the blockchain for recording transactions and employs

3.5.8 Blockchain in power system

Blockchain technology is a complex system that allows individuals to verify information and conduct transactions directly with one another in a trustless environment. Three essential properties of blockchain are security, transparency, and immutability [90]. These properties make blockchain technology unique and useful for various applications in the energy and power sectors, such as creating transparent and secure supply chains, enabling peer-to-peer energy trading, and providing secure and tamper-proof metering and billing systems. The absence of a third party can lead to cost, operational, and market efficiencies. Blockchain technology has the potential to be used in nearly any market, and many companies in the energy sector are exploring its use for grid-level transactions, peer-to-peer trading, energy financing, electric car charging, and tracking of RES. This can help to create more efficient, transparent, and secure systems for managing energy resources [91].

There is an urgent need for a more transparent and decentralized system. Such critical infrastructure requires a platform based on decentralized control while upholding tight reliability and security standards. A system that, instead of the other way around, adjusts to and supports new technologies and usage patterns. Because there are so many producing and/or consuming organizations in the distribution grid, blockchains can be utilized to enhance supply and demand balance, automated grid asset verification, increase the visibility of distributed resources and assets, and better TSO and DSO coordination [92]. As data is communicated immutably rather than being transmitted, blockchains can simplify the process by eliminating intermediaries, reducing delays, and guaranteeing data integrity.

3.6 Issues with Blockchain

Several common challenges and concerns are associated with blockchain technology. Let me provide some insights into each of these issues:

Scalability Issues: Scalability is a significant challenge in public blockchains like Bitcoin and Ethereum. As more users join the network, the transaction processing capacity can become limited, leading to slow confirmation times and high fees. Solutions like sharding and layer 2 scaling solutions (e.g., Lightning Network for Bitcoin and Ethereum's Layer 2 solutions) are being developed to address these problems.

Energy Consumption and Environmental Concerns: Public blockchains that use Proof of Work (PoW) consensus mechanisms, like Bitcoin and Ethereum, require substantial computational power and, as a result, consume a significant amount of energy. This has raised concerns about their environmental impact. Some blockchains are transitioning to more energy-efficient consensus mechanisms like Proof of Stake (PoS) to mitigate these concerns.

Lack of Privacy: Many public blockchains are designed to be transparent, making all transaction data visible to anyone. This lack of privacy can be a challenge, especially for use cases where confidentiality is crucial. Projects like Monero and Zcash aim to address this issue by implementing privacy-enhancing technologies.

Regulatory and Legal Challenges: Blockchain and cryptocurrency regulations vary widely from country to country. Regulatory uncertainty can create challenges for blockchain projects and users. Compliance with anti-money laundering (AML) and know-your-customer (KYC) requirements is a common concern, as are tax implications and securities regulations for token offerings.

User Experience and Accessibility Issues: Blockchain technology can be complex for non-technical users, which can hinder its adoption. Wallet management, key security, and the need to understand gas fees (in Ethereum) are some of the usability challenges. Improving user interfaces and educational resources can help address these issues.

Smart Contract Vulnerabilities and Security Risks: Smart contracts are code-based and can be vulnerable to bugs and vulnerabilities. High-profile incidents, like the DAO hack in Ethereum, have highlighted the importance of robust security practices in blockchain development. Ongoing code audits, formal verification, and best coding practices are used to enhance smart contract security.

As the technology matures and evolves, solutions to these issues are likely to emerge, making blockchain more efficient, secure, and accessible for a broader range of use cases.

3.6 Chapter Summary

In this chapter two ICT technologies in terms of exchanging information and data are critically reviewed. These two technologies are implemented in the thesis to evaluate the best possible for future data exchange within the power system. These technologies have been evaluated based on multiple performance metrics.

Chapter 4 Cloud Data Exchange Platform

4.1 Introduction

The use of cloud computing techniques for exchanging information and data between TSOs and DSOs has numerous benefits, making it a vital area of research for future smart grid systems. The proposed method in this research utilizes a cloud computing platform to enable the exchange of information and data between TSOs and DSOs, highlighting the advantages of using such a platform [93,94]. Many researchers have explored ways to use cloud computing to improve the efficiency, reliability, and scalability of power systems. Different cloud-based solutions, such as virtualization, distributed computing, and edge computing, address various challenges in power systems, such as integration of RES, demand response, and real-time monitoring and control [95,96]. Cloud computing allows for the distribution of work across the computational network, reducing the load on local computers. The main objective is to provide an ICT infrastructure that is scalable and cost-efficient. In this thesis, Cloudera has been selected as the data exchange platform because it can effectively and efficiently manage large quantities of data utilizing Apache Hadoop.

Before selecting Cloudera for information and data exchange. Three different cloud platforms were investigated namely; Cloudera, Amazon AWS, and Microsoft Azure for exchanging information and data between TSOs, DSOs, and other actors or participants. A range of Use Cases were implemented on three different platforms with different volumes of data to evaluate the performance of these platforms. All the requested data for data exchange is in XML format. For data set 1, the size of data used for Use Case 1, Use Case 2, and Use Case 3 was 5MB, 8MB, and 2MB respectively. For data set 2, the size of data used was 7MB, 10MB, and 4 MB. For data set 3 the size of data used was 9MB, 12MB, and 6MB. The reason for using three different datasets is the evaluate different data exchange platforms while increasing the volume of data. The datasets used in these demonstrations were both realistic and simulated available by the TDX-ASSIST project. The results have shown that the TSO, DSO, and other grid actors can exchange the requested data through these platforms successfully. The subsequent graphs depict the results obtained from these experiments.

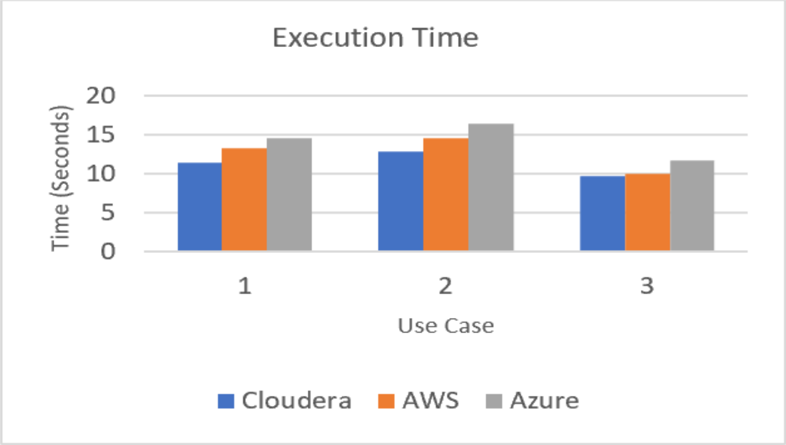


Figure 4-1 Execution time data set 1

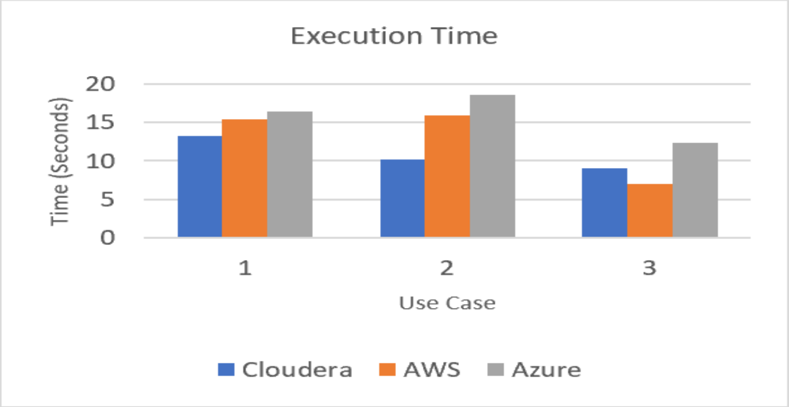


Figure 4-2 Execution time comparison data set 2

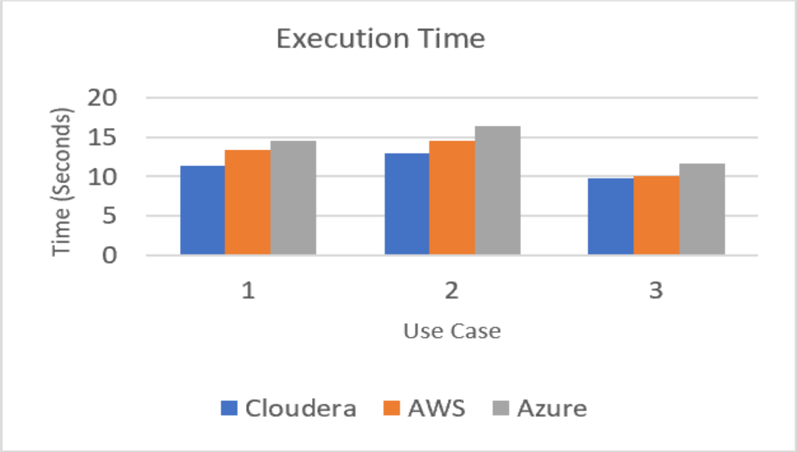


Figure 4-3 Execution time comparison data set 3

It is evident from figures, that the service execution time of Cloudera is less as compared to other platforms. It is mainly because in Cloudera actions are distributed on the virtual machine cluster. Cloudera has already deployed the Hadoop ecosystem and hence has a really strong performance in handling complex data. The execution time of Use Case 2 is higher in all three platforms because there are more actions required for the data to be exchanged as explained in the previous section. The execution time of Use Case 3 for Amazon AWS and Cloudera is almost similar. The execution time of the Microsoft Azure data platform is higher in all three cases as compared to other data exchange platforms mainly because of the location of the data center.

4.2 Introduction to the Cloudera data platform

Cloudera offers a comprehensive platform that makes it simple to handle growing volumes and types of data in your enterprise. Its products and solutions allow for the deployment and management of Apache Hadoop and related projects and enable the manipulation and analysis of data while ensuring it remains secure and protected [97]. Cloudera Data Platform is capable of managing data in any scenario. The Cloudera Platform also lets system administrators protect a cluster by encrypting data, verifying users, and giving permissions. Figure 4-4 illustrates the overview of the Cloudera data platform in figure [96].

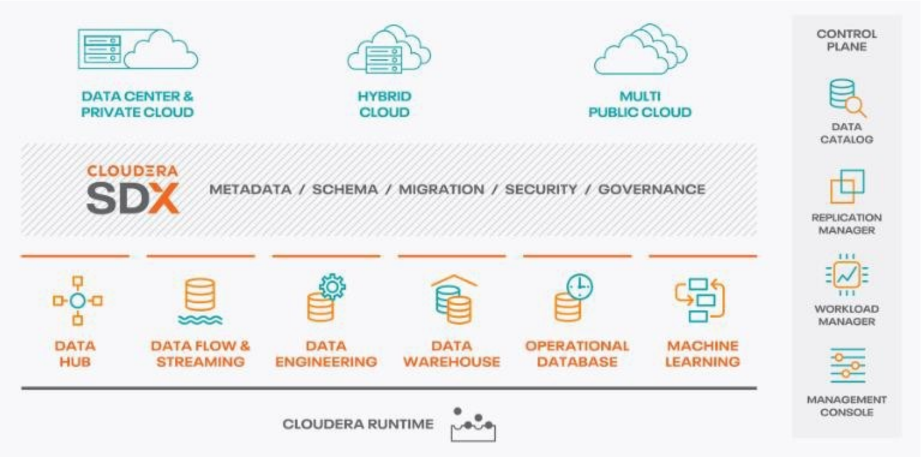


Figure 4-4 Overview of Cloudera data platform [96]

4.2.1 Cloudera data platform functionality

Cloudera data platform consists of four parts, Data science and engineering, Data warehouse, Data science workbench, and operational Database [98]. These parts are briefly explained below.

- Data Science and Engineering

  Cloudera Data Science and Engineering is capable of performing advanced data engineering and machine learning at a scale regardless of the location of data. It can process all types of data from all sources [96,98]. High-performance machine learning and continuous data processing streams are highly supported. Modern predictive analysis can be performed using Cloudera Data Science and Engineering as it gives better access to Apache Hadoop data. To ensure stable models Cloudera Data Science and Engineering provides a high-performance programming interface with modern liabilities.
- Data Wearhouse

  Data Wearhouse provides an enterprise solution to modern analytics. It is capable of analyzing all sorts of data, unstructured, semi-structured, machine-generated, and traditional data sources in a single Data warehouse environment. It is an auto-scaling, highly synchronized, and cost-effective hybrid solution. It provides a safe and secure solution with almost zero time with reduced IT cost [96,98]. In addition to that, for easier experimentation machine learning techniques and algorithms are present.
- Data science workbench

  The data Scientist can manage analytics pipelines using the Cloudera data Science workbench. It is a safe and self-service data Science platform. Existing tools and techniques, such as Python and Scala can be used in the platform for the data science team to run experiments. Any library can be installed within an isolated project environment and has direct access to secure clusters using Spark and Impala. Data pipelines can easily be monitored using built-in job secluding. It gives the leverage to share results with the whole team [96,98].
- Operational database

  It is an open-source platform with technologies such as Apache HBase, Apache Kudu, and Apache Spark. It can extract real-time inside of big data with high- concurrency and security. Security and governance are the core of this platform, which helps secure data and fulfills the needs of the industry. It can compare real-time data with historical data for better analysis of future events. An operational Database can process real-time data with continuously changing data for better decision-making [96,98].

4.2.2 Cloudera platform security

Data encryption and user authorization techniques are used by the system administrator to secure the cluster in the Cloudera platform. As this platform is designed to deal with large amounts and types of data, Cloudera clusters meet evolving security requirements that are imposed by industries, government, and regulating agencies [99]. Lightweight Directory Access Protocol (LDAP) and Kerberos are used by Cloudera clusters for the authentication process. Kerberos provides a strong authentication mechanism. It uses cryptographic mechanisms rather than using a password alone.

To protect data Cloudera provides an encryption mechanism. It is a process that uses digital keys to encode various components so that only the concerned users can decode and view the items. In Cloudera when this protected data is persisted on the storage devices it is called HDFS encryption. And when this data moves on the network it is called SSL encryption. Protecting the data specifically means encrypting the data when it is stored on the device and decrypting the data only by the authorized person when it is needed [100]. Figure 4-5 below explains the encryption mechanism of Cloudera.

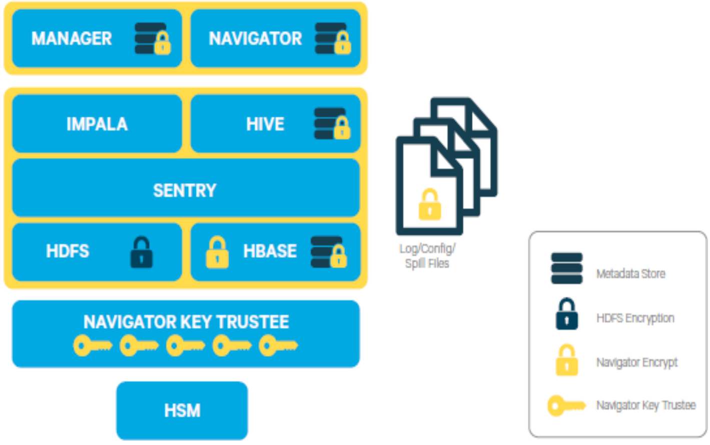


Figure 4-5 Encryption mechanism in Cloudera [100]

4.3 Architecture design of the platform

Cloud computing is a technology that allows for on-demand access to computer system resources, such as data storage and computing power, without the need for direct management by the user [101]. The technology that enables cloud computing is virtualization, which enables the separation of a physical computing device into one or more virtual devices. These virtual devices can be easily managed and used to perform computing tasks. Virtualization also improves efficiency by allowing for the more effective allocation and utilization of idle computing resources. Additionally, it helps to speed up IT operations and reduce costs by increasing infrastructure utilization.

Cloud computing is utilized to design data exchange platforms, with an emphasis on the design of service deployment and the HDFS file system. The architectural structure of the proposed data exchange cloud platform is illustrated in Figure 4-6. In this architecture, different entities of the power system will have access to the defined use case services through the Cloudera. A cluster of VMs is created, consisting of multiple VMs created on physical computers, and acts as the backend to support the platform. An HDFS file system is established on top of the VM cluster to enable data exchange among different actors within the services. The HDFS file system provides scalability, robustness, and resilience for actor interactions in terms of data exchange, storage, and extendable interfaces. These features are further discussed in the following section.

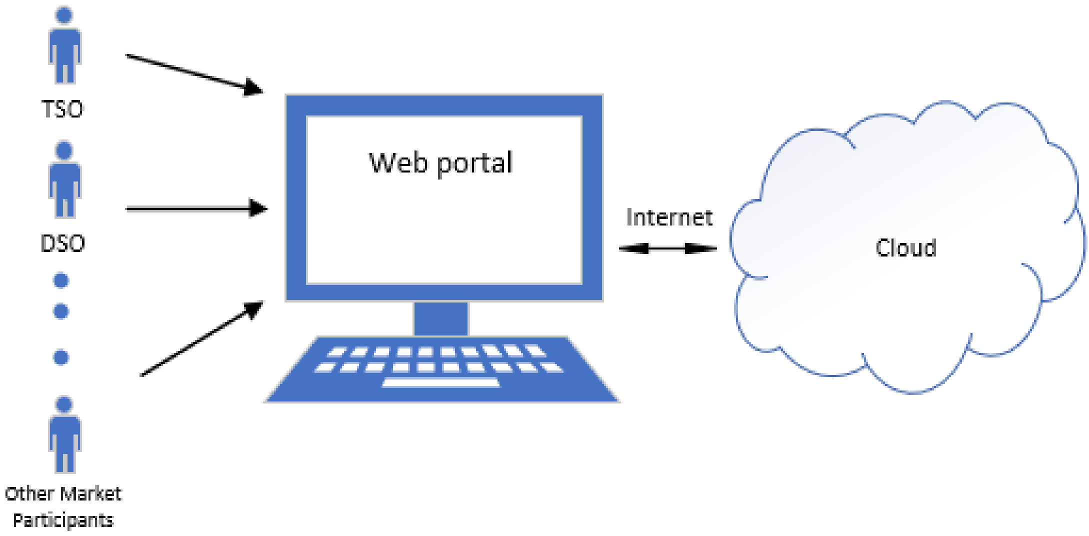


Figure 4-6 Data exchange cloud platform system design

A service-oriented strategy is implemented to effectively fulfill the service needs of various VMs and manage resources. Assigning different services to separate VMs is crucial for secure operation, manageable management, and enhanced system efficiency. An example of how services like Service 1 and Service 2 can be deployed on the data exchange platform is demonstrated in Figure 4-5. Service 1, which only requires one method and relatively fewer system resources, can be installed on VMs 1 and 2. Service 2, which includes more methods and requires more system resources, can be deployed on VMs 2, 3, and 4. By distributing services across multiple VMs, the services can still be accessed reliably even if one of the VMs fails. The architecture is structured with a service-based approach, where all activities are performed. Each virtual node can be controlled directly by the user through the use of a Cloudera manager, leading to increased flexibility and improved resource allocation. The service-based approach used to construct the VM cluster guarantees high modularity and interoperability. The allocation of services on the VM cluster is illustrated in Figure 4-7.

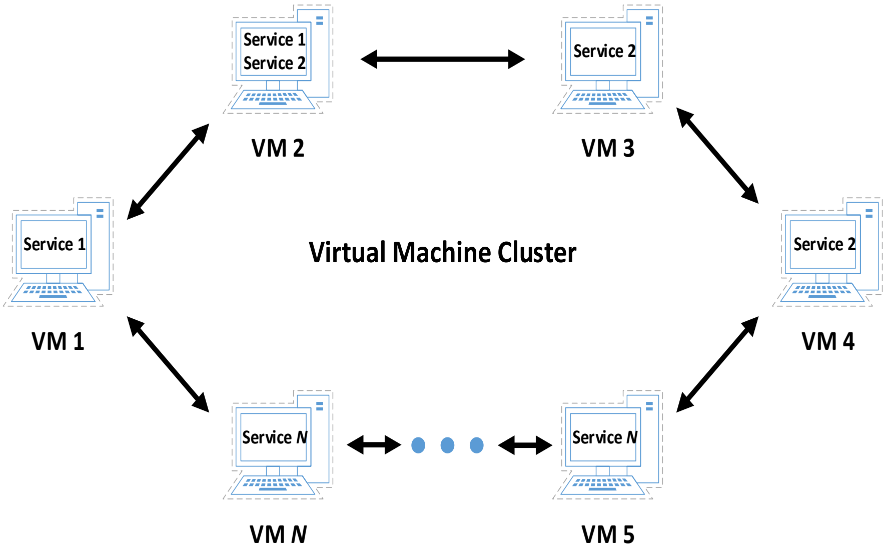


Figure 4-7 Data exchange platform services deployment

4.3.1 Experimental setup

The thorough experimental setup for the demonstration is shown in Table 4-1. In the presentation, CIM/XML (Common Information Model)-formatted real and simulated data are both used. Python is a programming language that is used to code the process by which different entities of power systems exchange information and data.

Table 4-1 Experimental Setup tools

Platform Software	Cloudera
Storage database	HDFS
Programming Language	Python
No of VMs	3-120
Disk Size	64 GB
Memory	8 GB
Processors	2
Data Format	CIM/XML
ETL Tool	Apache Nifi

shows the home page of Cloudera, which provides the status information of the VM cluster. Although several different parcels are included in Cloudera, such as HBase, Hive, Spark, and YARN [102], only HDFS is started and used for demonstration purposes. Besides, Figure 4-8 describes the summary of HDFS, including the health information and the status summary.

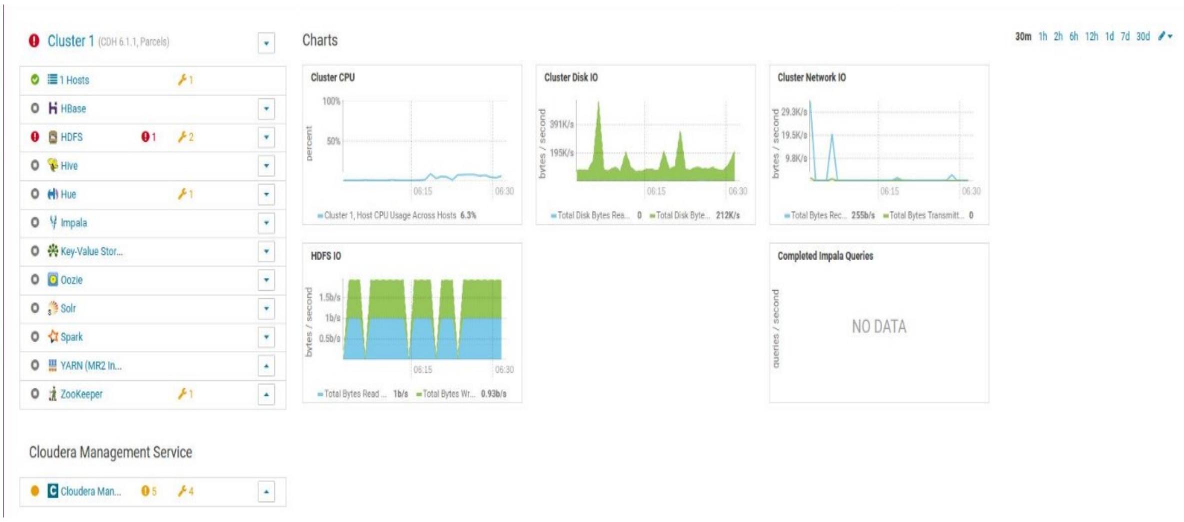
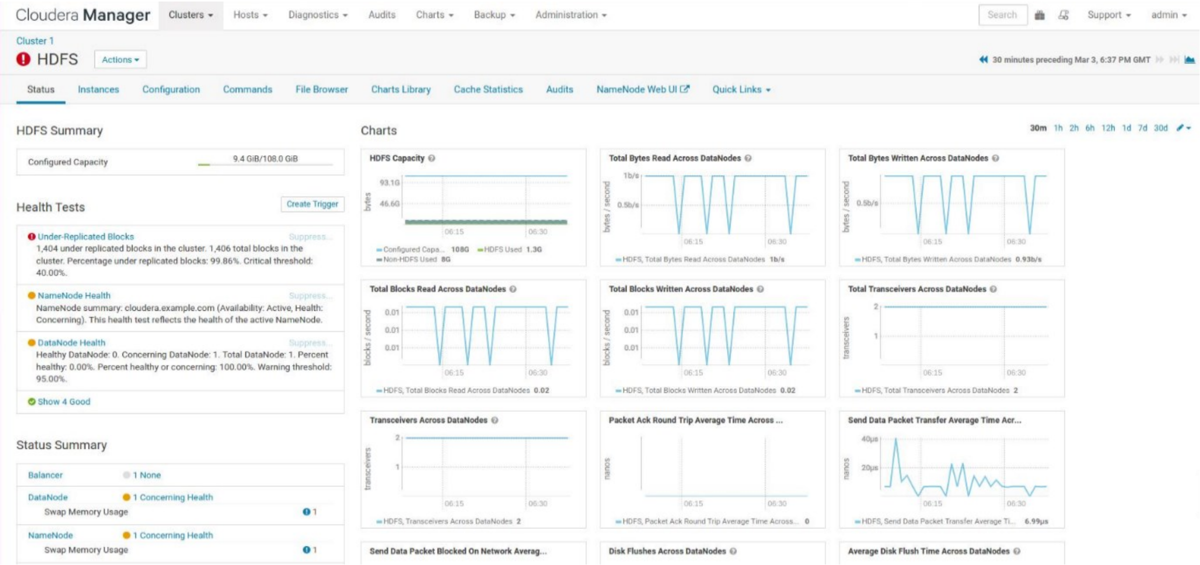


Figure 4-8 Platform home page


Cloudera Manager
Clusters
Hosts
Diagnostics
Audits
Charts
Backup
Administration
Search
Support
admin
Cluster 1
HDFS
Actions
30 minutes preceding Mar 3, 6:37 PM GMT
Status
Instances
Configuration
Commands
File Browser
Charts Library
Cache Statistics
Audits
NameNode Web UI
Quick Links
Charts
30m 1h 2h 6h 12h 1d 7d 30d
HDFS Capacity
bytes
93.1G
46.6G
06:15
06:30
Configured Capa... 108G
HDFS Used 1.3G
Non-HDFS Used 8G
Total Bytes Read Across DataNodes
bytes / second
1b/s
0.5b/s
HDFS, Total Bytes Read Across DataNodes 1b/s
Total Bytes Written Across DataNodes
0.5b/s
HDFS, Total Bytes Written Across DataNodes 0.93b/s
Total Blocks Read Across DataNodes
blocks / second
0.01
HDFS, Total Blocks Read Across DataNodes 0.02
Total Blocks Written Across DataNodes
HDFS, Total Blocks Written Across DataNodes 0.02
Total Transceivers Across DataNodes
transceivers
2
1
HDFS, Total Transceivers Across DataNodes 2
Transceivers Across DataNodes
HDFS, Transceivers Across DataNodes 2
Packet Ack Round Trip Average Time Across ...
nanos
HDFS, Packet Ack Round Trip Average Time Across... 0
Send Data Packet Transfer Average Time Acr...
40µs
20µs
HDFS, Send Data Packet Transfer Average Ti... 6.99µs
Send Data Packet Blocked On Network Averag...
Disk Flushes Across DataNodes
Average Disk Flush Time Across DataNodes
Create Trigger
Suppress...
Show 4 Good

HDFS Summary

Configured Capacity 9.4 GiB/108.0 GiB

Health Tests

Under-Replicated Blocks
1,404 under replicated blocks in the cluster. 1,406 total blocks in the cluster. Percentage under replicated blocks: 99.86%. Critical threshold: 40.00%.

NameNode Health
NameNode summary: cloudera.example.com (Availability: Active, Health: Concerning). This health test reflects the health of the active NameNode.

DataNode Health
Healthy DataNode: 0. Concerning DataNode: 1. Total DataNode: 1. Percent healthy: 0.00%. Percent healthy or concerning: 100.00%. Warning threshold: 95.00%.

Status Summary

Balancer	1 None	
DataNode	1 Concerning Health	
Swap Memory Usage		1
NameNode	1 Concerning Health	
Swap Memory Usage		1

Figure 4-9 Summary of HDFS

In general, the data exchange platform utilizes cloud computing techniques. Firstly, in terms of scalability, the designed platform is built up on the distributed framework, which is capable of scaling up and down based on the practical need (e.g., data exchanged, actors involved, etc.). However, ECCo-SP, proposed by ENTSO-E [103], is a centralized data exchange platform that is built on a single server where all the services are deployed. In addition, from the perspective of security, the designed platform is developed in the Linux operating system on the VM cluster, which allows itself to be compliant with open standards for security and enables itself to take advantage of the benefits of VMs. ECCo-SP makes use of a series of secure communication protocols and complies with IEC standards. Finally, regarding reliability, the HDFS file system of the designed platform supports data replication and fault tolerance across the VM cluster. Thus, even if a few VMs of the cluster are compromised, actors are still able to exchange data on the platform reliably and robustly.

4.4 Implementation of selected Business Use Cases (BUC)

This section illustrates the process of data exchange between different entities of the power system via the proposed platform in a specific scenario by highlighting the implementation of selected business use cases (BUCs). The detailed process of exchanging data between different actors in the BUC is described, and a table is provided to explain the steps of data exchange in the BUC. The table includes information about the producer and receiver of the data, the information exchanged, the data format, access control, and the time scale. This implementation is based on the BUCs from the European project TDX-Assist Horizon 2020 [103], which aims to enhance the coordination and data exchange between transmission and DSOs to ensure a more efficient and secure power system.

4.4.1 BUC 1: Coordination of operational planning activities between TSO and DSO

Business Use Case 1 (BUC 1) coordinates operational planning activities between TSOs and DSOs up to 72 hours in advance. It is implemented on the data exchange platform. The service associated with this scenario involves the exchange of information between TSOs and DSOs to enhance the programming of their network activities. DSOs can predict the load and distributed generation separately by technology type and location, with a sample interval of up to 15 minutes. Once shared with TSOs and correctly aggregated, this information allows the bulk power system to operate more efficiently and securely. Figure 4-10 illustrates the process of BUC 1 in this scenario. Throughout the entire process of BUC 1, in this BUC 12 actions are performed between the DSO, the TSO, and the market operator to exchange data. Table 4-2 provides a step-by-step analysis for BUC 1 [103].

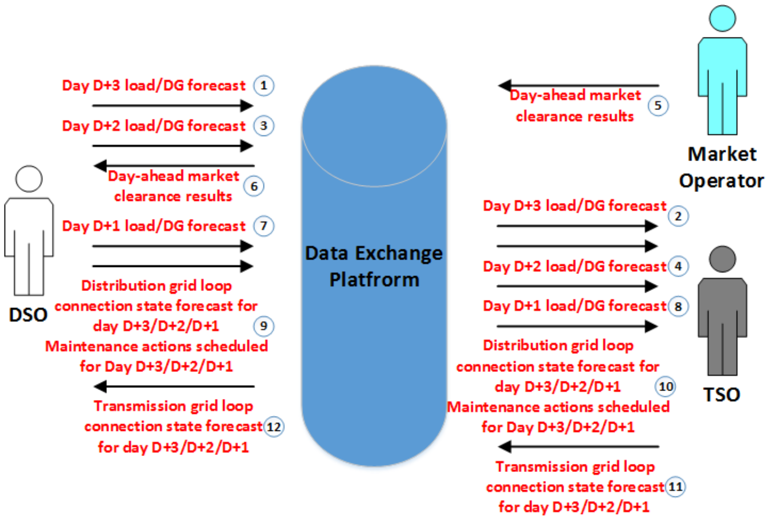
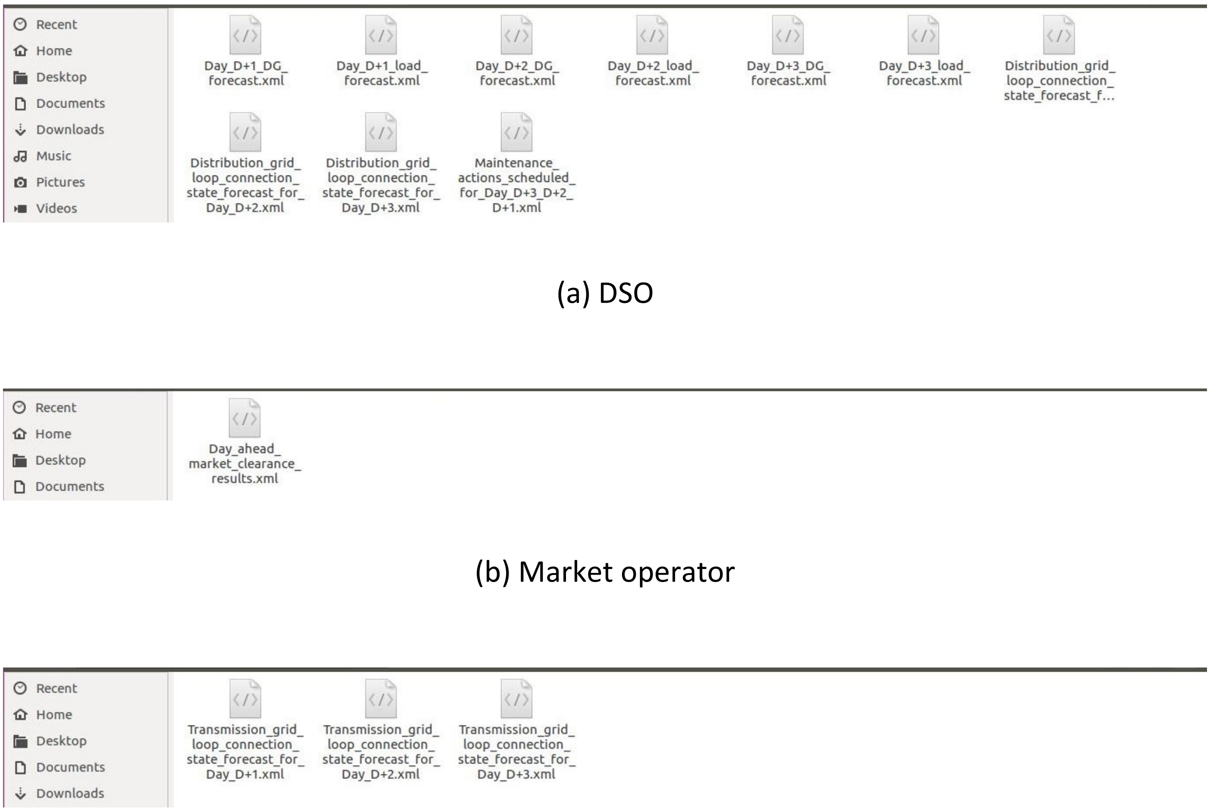

Day D+3 load/DG forecast 1
Day D+2 load/DG forecast 3
Day-ahead market clearance results 6
Day D+1 load/DG forecast 7
DSO
Distribution grid loop connection state forecast for day D+3/D+2/D+1 9
Maintenance actions scheduled for Day D+3/D+2/D+1
Transmission grid loop connection state forecast 12 for day D+3/D+2/D+1
Data Exchange Platfrorm
Day-ahead market clearance results 5
Market Operator
Day D+3 load/DG forecast 2
Day D+2 load/DG forecast 4
Day D+1 load/DG forecast 8
Distribution grid loop connection state forecast for day D+3/D+2/D+1 10
Maintenance actions scheduled for Day D+3/D+2/D+1
TSO
Transmission grid loop connection state forecast 11 for day D+3/D+2/D+1

Figure 4-10 Platform mechanism of BUC 1

Table 4-2 Steps Involved in BUC 1

Scenario name	All DRES under incentive					
Step No.	Information producer (actor)	Information receiver (actor)	Information exchanged	Information format	Access control	Time scale
1	DSO		Day D+3 load/DG forecast	CIM/XML	upload, display, delete	days, weeks, months
2		TSO			download, display, delete	
3	DSO		Day D+2 load/DG forecast	CIM/XML	upload, display, delete	days, weeks, months
4		TSO			download, display, delete	
5	Market Operator		Day-ahead market clearance results	CIM/XML	upload, display, delete	days, weeks, months
6		DSO			download, display, delete	
7	DSO		Day D+1 load/DG forecast	CIM/XML	upload, display, delete	days, weeks, months

8		TSO			download, display, delete	
9	DSO		Distribution grid loop connection state forecast for day D+3/D+2/D+1 Maintenance actions scheduled for Day D+3/D+2/D+1	CIM/XML	upload, display, delete	days, weeks, months
10		TSO			download, display, delete	
11	TSO		Transmission grid loop connection state forecast for day D+3/D+2/D+1	CIM/XML	upload, display, delete	days, weeks, months
12		DSO			download, display, delete	



Day_D+1_DG_forecast.xml
Day_D+1_load_forecast.xml
Day_D+2_DG_forecast.xml
Day_D+2_load_forecast.xml
Day_D+3_DG_forecast.xml
Day_D+3_load_forecast.xml
Distribution_grid_loop_connection_state_forecast_f…
Distribution_grid_loop_connection_state_forecast_for_Day_D+2.xml
Distribution_grid_loop_connection_state_forecast_for_Day_D+3.xml
Maintenance_actions_scheduled_for_Day_D+3_D+2_D+1.xml

(a) DSO

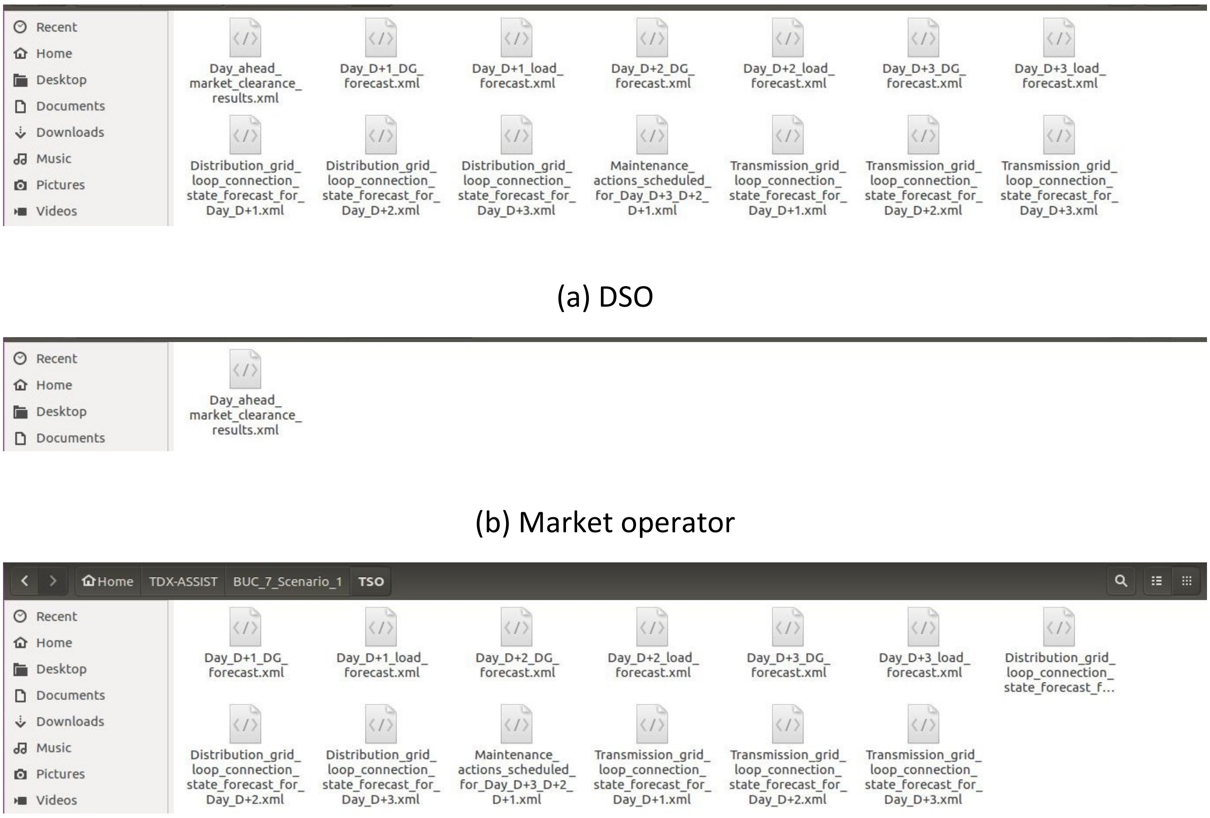
Day_ahead_market_clearance_results.xml

(b) Market operator

Transmission_grid_loop_connection_state_forecast_for_Day_D+1.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+2.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+3.xml

(c) TSO

Figure 4-11 BUC 1 data storage in the local file system before data exchange



Day_ahead_market_clearance_results.xml
Day_D+1_DG_forecast.xml
Day_D+1_load_forecast.xml
Day_D+2_DG_forecast.xml
Day_D+2_load_forecast.xml
Day_D+3_DG_forecast.xml
Day_D+3_load_forecast.xml
Distribution_grid_loop_connection_state_forecast_for_Day_D+1.xml
Distribution_grid_loop_connection_state_forecast_for_Day_D+2.xml
Distribution_grid_loop_connection_state_forecast_for_Day_D+3.xml
Maintenance_actions_scheduled_for_Day_D+3_D+2_D+1.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+1.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+2.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+3.xml

(a) DSO

Day_ahead_market_clearance_results.xml

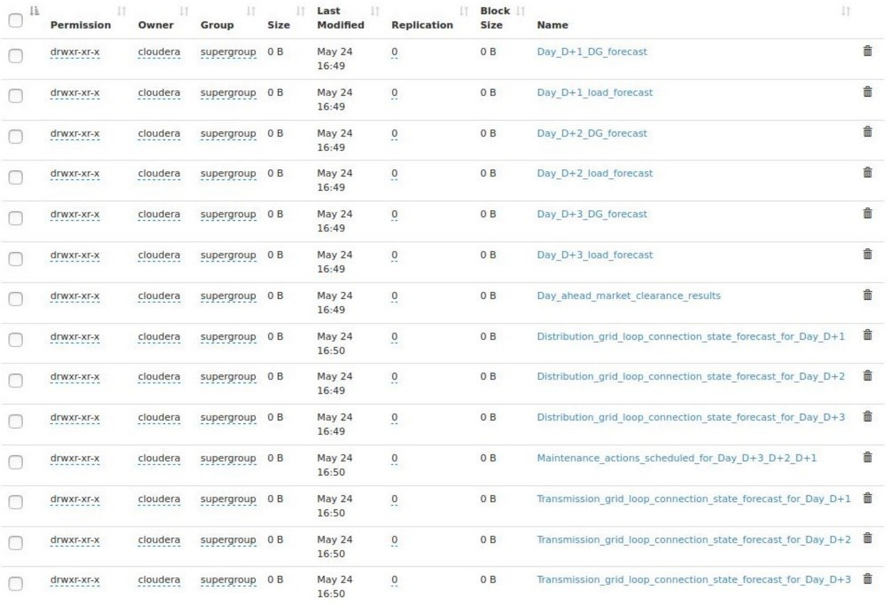
(b) Market operator

Day_D+1_DG_forecast.xml
Day_D+1_load_forecast.xml
Day_D+2_DG_forecast.xml
Day_D+2_load_forecast.xml
Day_D+3_DG_forecast.xml
Day_D+3_load_forecast.xml
Distribution_grid_loop_connection_state_forecast_f…
Distribution_grid_loop_connection_state_forecast_for_Day_D+2.xml
Distribution_grid_loop_connection_state_forecast_for_Day_D+3.xml
Maintenance_actions_scheduled_for_Day_D+3_D+2_D+1.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+1.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+2.xml
Transmission_grid_loop_connection_state_forecast_for_Day_D+3.xml

(c) TSO

Figure 4-12 BUC 1 data storage in the local file system after data exchange

Figures 4-11 and 4-12 demonstrate the data storage of different actors (TSO, DSO, and Market operator) of BUC 1 in the local file system, before and after data exchange, respectively. After the actors exchange data through the platform, they have the requested data stored in their respective directories. As seen in Figures 4-11(a) and 4-12(a), the DSO receives the requested data, such as the transmission grid loop connection state forecast for Day D+3/D+2/D+1, after data exchange through the platform. Similarly, as seen in Figures 4-11(c) and 4-12(c), the TSO receives the requested data, such as the Day D+3 load forecast and the Day D+3 DG forecast, after data exchange through the platform. Furthermore, Figure 4-13 illustrates the data storage for BUC 1 in HDFS. It can be observed that fourteen types of requested data are stored in the corresponding directories of HDFS. All the requested data is stored and exchanged on the platform in the CIM/XML format. For example, as seen in Figure 4-9(a), the Day D+1 load forecast and the Day D+1 DG forecast are stored in the CIM/XML format in the directories of "Day D+1 load forecast" and "Day D+1 DG forecast" in HDFS, respectively.



□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+1_DG_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+1_load_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+2_DG_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+2_load_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+3_DG_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_D+3_load_forecast	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Day_ahead_market_clearance_results	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:50	0	0 B	Distribution_grid_loop_connection_state_forecast_for_Day_D+1	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Distribution_grid_loop_connection_state_forecast_for_Day_D+2	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:49	0	0 B	Distribution_grid_loop_connection_state_forecast_for_Day_D+3	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:50	0	0 B	Maintenance_actions_scheduled_for_Day_D+3_D+2_D+1	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:50	0	0 B	Transmission_grid_loop_connection_state_forecast_for_Day_D+1	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:50	0	0 B	Transmission_grid_loop_connection_state_forecast_for_Day_D+2	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 24 16:50	0	0 B	Transmission_grid_loop_connection_state_forecast_for_Day_D+3	🗑

Figure 4-13 Data storage for BUC 1 in HDFS

4.4.2 BUC 2: Coordination of long-term network planning between TSO and DSO

BUC 2 coordinates the long-term network development plans between TSOs and DSOs on the TSO/DSO interface and is implemented on the data exchange platform. This service includes exchanging information between TSOs and DSOs to develop long-term investment, expansion, and reinforcement plans to ensure long-term network stability and resilience. DSOs and TSOs discuss potential improvements to the current network model simplifications, which also include the addition of new interface substations and HV power lines and the removal of existing ones. The network plans may also include DSO or TSO network reinforcement plans, as well as the anticipated connectivity of key grid users to the DSO or TSO network. When relevant changes to the TSO/DSO interface plan are made, this information is exchanged. By considering both TSO and DSO network plans, both TSO and DSO can identify synergies and the best time to execute the plans. Figure 4-14 illustrates the process of BUC 2 in the scenario of Development plans. Throughout the entire process of BUC 2, in this BUC 4 actions are performed between the DSO and the TSO to exchange data. These actions involve the exchange of development plans for the transmission network and development plans for the distribution network, via the platform. Table 4-3 provides a step-by-step process for BUC 2 [103].

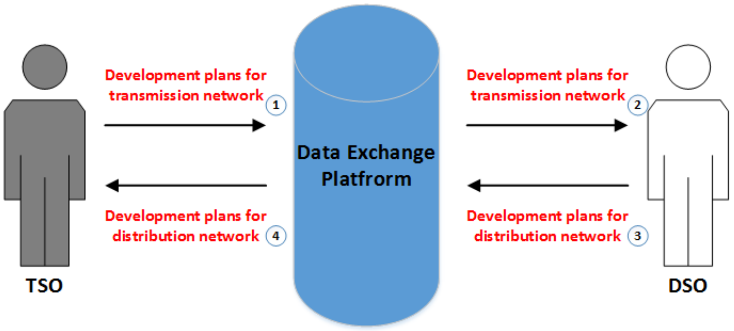
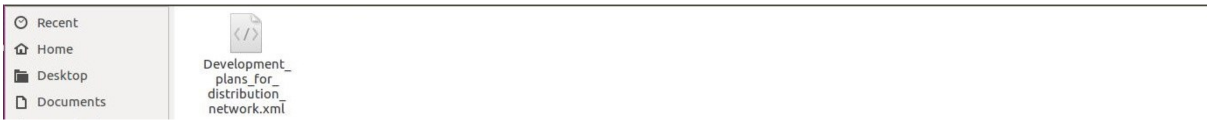

Development plans for transmission network
1
Development plans for distribution network
4
TSO
Data Exchange Platfrorm
Development plans for transmission network
2
Development plans for distribution network
3
DSO

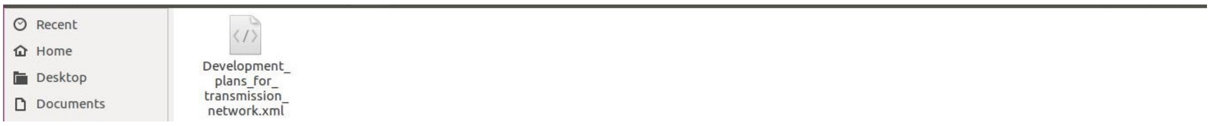
Figure 4-14 Platform mechanism of BUC 2

Table 4-3 Steps Involved in BUC 2

Scenario name	Coordination of Long-Term Network Development Plans on the TSO/DSO Interface					
Step No.	Information producer (actor)	Information receiver (actor)	Method/Information exchanged	Information format	Access control	Time scale
1	TSO		Development plans for transmission network	CIM/XML	upload, display, delete	years
2		DSO			download, display, delete	
3	DSO		Development plans for the distribution network	CIM/XML	upload, display, delete	years
4		TSO			download, display, delete	

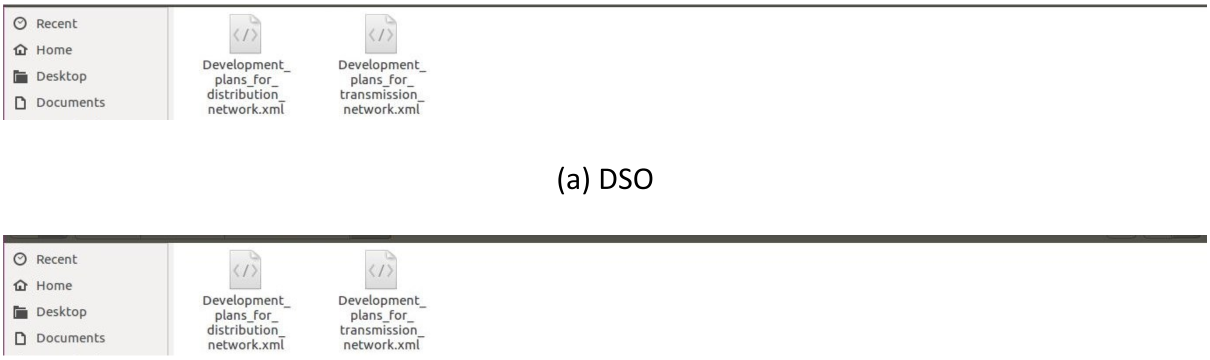


(a) DSO



(b) TSO

Figure 4-15 BUC 2 data storage in the local file system before data exchange

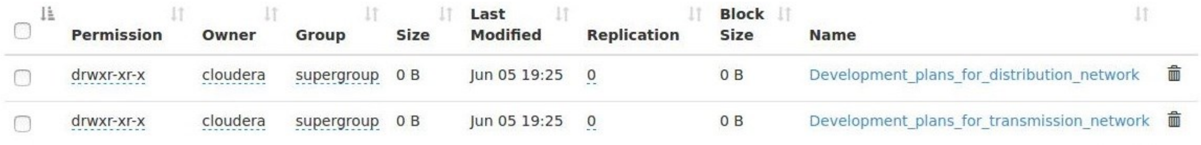

Recent
Home
Desktop
Documents
Development_plans_for_distribution_network.xml
Development_plans_for_transmission_network.xml
Recent
Home
Desktop
Documents
Development_plans_for_distribution_network.xml
Development_plans_for_transmission_network.xml

(a) DSO

(b) TSO

Figure 4-16 BUC 2 data storage in local file system after data exchange

Figures 4-15 and 4-16 demonstrate the data storage of different actors (TSO and DSO) of BUC 2 in the local file system before and after data exchange, respectively. After the actors exchange data through the platform, they have the requested data stored in their respective directories. As seen in Figures 4-15(a) and 4-16(a), the DSO receives the requested data, which is the transmission network development plans, after data exchange through the platform. Similarly, as seen in Figures 4-15(b) and 4-16(b), the TSO receives the requested data, which is the distribution network development plans, after data exchange via the platform. Furthermore, Figure 3-17 illustrates the data storage for BUC 2 in HDFS. It can be observed that two types of requested data are stored in the corresponding directories of HDFS. All the requested data is stored and exchanged on the platform in the CIM/XML format. For example, as seen in Figures 4-15(a) and 4-16(b), the development plans for the distribution network and the development plans for the transmission network are stored in the CIM/XML format in the directories of "Development plans for distribution network" and "Development plans for transmission network" in HDFS, respectively.



□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	drwxr-xr-x	cloudera	supergroup	0 B	Jun 05 19:25	0	0 B	Development_plans_for_distribution_network
□	drwxr-xr-x	cloudera	supergroup	0 B	Jun 05 19:25	0	0 B	Development_plans_for_transmission_network

Figure 4-17 Data storage for BUC 2 in HDFS

4.4.3 BUC 3: Improve system real-time supervision and control through better coordination

BUC 3 manages two different scenarios, each of which outlines a specific process of data exchange among various actors. The scenario that is implemented is "Real-time information exchange" and it provides the service of exchanging real-time information between TSOs and DSOs regarding their networks and other connected resources. The main focus is on making sure that the essential real-time information flow between TSOs and DSOs is in place, to enable better monitoring and management of transmission and distribution networks. Standardization/normalization of real-time data flow between TSOs and DSOs will be addressed to improve comprehension of counterparty signals. It will also develop the necessary practices to allow for recurring changes to the TSO and DSO's observability regions. This will guarantee that the networks are sufficiently visible to one another, regardless of how the topology of the network changes over time. Figure 4-18 illustrates the platform mechanism of BUC 3 in the scenario for Real-time information exchange. The illustration shows that there are a total of 8 actions between the DSO and the TSO to exchange data, such as IDs of data signals of DSO within the TSO observability area and short-circuit power values of transmission network buses within the DSO observability area, via the platform. Table 4-4 provides a step-by-step process for BUC 3 [103].

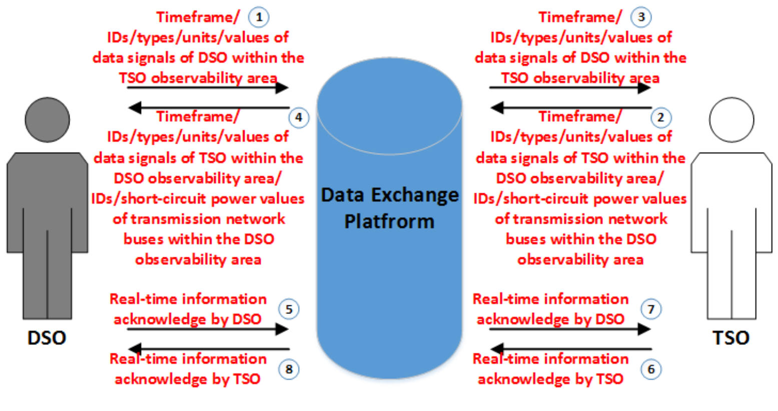
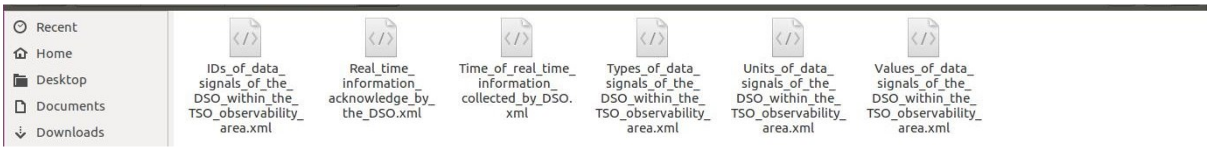


Figure 4-18 Platform mechanism of BUC 3

Table 4-4 Steps Involved in BUC 3

Scenario name	Real-time information exchange					
Step No.	**Information producer (actor)**	**Information receiver (actor)**	**Method/Information exchanged**	**Information format**	**Access control**	**Timescale**
1	DSO		Timeframe/ IDs/types/units/values of data signals of DSO within the TSO observability area	CIM/XML	upload, display, delete	real-time
2	TSO		Timeframe/ IDs/types/units/values of data signals of TSO within the DSO observability area/ IDs/short-circuit power values of transmission network buses within the DSO observability area	CIM/XML	upload, display, delete	real-time
3		TSO			download, display, delete	
4		DSO			download, display, delete	

5	DSO		Real-time information acknowledged by DSO	CIM/XML	upload, display, delete	real-time
6	TSO		Real-time information acknowledged by TSO	CIM/XML	upload, display, delete	real-time
7		TSO			download, display, delete	
8		DSO			download, display, delete	



IDs_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Real_time_information_acknowledge_by_the_DSO.xml

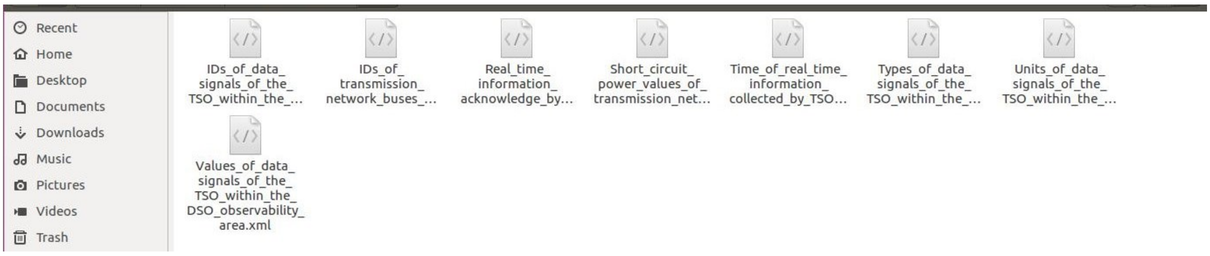
Time_of_real_time_information_collected_by_DSO.xml

Types_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Units_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Values_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

(a) DSO



IDs_of_data_signals_of_the_TSO_within_the_…

IDs_of_transmission_network_buses_…

Real_time_information_acknowledge_by…

Short_circuit_power_values_of_transmission_net…

Time_of_real_time_information_collected_by_TSO…

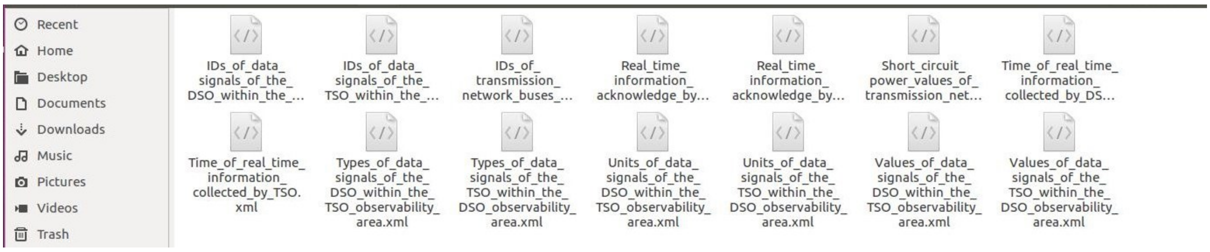
Types_of_data_signals_of_the_TSO_within_the_…

Units_of_data_signals_of_the_TSO_within_the_…

Values_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml

(b) TSO

Figure 4-19 BUC 3 Data storage in the local file system before data exchange



IDs_of_data_signals_of_the_DSO_within_the_…

IDs_of_data_signals_of_the_TSO_within_the_…

IDs_of_transmission_network_buses_…

Real_time_information_acknowledge_by…

Real_time_information_acknowledge_by…

Short_circuit_power_values_of_transmission_net…

Time_of_real_time_information_collected_by_DS…

Time_of_real_time_information_collected_by_TSO.xml

Types_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Types_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml

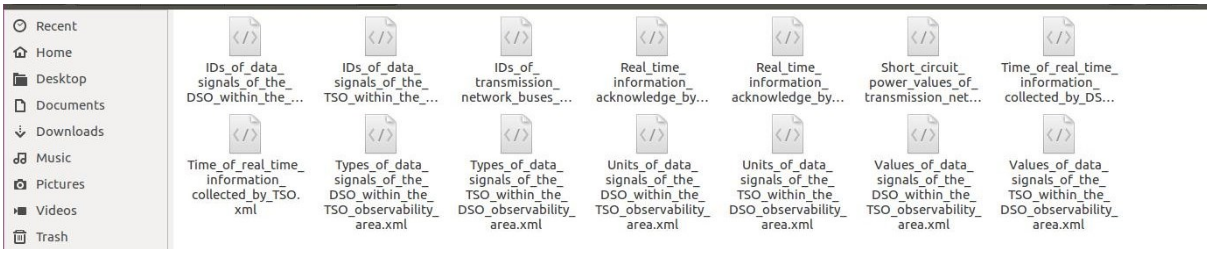
Units_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Units_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml

Values_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml

Values_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml

(a) DSO

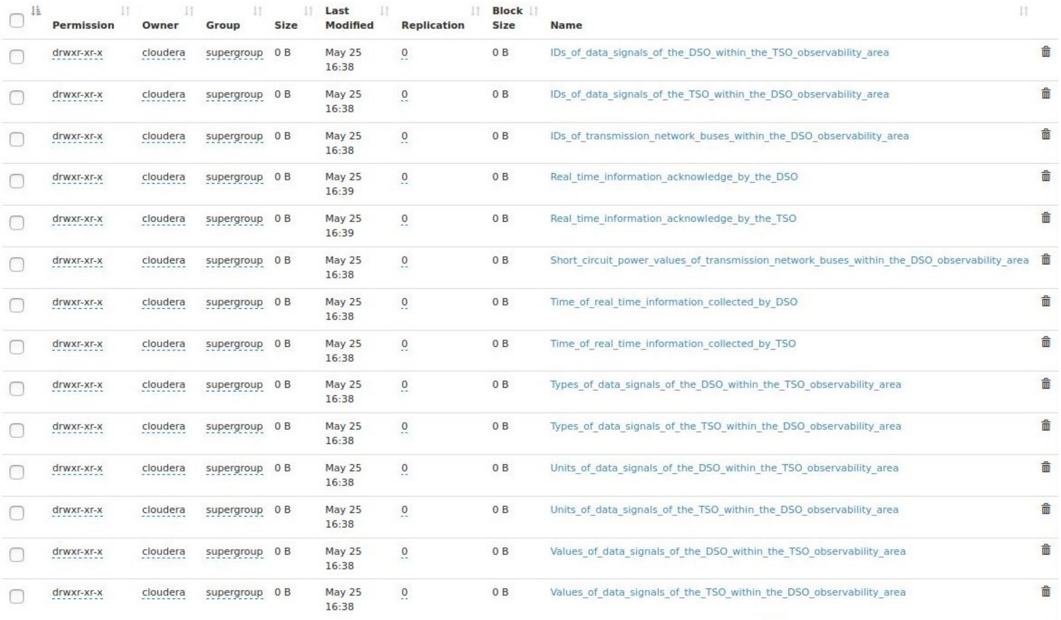


IDs_of_data_signals_of_the_DSO_within_the_...
IDs_of_data_signals_of_the_TSO_within_the_...
IDs_of_transmission_network_buses_...
Real_time_information_acknowledge_by...
Real_time_information_acknowledge_by...
Short_circuit_power_values_of_transmission_net...
Time_of_real_time_information_collected_by_DS...
Time_of_real_time_information_collected_by_TSO.xml
Types_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml
Types_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml
Units_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml
Units_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml
Values_of_data_signals_of_the_DSO_within_the_TSO_observability_area.xml
Values_of_data_signals_of_the_TSO_within_the_DSO_observability_area.xml

(b) TSO

Figure 4-20 BUC 3 data storage in the local file system after data exchange

Figures 4-19 and 4-20 demonstrate the data storage of different actors (TSO and DSO) of BUC 10 in the local file system before and after data exchange, respectively. It is clear that after the actors exchange the data through the platform, they have the requested data stored in their respective directories. As seen in Figures 4-19(a) and 4-20(a), the DSO receives the requested data, such as the time of real-time information collected by TSO and the IDs/short circuit power values of transmission network buses within the DSO observability area, after data exchange through the platform. Similarly, as seen in Figures 4-19(b) and 4-20(b), the TSO receives the requested data, such as the IDs/types/units/values of data signals of the DSO within the TSO observability area and the real-time information acknowledged by the DSO, after data exchange through the platform. Furthermore, Figure 4-21 illustrates the data storage for BUC 10 in HDFS. There are a total of fourteen types of requested data that are stored in the corresponding directories of HDFS for data exchange. All the requested data is stored and exchanged on the platform in the CIM/XML format. For instance, as seen in Figure 4-19(b), the time of real-time information collected by TSO and the real-time information acknowledged by the TSO are stored in the CIM/XML format in the directories of "Time of real-time information collected by TSO" and "Real-time information acknowledged by the TSO" in HDFS, respectively.



□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	IDs_of_data_signals_of_the_DSO_within_the_TSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	IDs_of_data_signals_of_the_TSO_within_the_DSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	IDs_of_transmission_network_buses_within_the_DSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:39	0	0 B	Real_time_information_acknowledge_by_the_DSO	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:39	0	0 B	Real_time_information_acknowledge_by_the_TSO	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Short_circuit_power_values_of_transmission_network_buses_within_the_DSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Time_of_real_time_information_collected_by_DSO	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Time_of_real_time_information_collected_by_TSO	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Types_of_data_signals_of_the_DSO_within_the_TSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Types_of_data_signals_of_the_TSO_within_the_DSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Units_of_data_signals_of_the_DSO_within_the_TSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Units_of_data_signals_of_the_TSO_within_the_DSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Values_of_data_signals_of_the_DSO_within_the_TSO_observability_area	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:38	0	0 B	Values_of_data_signals_of_the_TSO_within_the_DSO_observability_area	🗑

Figure 4-21 Data storage for BUC 3 in HDFS

4.4.4 BUC 4: Improve fault location close to the TSO-DSO Interface

BUC 4 handles a specific data exchange process among various actors using the scenario "Improve Fault Location Near the TSO-DSO Interface." This scenario provides the service of determining the location of faults on distribution network lines that are connected to the transmission network interface. The goal is to improve the accuracy of fault locations on distribution lines that are directly connected to transmission bays owned and operated by both the TSO and the DSO by using data gathered on the TSO side. This requires real-time exchange of additional information between the two system operators. Figure 4-22 illustrates the platform mechanism of BUC 4 in the scenario for Fault occurrence, detection, and information exchange. Throughout the entire process of BUC 4, there are a total of 4 actions between the DSO and the TSO to exchange data, such as network fault and impedance to a fault, via the platform. Table 4-5 provides a step-by-step analysis for BUC 4 [103].

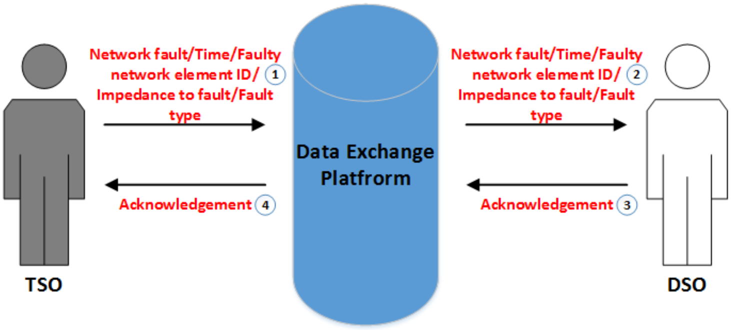
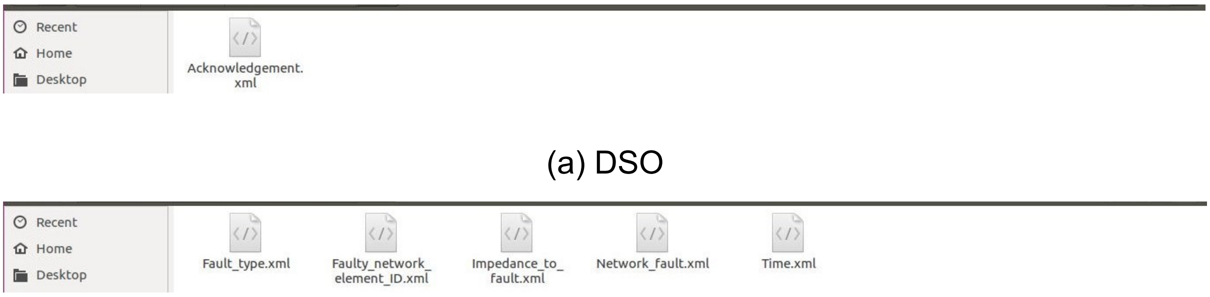

Network fault/Time/Faulty network element ID/ Impedance to fault/Fault type
1
Acknowledgement
4
Data Exchange Platfrorm
Network fault/Time/Faulty network element ID/ Impedance to fault/Fault type
2
Acknowledgement
3
TSO
DSO

Figure 4-22 Platform mechanism of BUC 4

Table 4-5 Steps Involved in BUC 2

Scenario name	Fault occurrence, detection, and information exchange					
Step No.	Information producer (actor)	Information receiver (actor)	Method/Information exchanged	Information format	Access control	Time scale
1	TSO		Network fault/Time/Faulty network element ID/Impedance to fault/Fault type	CIM/XML	upload, display, delete	Real-time
2		DSO			download, display, delete	
3	DSO		Acknowledgment	CIM/XML	upload, display, delete	Real-time

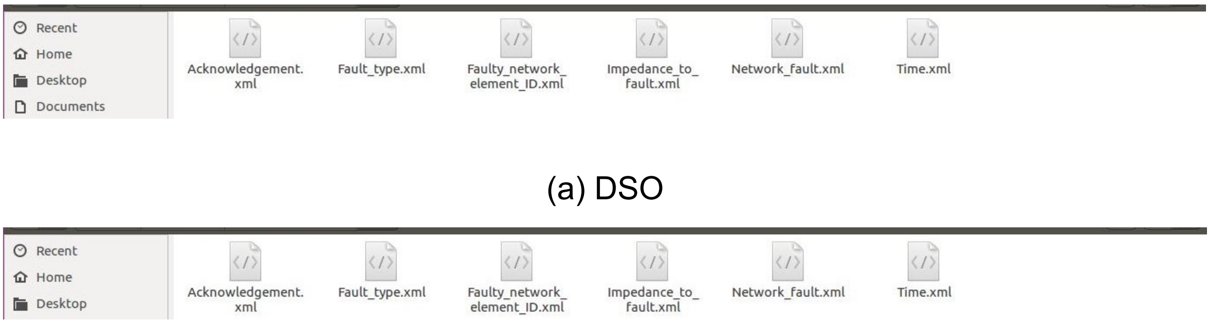
4		TSO			download, display, delete	



(a) DSO

(b) TSO

Figure 4-23 BUC 4 data storage in the local file system before data exchange

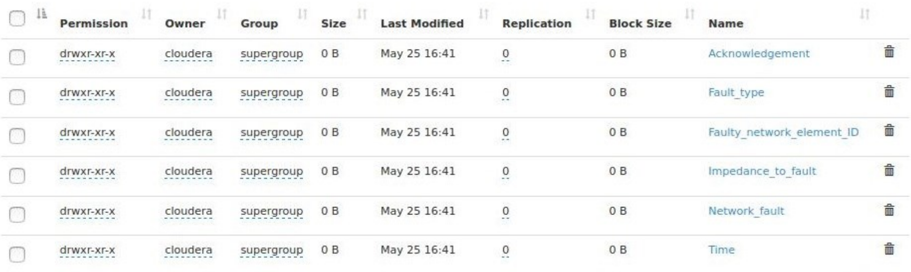


(a) DSO

b) TSO

Figure 4-24 BUC 4 data storage in the local file system after data exchange

Figures 4-23 and 4-24 demonstrate the data storage of the various actors (TSO and DSO) of BUC 4 in the local file system before and following the data exchange, respectively. After the actors exchange data through the platform, they have the requested data stored in their respective directories. As seen in Figures 4-23(a) and 4-24(a), the DSO receives the requested data, which includes network fault, time, faulty network element ID, impedance to fault, and fault type, after data exchange through the platform. Similarly, as seen in Figures 4-23(b) and 4-24(b), the TSO receives the requested data, which is the acknowledgment, after data exchange through the platform. Furthermore, Figure 4-25 illustrates the data storage for BUC 4 in HDFS. It can be observed that six types of requested data are stored in the corresponding directories of HDFS for data exchange. All the requested data is stored and exchanged on the platform in the CIM/XML format. For instance, as seen in Figure 4-23(b), the fault type and the network fault are stored in the CIM/XML format in the directories of Fault type and Network fault in HDFS, respectively.



□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Acknowledgement	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Fault_type	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Faulty_network_element_ID	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Impedance_to_fault	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Network_fault	🗑
□	drwxr-xr-x	cloudera	supergroup	0 B	May 25 16:41	0	0 B	Time	🗑

Figure 4-25 Data storage for BUC 4 in HDFS

4.5 Experimental results

This section evaluates the performance of the platform designed in the previous section. It assesses the platform's ability to meet the data access needs of the web portal and priority portal information, as demonstrated by the services extracted from the selected BUCs. The evaluation covers three key areas: scalability, reliability, and the benefits of the platform. The aim is to determine the suitability of the platform for its intended purpose.

4.5.1 Scalability test

To examine the scalability of the proposed data exchange platform, experiments were conducted using services extracted from BUCs. These experiments simulate scenarios where data is being exchanged between multiple TSOs and DSOs. The experiments were designed to test the platform's ability to handle increasing amounts of data and users without experiencing a significant decrease in performance. The results of these experiments are used to determine the platform's scalability and the ability to add additional users.

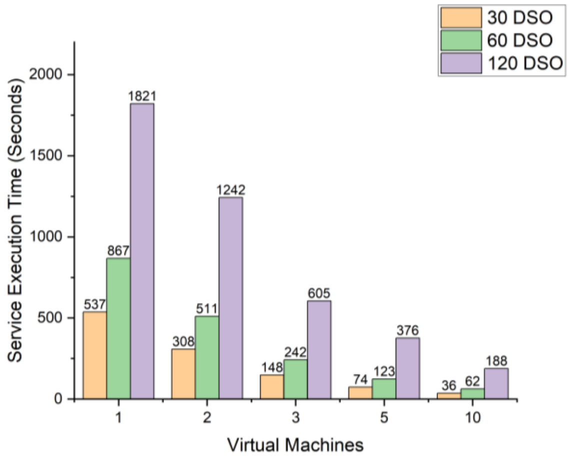


Figure 4-26 BUC 1 Service execution time with One TSO

As shown in Figure 4-26, there is a correlation between the number of DSOs participating in the service and the time it takes to execute the service. As the number of DSOs increases, the execution time also increases in a nearly linear fashion. This is because more actions are required for data exchange when more DSOs are involved. However, when the number of VMs increases, the execution time of the service decreases linearly. This is because the actions are distributed among the VMs, which all have similar specifications and configurations. This linear performance scaling is observed as more resources are added to the system, highlighting the scalability of the platform. To further evaluate scalability, similar experiments were conducted using services extracted from BUC 1, BUC 3, and BUC 4, with 3 TSOs interacting with 60 DSOs to exchange data. The results of these experiments also confirm that the platform is highly scalable, even when more resources, TSOs, and DSOs participate in the data exchange process.

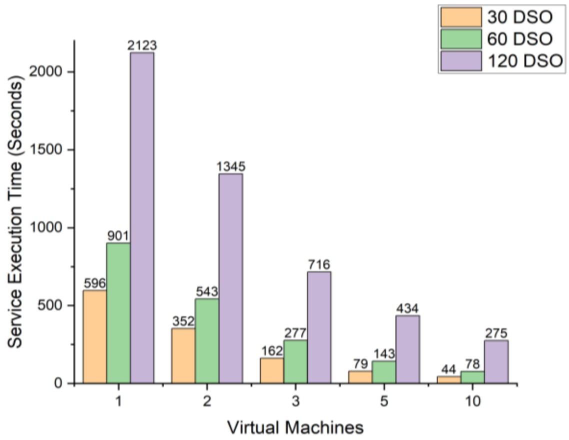


Figure 4-27 BUC 1 Service execution time with 2 TSOs and increasing numbers of DSOs

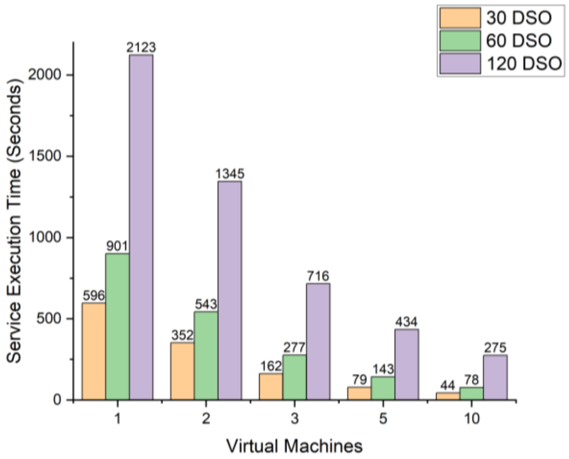


Figure 4-28 BUC 1 Service execution time with 3 TSOs and increasing numbers of DSOs

As illustrated in Figure 4-27, an increase in the number of VMs leads to a decrease in the service execution time. This is because the processes and data necessary for communication between TSOs are distributed among multiple VMs. This allows for more resources to be devoted to data communication, thereby speeding up the process. As the number of VMs continues to grow, the service execution time will continue to decrease, but there may be a point where the improvements become less substantial and may plateau as a result of the management overhead associated with maintaining a cluster of VMs.

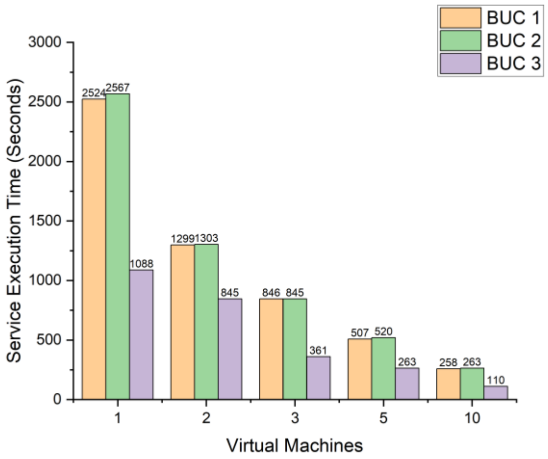


Figure 4-29 Service execution time with increasing number of VMs in BUC 1, BUC 2, and BUC 3

To evaluate the platform's ability to handle increased workloads, experiments were conducted by deploying services extracted from four different BUCs (BUC 1, BUC 2, BUC 3, and BUC 4) onto various numbers of VMs. The experiment assumed that each

BUC's services involved 2 TSOs and 30 DSOs, and all services were run concurrently on the VM cluster. Three different deployment scenarios were considered:

Case 1: All BUC services were deployed on a single VM (VM 1)

Case 2: BUC 1 and BUC 2 services were deployed on one VM (VM 1), while BUC 3 and BUC 4 services were deployed on a separate VM (VM 2)

Case 3: Each BUC service was deployed on a different VM (VM 1 for BUC 1, VM 2 for BUC 2, VM 3 for BUC 3, and VM 4 for BUC 4)

This experiment was designed to test the platform's ability to handle an increasing amount of data and a larger number of users while maintaining consistent performance as the number of VMs increases. This will help to determine the platform's scalability and its capacity to handle high user volume.

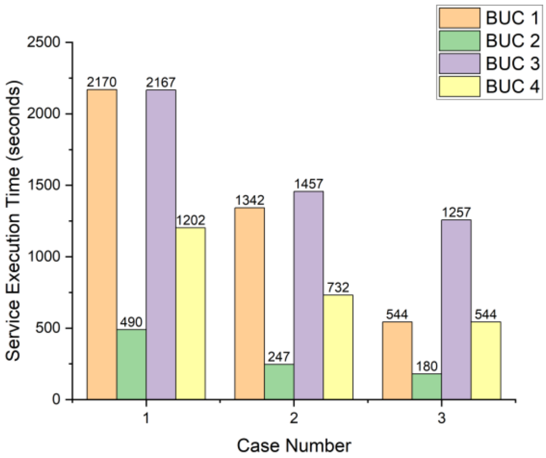


Figure 4-30 Service execution time with 2 TSOs and 30 DSOs for BUC 1, BUC 2, BUC 3 and BUC 4

The results of the performance evaluation for the proposed approach are presented in Figure 4-30. The graph demonstrates that when all services were executed on a VM with adequate computing and storage resources, as in Case 3, the total time for service execution was the most efficient. However, when all services were executed on a single VM as in Case 1, the total time for service execution was the least efficient. This is because all services share the same computing and storage resources. In summary, the scalability of the data exchange cloud platform was evaluated through three different experiments. The results of the experiments suggest that the platform has a high level of scalability and adaptability. As the number of VMs increases, the platform can support more services, handle more data, and involve more participants with faster execution times.

4.5.2 Reliability testing

The platform's reliability was evaluated by conducting three different deployment scenarios for services extracted from BUC 1, BUC 2, BUC 3, and BUC 4. As shown in Figure 4-31, the platform's ability to handle VM failure was assessed. Out of the three deployments, the first cluster (Cluster 1) was determined to be the most reliable. This is because each BUC's service was deployed and executed on multiple VMs, ensuring that even if one VM went down, the services would still be available. However, the other two clusters were found to be less reliable as a failure in one VM in these clusters would result in service interruption.

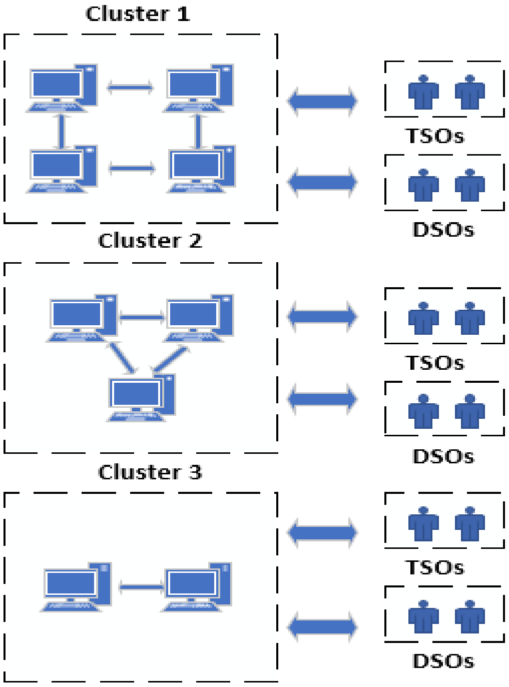


Figure 4-31 Virtual Machine Clusters for reliability analysis

It is particularly noteworthy that VM cluster 3 is vulnerable to failure if a single VM is compromised, as all services in the cluster are hosted on just two VMs. This highlights that the cloud platform for data exchange is more reliable, resilient, and robust with a larger number of VMs in the cluster. This is because deploying each service on multiple VMs provides redundancy, ensuring that services can continue to be supported even if one or more VMs are compromised.

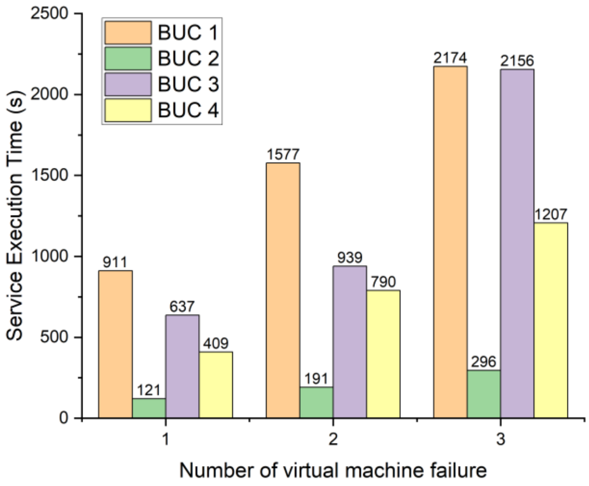


Figure 4-32 Reliability of Cloud platform with regards to the number of virtual machine failure

The performance of the BUC deployments on three different VM clusters, each consisting of 2 TSOs and 30 DSOs, during data exchange, is displayed in Figure 4-32. When three VMs malfunctioned in the proposed cluster, there was a considerable increase in service execution time. This demonstrates that the cluster presented in Figure 4-31 is superior in performance as compared to the other clusters. The results imply that the proposed cluster is reliable, as it can still exchange data even when VMs malfunction or are removed.

4.6 Chapter Summary

This chapter presents a cloud-based platform for the exchange of information and data among various entities involved in power systems. The utilization of cloud computing provides benefits such as scalability and reliability. The platform has been evaluated with multiple use cases to assess its suitability for data exchange between TSOs and DSOs. The platform has been designed to be secure, reliable, and scalable, and to enable interoperability among various actors in the power systems industry. A service-oriented mapping approach is employed on a cluster of VMs to deploy different use case scenarios. The platform utilizes Cloudera as a cloud service provider for data exchange. Simulation results indicate that the platform is scalable and reliable for exchanging data among power system entities, and it allows actors to access services through the cloud, with access to different services based on their roles or needs.

Chapter 5 Blockchain-Enabled Data Exchange Platform for Power System

5.1 Introduction

Blockchain technology, known for its ability to decentralize systems and eliminate intermediaries, has gained widespread interest across various industries and applications. It is a shared and distributed ledger that records transactions and is managed by a network of untrusted nodes. Each node holds a copy of the ledger, typically represented as a chain of blocks, each containing a logical sequence of transactions. The blocks include the hash of the preceding block, making the ledger tamper-proof and providing a secure, transparent, and tamper-proof record of transactions. This capability has enabled the use of blockchain technology in various applications, including financial services, supply chain management, and digital identity verification.

This chapter presents a platform for the exchange of information and data among entities in the power systems industry. This platform combines Hyperledger Fabric blockchain technology with Apache Hadoop to provide a secure and private environment for data exchange. By using blockchain technology, the platform ensures that data is kept secure and private, while the integration of Apache Hadoop allows for third-party computation to be performed within the data owner's environment. This enables the platform to provide a high level of security and privacy while also allowing for powerful computation capabilities. The platform can be beneficial to the Transmission TSOs and DSOs by providing transparency and accountability for data access and usage through blockchain and smart contract technology. Additionally, the platform allows for specific conditions to be set for data exchange, ensuring that data is used in an appropriate and authorized manner.

A blockchain is a secure, distributed ledger that is maintained by network nodes and users. On a ledger, transactions do not require monitoring by an external party. Instead, the network's nodes use a consensus process to alter the ledger's state. In a public blockchain, anyone can participate in the network under an anonymous identity, while in a permissioned blockchain, a node must receive approval from the network before joining. In contrast, traditional databases and private blockchains are centralized systems that rely on a central authority. While private blockchains share many of the same characteristics as public blockchains, such as immutability and digital signatures, they offer additional security and control benefits. The design and operation of the prototype will be discussed in subsequent sections.

5.2 System model design

The proposed architecture, as depicted in Figure 5-1, comprises three main components: data storage, blockchain, and computation. The primary users of this blockchain consortium are data suppliers, data consumers, TSOs, DSOs, and market participants. The data is provided by the users and is transferred to Apache Hadoop for processing via the blockchain. Authorized data consumers can only access the provided data set to run their code. The smart contract, created by the data providers, governs the use of the data set and monitors the computational complexity of the consumer code, preventing any harmful functions. The smart contract is a collaborative effort between the data providers. The Hadoop Distributed File System (HDFS) is used as the storage layer, where the data is stored and is also included in the architecture. This approach allows large amounts of data to be stored off-chain, thus increasing the computational efficiency of the blockchain.

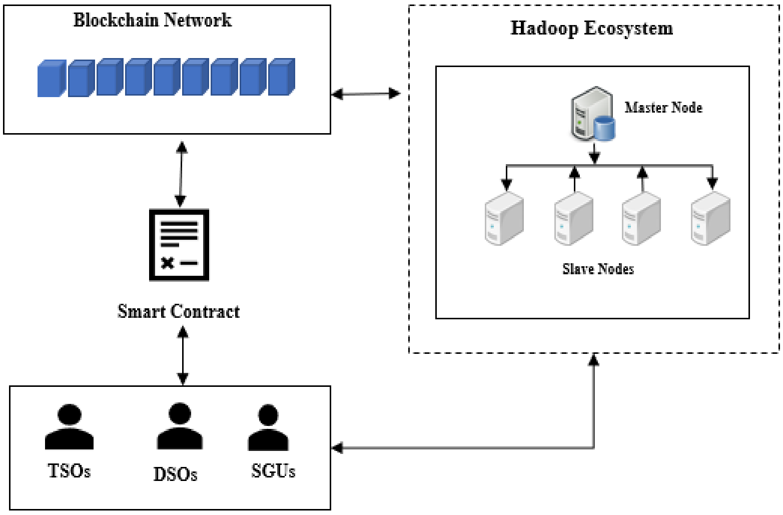


Figure 5-1 Blockchain-based data exchange architecture

The proposed system incorporates Hyperledger Fabric (HLF) blockchain technology to trace data provenance by maintaining a shared ledger. However, the storage capacity of the HLF blockchain can become a constraint as the size of the ledger grows. To overcome this limitation, the system stores data in an Apache Hadoop environment, which allows for off-chain storage. The integrity of the stored data is verified by using checksums, which are compared with the recorded information in the shared ledger to validate the stored data. The HLF network utilizes chain code to facilitate these processes on each peer node, and the data checksum and provenance data are sent using the built-in client library, eliminating the need for file storage operators. The distributed Hadoop ecosystem provides a secure and validated data storage alternative. The proposed method uses Hadoop to store data, this allows the system to speed up data processing. The data is first stored on a Hadoop storage system and then transferred to the blockchain for verification. The ledger is used to obtain the location and address of the data, which is then retrieved from the Hadoop storage. The decentralized consensus framework of blockchain technology allows for verifiable data transactions, and the use of HDFS allows for efficient storage and retrieval of large amounts of data. This workflow is illustrated in Figure 5-2, it shows how the system uses Hadoop for data storage, and blockchain for data verification and retrieval.

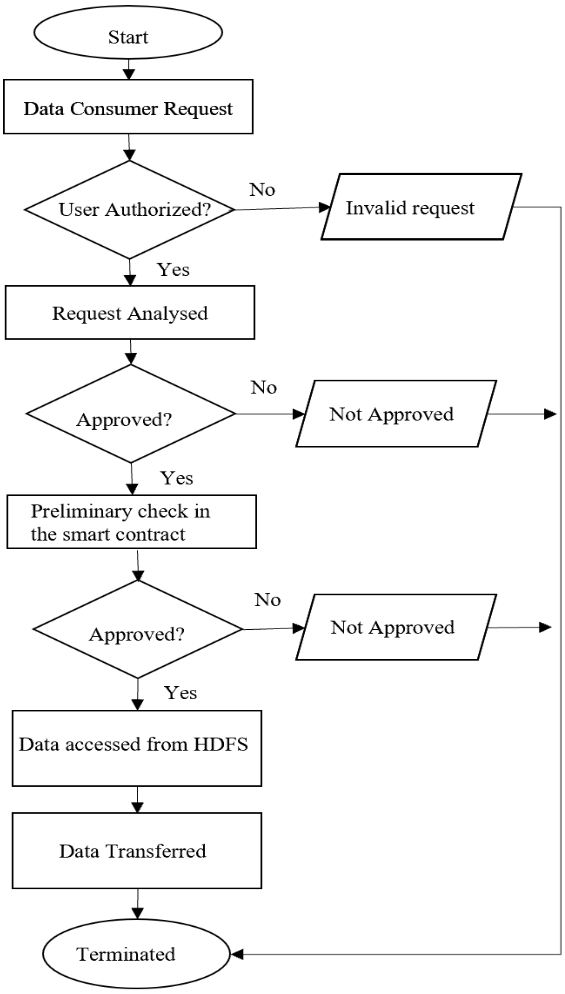


Figure 5-2 Blockchain platform workflow

In this system, the blockchain serves as a decentralized and immutable ledger that maintains a record of transactions related to files stored in a 'traditional' cloud-based file store. When a user wants to upload a file to the system, they initiate a transaction on the blockchain that includes relevant metadata about the file, such as its name, size, owner, and a URL pointing to the location of the file in the cloud-based file store.

Decentralized and Immutable Ledger: The blockchain ensures that all transactions related to file uploads are recorded in a decentralized manner. Instead of relying on a central authority or server, the information is distributed across multiple nodes within the blockchain network, making it resilient to single points of failure and tamper-resistant due to its immutability.

Transaction Transparency: Since the blockchain is transparent, all participants within the network can see the details of each file upload transaction, enhancing trust and accountability.

Smart Contracts: Smart contracts can be utilized to automate specific actions or validation processes when certain conditions are met. For instance, a smart contract could be triggered to validate the authenticity of a file after it is uploaded.

Validating the Authenticity of the File:

To ensure the authenticity of the file itself, our system employs checksums (cryptographic hash functions) as part of the file upload process. When a user uploads a file, the system generates a checksum (also known as a hash) of the file's content. This checksum is a unique fixed-length string that acts as a digital fingerprint for the file.

The generated checksum is then stored as part of the metadata in the blockchain transaction associated with the file upload. This way, the checksum becomes an integral part of the blockchain's permanent record.

Whenever someone retrieves the file from the cloud-based file store, the system can recalculate the checksum of the file's content. If the recalculated checksum matches the original checksum stored in the blockchain, it means that the file has not been altered or corrupted since its initial upload. In case the checksums do not match, it indicates that the file's content has been modified, and the system can raise an alert or take appropriate actions to address the issue.

By combining the blockchain's decentralized and immutable properties with checksum-based authentication, our system ensures the integrity and authenticity of files throughout their lifecycle, providing users with greater confidence in the security and reliability of their data

5.2.1 Role of regulator and network participants

The proposed information and data exchange platform for power systems includes several key participants, each with specific roles and responsibilities.

Regulator: An administrator is responsible for controlling and managing the exchange of data within the power systems network. With the consent of enough network users, the administrator can exchange data with the requested user. The administrator is responsible for keeping the database in good condition and ensuring that when a user revokes their access, their data is removed from both the database and the organization's database. Additionally, the administrator conducts audits to ensure that no data is being stored illegally. This person has to be a reliable one. The proposed data exchange system for the power sector involves three key elements: Data Producers, the Exchange Platform, and Data Consumers.

Data Producers: TSOs, DSOs, and other market participants generate data and have the ability to upload it to cloud servers and the blockchain network for storage. They also have the option to retrieve data and share it with other network participants.

Exchange Platform: This platform utilizes cloud storage servers to store power system data. The blockchain network stores the index record of data such as the location of data storage, serving as an off-chain solution for data storage.

Data Consumers: Data Consumers are required to be registered with the regulator. The platform grants access to data based on the acceptance of requests, ensuring secure and controlled access to data while maintaining the network's privacy and security.

Figure 5-3 illustrates the system participants involved in the proposed information and data exchange platform for power systems.

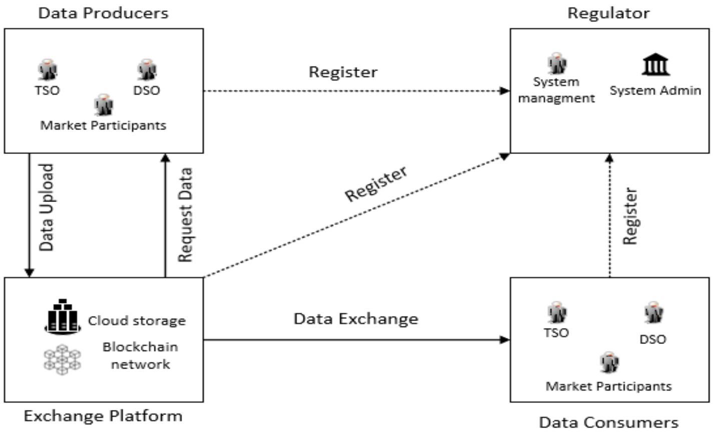


Figure 5-3 System participants

5.2.1 Off-chain storage

This chapter proposes the use of off-chain storage for the storage of data from power system entities. Data that is too large to be stored on the blockchain is passed to the Hadoop Distributed File System (HDFS) for storage. This off-chain storage method reduces the amount of storage required by each blockchain node on the network and reduces blockchain traffic, enabling more efficient archiving of the various types and levels of data in the system. Only file metadata, such as access time, modification time, and hash value, is stored using Hyperledger Fabric. The HDFS is responsible for managing the files, while the information is stored on the blockchain in the form of secured transactions, improving security and traceability. Users can always submit a query to the Hyperledger Fabric to read changes that have occurred to a specific file.

5.2.2 Smart contract

The proposed system utilizes smart contracts, also known as chain code, to perform essential functionalities on the network. Chain code is a section of code written in a supported language such as Java or Go [105], that is installed on the peers of the network and enables interaction with the shared ledger. The primary tasks of the chain code include logging network-provided consent data, requesting user consent information, and providing history data. Users can view a list of the companies with which they have shared data by using the ledger's history information, which acts as a log for them. Organizations must join the network and install the chain code on peers to use its functionalities. The chain code carries out certain operations when specific conditions are met.

The results of the transaction execution are then uploaded to the blockchain network and are connected to all of the peers' respective copies of the ledger. Smart contracts are automatically carried out when executed on a blockchain. Payments or other items of value can be exchanged by the terms of the contract if the provisions of the contract are satisfied. If the terms of the contract are not satisfied, payments may be withheld, if that provision is included in the smart contract. On a decentralized network of computers on the blockchain, smart contracts execute exactly as their programming directs them to, eliminating the risks associated with illegal alterations. Without the assistance of attorneys or the judicial system, the contract is automatically carried out, resulting in an exchange of value and payments between the parties involved. The timestamps of each entry on the blockchain, for example, are immutable and cannot be changed. This results in the creation of a platform that is suitable for contracts since any changes made to contracts are timestamped, and the blockchain stores prior versions of the contracts.

5.3 Network development

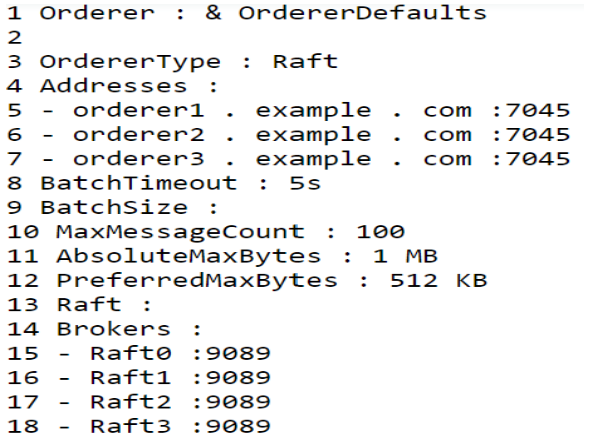
Table 5-1 lists the tools and technologies that are used in the proposed blockchain-based data exchange platform. Hyperledger Fabric 2.0, which is installed on a Linux machine, is the blockchain framework used in this proposed approach. There are VMs used, and each Hyperledger Fabric element is a Docker machine that is built into a Docker container. For the blockchain and HDFS to work together, a Representational State Transfer (REST) API is being built. Using the REST API, transactions are added to and removed from the distributed ledger. The API is made with Node.js and an open-source JavaScript library. The setup has one NameNode and one DataNode, which makes it a Single Node. Intel i7 CPU@3.00GHz and 16 GB of RAM are the specs for the hardware that is being used. In this network, only users who have been verified can change the information on the blockchain. See Appendix A for more details

Table 5-1 Tools for the proposed framework

Components	**Description**
Operating System	Ubuntu 22.02
CPU	Intel Core i7-3.00GHz
Memory	16 GB
Python	V2.8.12
Blockchain Network	Hyperledger Fabric
On-chain storage	CouchDB
Off-chain Storage	HDFS

There are a total of four nodes running in the VMs, and they are depicted in Table 5-1. Node 1 has 16GB of RAM and 4 VCPUs, while the remaining nodes each have 4GB of RAM and 1 VCPU, and all of them are running Ubuntu 22.02. In a Hyperledger fabric network, the various nodes are implemented as Docker containers running on separate VMs. The peers and clients/CLIs in the network are the players represented by the containers for these roles. The network will be crash fault resilient thanks to Node 0's three Orderer containers.

Contains information on the Orderer, such as the address of the Orderer and their port numbers, as well as whether the Orderer is "solo" or "Kafka" [106]. Batch Timeout and BatchSize are the parameters that the Orderer uses to specify the generation of blocks; the Orderer will use either of these two parameters to generate each block. A block will be generated regardless of which of these two parameters is reached first: the Batch Timeout or the Batch Size. Because of the application-specific nature of these configurations and the possibility that different networks will have distinctive policies on block configuration, Hyperledger did not supply any combination of these options. We based our configurations on the fact that we would like to have a balanced amount of data in each block. As a result, we selected 100 transactions, and in circumstances in which the transactions might take some time to reach the Orderer, we selected a delay of 5 seconds.

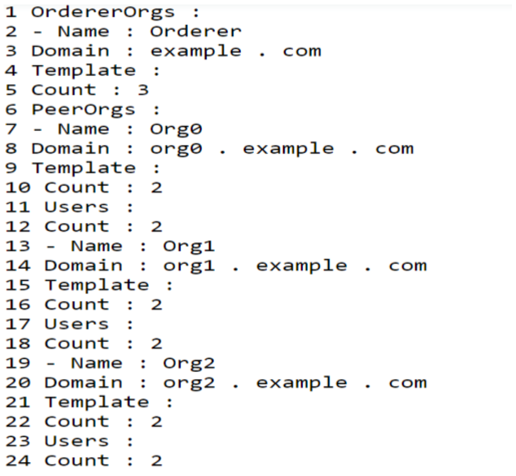


```
Orderer : & OrdererDefaults

OrdererType : Raft
Addresses :
- orderer1 . example . com :7045
- orderer2 . example . com :7045
- orderer3 . example . com :7045
BatchTimeout : 5s
BatchSize :
MaxMessageCount : 100
AbsoluteMaxBytes : 1 MB
PreferredMaxBytes : 512 KB
Raft :
Brokers :
- Raft0 :9089
- Raft1 :9089
- Raft2 :9089
- Raft3 :9089
```

Figure 5-4 Orderer setup of network

We do not want each block to take up an excessive amount of memory space because we would rather have more blocks with a balanced number of transactions than a few blocks with a lot of data. Because of this, the maximum block size has been set to 1 megabyte (MB), which will limit the block size. To provide the information required for each object to identify and communicate in the network, YAML files are implemented in Docker containers to represent the entities in the network. YAML is a human-readable data serialization format often used for configuration files and data exchange between programming languages. It uses indentation to represent data structures, making it easier for humans to read and write compared to XML. The yaml files allow us to configure each entity for a specific task; for example, we can map crypto-material to a container so that the docker client knows how to identify itself to the network, open channels, list its peers, and initiate or query transactions.



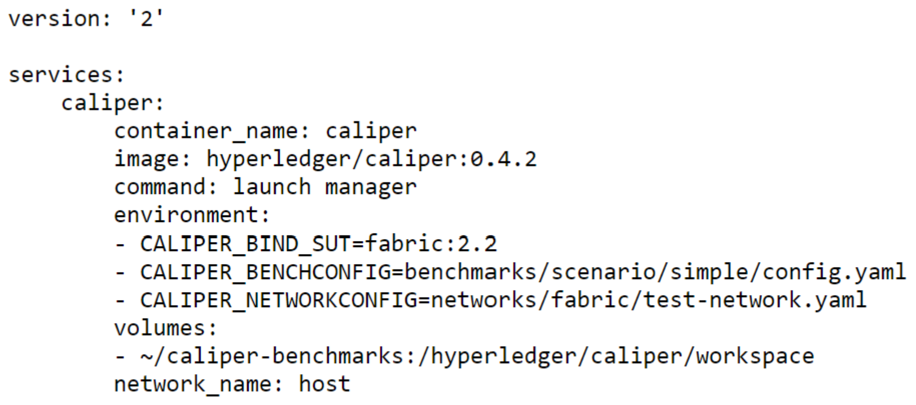
```
OrdererOrgs :
- Name : Orderer
Domain : example . com
Template :
Count : 3
PeerOrgs :
- Name : Org0
Domain : org0 . example . com
Template :
Count : 2
Users :
Count : 2
- Name : Org1
Domain : org1 . example . com
Template :
Count : 2
Users :
Count : 2
- Name : Org2
Domain : org2 . example . com
Template :
Count : 2
Users :
Count : 2
```

Figure 5-5 Crypto-material file

A replica of the produced folders containing the crypto-material, container deployment files, and channel files will be sent to each VM. Chaincode will be distributed to nodes that aren't part of the Orderer network so that they can participate in simulations and invoke transactions. After the containers have been set up, a participant can start a transaction to create a channel. Before invoking or querying a transaction, the network must first install and instantiate the chaincode, which saves resources like processing time and data storage. All of the peers run the same version of the chaincode, and the chaincode is instantiated just once on the channel. When you install chaincode, a docker container is created, which keeps any potentially harmful or faulty code contained.

5.3.1 Hyperledger Caliper

Hyperledger Caliper is a powerful tool for evaluating the performance of blockchain implementations and comparing them to established benchmarks [107]. It provides detailed performance reports and various indicators. Caliper is compatible with different versions of Hyperledger Fabric SDK, such as 1.1.0, 1.4.11, 2.1.0, and the most recent version. This tool can be used to measure the performance of a blockchain platform and compare it with industry standards [107].

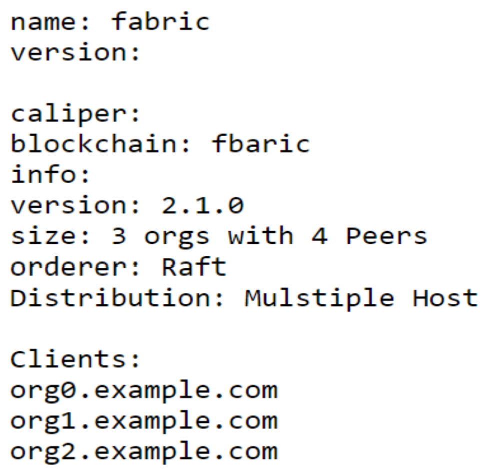


```
version: '2'

services:
    caliper:
        container_name: caliper
        image: hyperledger/caliper:0.4.2
        command: launch manager
        environment:
        - CALIPER_BIND_SUT=fabric:2.2
        - CALIPER_BENCHCONFIG=benchmarks/scenario/simple/config.yaml
        - CALIPER_NETWORKCONFIG=networks/fabric/test-network.yaml
        volumes:
        - ~/caliper-benchmarks:/hyperledger/caliper/workspace
        network_name: host
```

Figure 5-6 Hyperledger Caliper Setup

The Hyperledger Caliper tool utilizes a YAML file called the network-config file to create the configuration file for the network. This file is customized to meet the specific configuration needs. An illustration of the network configuration used to establish a connection to Caliper is shown in Figure 5-7.



```
name: fabric
version:

caliper:
blockchain: fbaric
info:
version: 2.1.0
size: 3 orgs with 4 Peers
orderer: Raft
Distribution: Mulstiple Host

Clients:
org0.example.com
org1.example.com
org2.example.com
```

Figure 5-7 Network Configuration for Connection with Caliper

After the configurations are completed, the Docker container is launched. The process began with the creation of two test cases, one for receiving data from the network and the other for simultaneous reading and writing of data to the network. The rates for both test cases were maintained consistent throughout the development.

5.4 Performance metrics

To measure the performance of a blockchain network, throughput, and latency are important metrics that are used. Both indicators are often used to compare different blockchain platforms and consensus mechanisms. Throughput can be further divided into two categories: "read throughput" and "transaction throughput". Read throughput is a measurement of the number of read operations completed in a given amount of time, typically expressed as "reads per second" (rps). It is not commonly used as a primary performance metric for evaluating the performance of a blockchain network, but rather as a supplementary metric to assess the overall efficiency of the system when combined with other technologies. On the other hand, transaction throughput is the rate at which valid transactions are committed by the blockchain within a specified period and is expressed as tps.

$$\text{Transaction Per Second} = \frac{\text{Total successful Transaction}}{\text{Time(sec)}} \qquad (1)$$

Transaction latency and read latency are two different classes of latency in blockchain systems. Read latency is a metric that captures the time elapsed between a read request being sent and the receipt of the response. It indicates how long it takes for a read operation to be completed. It is a measure of how quickly data can be retrieved from the system. It is typically measured in milliseconds.

Transaction latency, on the other hand, refers to the overall time it takes for a transaction to be validated by the entire network, including the duration of broadcasting and consensus allocation processes. It also includes the network threshold, or the time required for the network to confirm a transaction. This time is measured in seconds.

Both read latency and transaction latency are key performance indicators used to measure the efficiency and scalability of a blockchain system. A low read latency and transaction latency indicate a system that is fast and responsive, while high latencies can indicate scalability issues or bottlenecks in the system [108]. This time is represented in seconds.

Furthermore, Equation 2 [109] specifies how to calculate the Read/Write rate per second about the total number of tasks performed.

$$\text{Read/Write Rate} = \frac{\text{Read/Write Total transactions}}{\text{Time(sec)}} \tag{2}$$

It is important to note that the equation for calculating read/write latency (Equation 3) was not provided in the previous text. To evaluate the proposed framework, it would be necessary to have the specific equation or method used to calculate this metric. Additionally, it would be helpful to have further information on how the framework was evaluated, such as the specific parameters used and any results or conclusions that were drawn from the evaluation.

$$\text{Read/Write Latency} = \text{Response time received} - \text{Request time} \tag{3}$$

Network latency is defined as the total time taken for a transaction to be approved, including the time taken for the nodes to reach a consensus. Equation 4 would provide the specific method for calculating this metric.

$$\text{Transaction Latency} = \text{Commitment time} - \text{Request time} \tag{4}$$

5.5 Experimental results

Multiple experiments are performed in this section to assess the performance of our approach to exchanging data within a power system using a blockchain network.

5.5.1 Read/Write rate test

The proposed approach's performance is evaluated by comparing the storage of data using an on-chain approach versus an on/off-chain approach. The results of this comparison are illustrated in Figure 5-8 and Figure 5-9. The analysis shows that when a large amount of data needs to be stored, distributed data storage (using the on/off-chain approach) is more efficient than non-distributed data storage (using the on-chain approach) in terms of reading and writing data. The use of a distributed storage system allows for each piece of data to be stored individually, thus reducing the volume of data on the blockchain network. Additionally, the distributed database is maintained locally, which eliminates any storage capacity limitations, enabling easy scalability. This separate storage scheme separates the data storage from the blockchain network, which minimizes the amount of data stored on the blockchain. This provides several benefits such as improved system scalability and efficient use of storage capacity. By storing data off-chain, the system can handle a larger amount of data without overwhelming the blockchain network. This means that the system can continue to function effectively as the amount of data grows.

The proposed approach combines the strengths of both blockchain and Hadoop, by utilizing the decentralization and security of blockchain and the storage and processing capabilities of Hadoop. By integrating these two technologies, the proposed approach provides a balance of security and efficiency. While this approach may be less secure than keeping data solely on a blockchain network, it is still more secure than traditional centralized data storage methods. The core concept of blockchain technology is to store data in a decentralized manner to provide data security. This approach ensures that the data is kept safe and private by distributing it across a network of nodes, rather than storing it in a central location. The decentralized architecture of this structure makes it challenging for unauthorized parties to access the data and also enhances its resistance to tampering and data breaches.

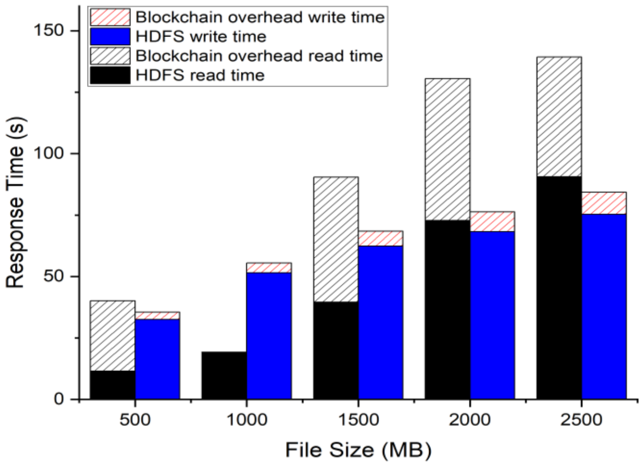


Figure 5-8 Read and write rate of the proposed platform

The results of the performance evaluation of the proposed approach are displayed in Figure 5-8 and Figure 5-9. Figure 5-8 illustrates the response time for writing and reading data of our proposed platform with 4 users involved in the process. The response time was measured by generating fixed file sizes ranging from 500 to 2500 MB. The black bar in the figure represents the time taken when the file is stored in the Hadoop Distributed File System (HDFS). The shaded portion of the black bar represents the additional time required when blockchain technology is used to write the file, while also validating the transaction. It can be observed that using blockchain technology incurs some overhead in terms of performance when writing data. However, as the file size increases, the proportion of the blockchain overhead decreases. This is mainly because HDFS takes more time to write larger files. The use of blockchain technology adds an extra step of validation and security, which increases the response time slightly. However, as the file size increases, the proportion of this overhead decreases about the time taken by HDFS to write the file.

The blue bar in Figure 5-8 represents the read time for the file. The shaded portion in red of the bar represents the overhead time when the blockchain is used to read the file while validating the transaction against the hashed value stored on Hyperledger. It can be seen that the blockchain overhead in reading the file is much less compared to writing the file using blockchain. The proportion of blockchain overhead decreases as the file size increases. This suggests that while reading files, the overhead of using blockchain is relatively insignificant and it can be used to ensure the integrity of the data being read without introducing a significant overhead. Figure 5-9 illustrates the read and write rate of data of our proposed platform with only 2 users. The black bar in the figure represents the rate at which the file is written to the Hadoop Distributed File System (HDFS). The shaded portion of the bar represents the overhead when blockchain technology is used to write the file into HDFS. This overhead is caused by the additional steps involved in using the blockchain, such as creating and validating transactions, updating the shared ledger, and ensuring data integrity through the use of checksums. This overhead may slow down the writing process, but it also provides a higher level of security and data integrity. It can be seen from the figure that the rate of reading and writing data is affected when blockchain is used, but it is not specified in the text provided how significant the difference is.

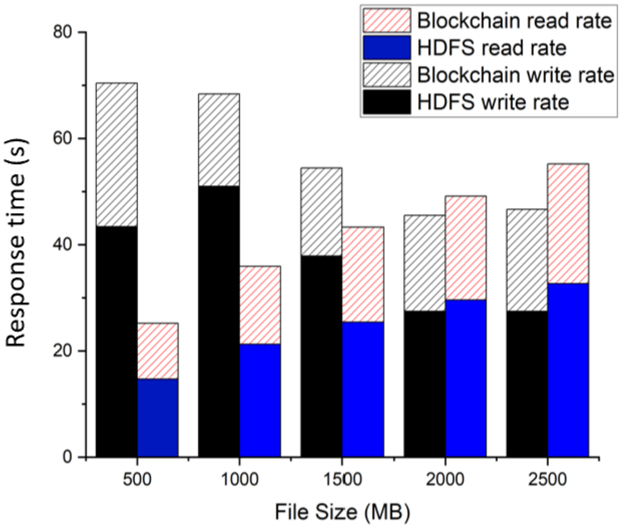


Figure 5-9 Read and write of the proposed platform

As shown in Figure 5-10, the performance of both HDFS and blockchain is evaluated by comparing their throughput while varying the number of nodes and using a file size of 1500 MB. A replication factor of 3 is applied to HDFS. The results indicate that as the number of nodes increases, the TPS (transactions per second) for both HDFS and blockchain also increases, meaning that higher throughput can be achieved by adding more nodes. This linear scaling relationship suggests that blockchain can be a viable framework for data exchange in power systems. However, it's worth noting that the ratio between HDFS and blockchain performance may be lower when using just 2 nodes, potentially due to variations in network latency caused by unreliable network connections. Therefore, to ensure scalability and reliability, it's essential to conduct performance testing under various scenarios.

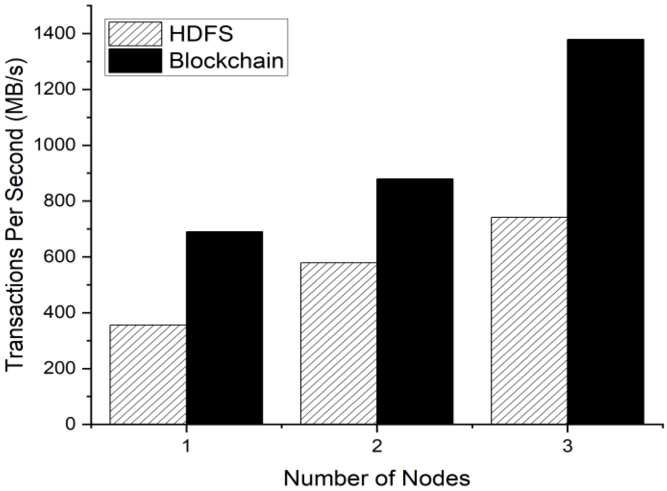


Figure 5-10 Performance evaluation of the proposed approach

5.5.2 Transaction throughput and latency test

This section presents the results of the evaluation of various customizable parameters. To examine the impact of block size on the performance of blockchain networks, experiments were conducted using three different transaction send rates and three different block sizes (35, 65, and 100). The performance was measured in terms of transaction throughput, which varied from 30 to 300 tps. The results of these experiments are shown in Figure 5-8, which illustrates the average transaction throughput. The graph indicates that as the transmit rate increases, the transaction throughput also increases linearly until it reaches around 210 tps. After this point, the increase in transaction throughput slows down significantly and approaches a level state. When the send rate was set to 300 tps, the throughput at every block size was found to be 290.4 tps, 270.3 tps, and 266.5 tps respectively.

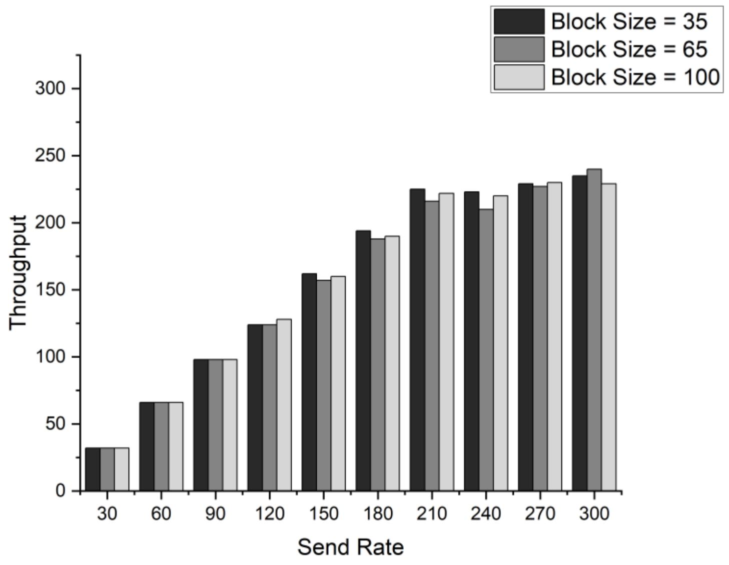


Figure 5-11 Transaction Throughput with one TSO and one DSO

The results of the experiments are illustrated as a plot in Figure in terms of the average transaction latency. The amount of time it takes for a transaction to complete decreases when the send rate is increased. For example, the latency dropped from 166.5 to 110.3 ms with the block size at 35 with the send rate increasing from 30 to 120. The findings of this experiment suggest that the configuration options for the block size have a minimal impact on the performance. There is only a slight but noticeable improvement in application performance when using a smaller block size, both in terms of throughput and latency.

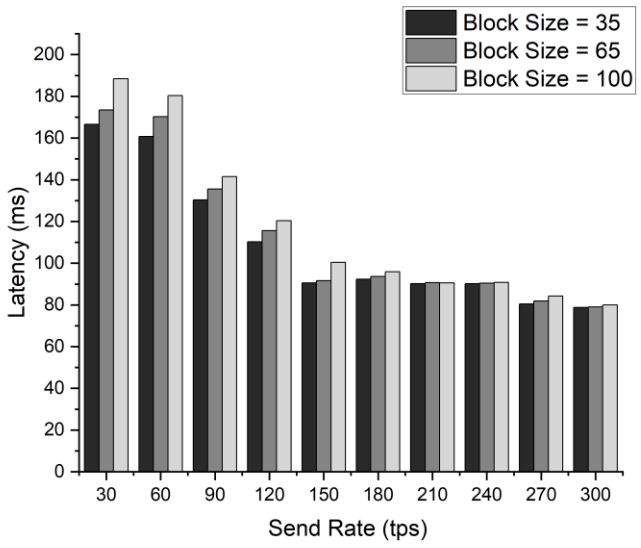


Figure 5-12 Transaction Latency with one TSO and one DSO

Hardware bottlenecks can have a profound impact on the performance of experiments, especially in cloud-based environments. The limit of the number of CPUs can be highly significant, and the performance of virtual machines (VMs) can vary based on the allocation of CPUs. Let's explore these aspects in detail:

1. Impact of Hardware Bottlenecks:

a. CPU Bottlenecks: If the CPU resources allocated to VMs are insufficient for the computational demands of the experiments, it can lead to performance degradation. Tasks that require significant CPU processing may experience delays, resulting in longer execution times.

b. Memory Bottlenecks: Insufficient RAM can lead to memory bottlenecks, where the VMs are forced to swap data between RAM and slower disk storage. This can significantly slow down data-intensive tasks and reduce overall system performance.

c. Disk I/O Bottlenecks: Slow disk I/O operations can affect the speed at which data is read from or written to storage. If the experiments involve frequent disk I/O, a bottleneck in this area can lead to delays and reduced performance.

d. Network Bottlenecks: In cloud-based environments, network bottlenecks can occur if data transfer rates between VMs or with external resources are constrained. This can affect data exchange and communication, impacting the overall experiment performance.

2. Significance of the Number of CPUs:

The number of CPUs allocated to VMs is crucial, and its significance depends on the nature of the workload:

a. CPU-Intensive Workloads: Workloads that are highly CPU-intensive, such as complex simulations or data processing tasks, benefit significantly from having more CPUs. An increase in CPU resources can lead to shorter execution times and improved performance.

b. Parallel Processing: Some tasks can be parallelized to take advantage of multiple CPUs simultaneously. VMs with more CPUs can process multiple tasks in parallel, which can significantly enhance performance for parallelizable workloads.

c. Diminishing Returns: However, it's essential to recognize that adding more CPUs does not always lead to proportional performance gains. The efficiency of parallelization depends on the specific workload and the software's ability to utilize multiple cores effectively.

3. VM Performance with More CPUs:

The performance of VMs when given more CPUs depends on several factors:

a. Workload: If the workload is CPU-bound and can effectively use multiple cores, increasing the number of CPUs will likely result in improved performance and reduced execution times.

b. Software Optimization: The software being used should be optimized for multi-threading and parallel processing to make effective use of additional CPUs. Not all applications or tasks can efficiently utilize multiple cores.

c. Resource Balance: Proper resource allocation is crucial. Increasing the number of CPUs should be balanced with sufficient memory and I/O resources. Over-allocating CPUs without addressing memory or disk bottlenecks may not yield expected performance gains.

d. Testing and Benchmarking: To determine the optimal number of CPUs for your specific workload, thorough testing and benchmarking are necessary. Performance may vary based on factors like data volume and task complexity.

In conclusion, hardware bottlenecks and the allocation of CPUs can significantly affect the performance of experiments in a cloud-based environment. Adding more CPUs can improve performance for CPU-intensive and parallelizable workloads, but it must be done strategically and in conjunction with other resource optimizations to avoid creating new bottlenecks.

5.6 Chapter Summary

In this chapter, a new method for information and data exchange among different entities within power systems was presented. The proposed approach combines blockchain and big data technologies like Hadoop Distributed File System (HDFS) to create a secure and efficient platform for data exchange.

Most of the present data exchange platforms are centralized, which can lead to vulnerabilities such as single points of failure, malicious attacks, and data alteration. The proposed approach, however, utilizes the decentralized nature of blockchain technology to address these issues. It uses a decentralized consensus system to ensure the security of data transactions and prevent unauthorized access.

The evaluation of our proposed approach suggests that it can be implemented with an acceptable level of overhead. Furthermore, experimental results indicate that this platform is effective in sharing data and information in a power system. The use of Hadoop allows for off-chain storage which provides additional benefits such as improved scalability and efficient use of storage resources. This platform provides a secure and efficient solution for exchanging among TSOs, DSOs, and other stakeholders in the power systems industry.

Chapter 6 Performance Optimization of the Blockchain-Enabled Data Exchange Platform for Power System

6.1 Introduction

As previously mentioned in Chapter 5, permissioned blockchain platforms like Hyperledger Fabric have faced criticism for performance issues and scalability concerns. Hyperledger Fabric is made up of various components like endorsers, ordering services, and committers among others. It also includes multiple phases in the processing of a transaction such as the commit phase, validation phase, ordering phase, and endorsement phase. Due to the many components and stages it contains, Fabric has a wide range of adjustable characteristics, including block size, block interval, endorsement policy, channels, and state database. Therefore, determining the appropriate values for each of these factors is one of the most challenging aspects of building an effective blockchain network.

Hyperledger Fabric's performance can be limited by its low throughput and high latency. The rate at which the blockchain network commits valid transactions during a given period is referred to as transaction throughput, measured in the number of transactions per second (TPS). Transaction latency, on the other hand, is the amount of time that passes between the initiation of a transaction and its confirmation as committed across the network. According to Swan [110], there are several technological challenges associated with the adoption of blockchain technology, including throughput, latency, size and bandwidth, security, wasted resources, usability, versioning, and hard forks. Latency and throughput refer to the time taken for a transaction to be processed and the number of transactions that can be processed per second, respectively. While there has been a significant amount of research on blockchain technology, latency, and throughput are still considered to be key challenges that have not been widely studied [111]. This study was based on the difficulties mentioned above and the fact that a scalability study is necessary as it is expected that implemented blockchain frameworks will require a high number of nodes [111]. The article highlights the efforts being made by academics to optimize the blockchain system to improve its scalability. It mentions that research conducted by a source [112] focused on optimizing the block construction process, transaction security mechanism, block size, and time control to improve the scalability of private blockchain. Furthermore, it notes that the use of deep reinforcement learning was proposed by Liu et al. [113] to maximize the scalability of the blockchain-based industrial Internet of

Things (IoT). It also refers to other recent solutions that address blockchain scalability from a storage perspective, proposing storage optimization schemes to reduce the storage demand of peers [114,115]. These solutions are referred to as "storage-first" approaches. Additionally, the passage mentions that very few papers have attempted to evaluate network optimization in conjunction with blockchain [116-118].

In this chapter, an optimization technique was proposed to enhance the performance of blockchain-enabled information and data exchange for power systems. The framework includes the application of a machine learning technique to the proposed data exchange platform to enhance the performance of our blockchain platform. It explains how the proposed approach interacts with an external machine learning module to optimize the performance. The proposed approach has an extensible architecture that supports an interface to interact with different external machine-learning modules. The proposed approach was tested using case studies of data exchange between power systems entities, which indicates that the implementation of machine learning techniques can be useful to improve the performance of the blockchain network by analyzing the data and optimizing the network's parameters in real time.

6.2 Proposed optimization mechanism based on ANN

Figure 6-1 illustrates the design of using an artificial neural network (ANN) to enhance the performance of our blockchain network. The blockchain system is composed of multiple nodes that serve as hosts for smart contracts and maintain copies of the distributed ledger to maintain the network's stability. The ANN-based prediction module is separate from the blockchain network and can be linked to it. Users can submit transactions by utilizing the functions outlined in the smart contract. The network's performance is tracked in real-time and receives these measurements. The ANN module is implemented to enhance the overall performance of the blockchain network. The network administrator can adjust the configuration based on the predicted throughput and latency values after each test, and testing will stop once optimal conditions are reached.

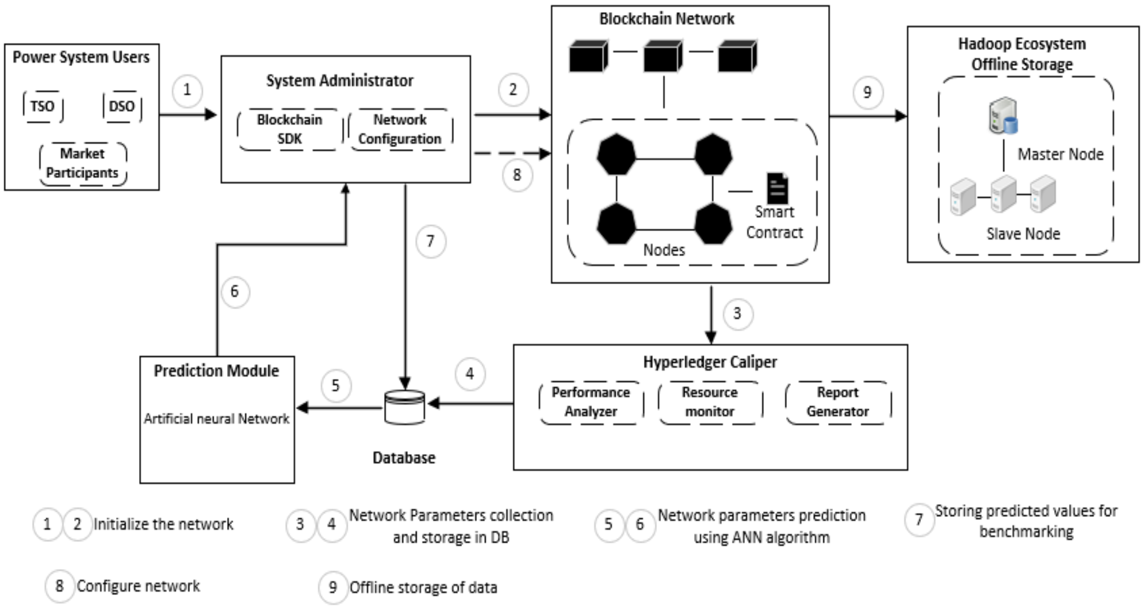


Figure 6-1 The architecture of the ANN-based performance optimization of blockchain

6.3 Development of the ANN-based optimization mechanism

An Artificial Neural Network (ANN) is a computer system modelled after the human brain that is made up of many interconnected processing elements, known as neurons, which work together to solve a problem [119]. ANNs are built with weighted, directional graphs as their architecture, where the artificial neurons are nodes, and the arrows and weights represent the relationship between the neurons' inputs and outputs. ANNs can be broadly classified into two categories based on their architecture: feed-forward networks and recurrent networks. Feed-forward networks are networks in which the information flows in one direction from the input layer to the output layer. Recurrent networks, on the other hand, have feedback connections allowing information to flow in cycles, which allows them to maintain a state and process sequences of inputs [120]. In this research, a feed-forward ANN is used because it can handle non-linear data. Different configurations of the ANN are tested to find the best training method by adjusting the number of neurons in the hidden layer, the learning rate, and the activation function. The experiments were conducted multiple times for each network configuration to train and the average results were recorded to analyse the random factor for initializing the weights of the ANN network. First, the data is imported, and initial pre-processing is performed, which includes checking for missing values and providing a data description. Afterward, labels are assigned to the data as optimal and non-optimal classes. The data is then divided into three categories before being input into the network: 15% for validation, 70% for training, and 15% for testing. The network configuration for training is set up with 3 inputs, 20 neurons in the hidden layer, and two outputs. The benchmark results are examined by the optimization module, which is then run repeatedly to get the optimal network parameters. These results are then provided to the system administrator to update the system configurations based on the predicted throughput and latency. This learn-to-predict model is executed outside of our blockchain network.

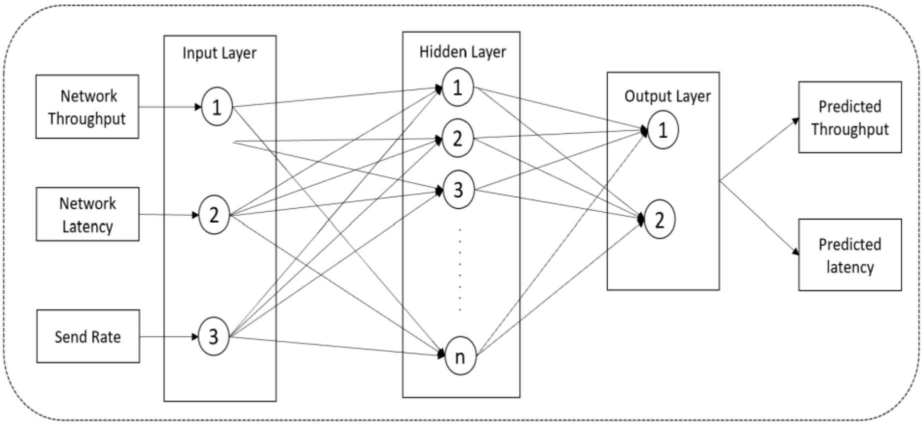


Figure 6-2 The detailed structure of the Learn to Predict model

6.4 Parameter effecting blockchain performance

Several parameters can affect the performance of a blockchain network, including:

- Block size: The size of the blocks in a blockchain can affect the speed and efficiency of the network. Larger block sizes can lead to faster transaction processing times, but they can also increase the risk of network congestion and slow down the network.
- Consensus mechanism: The consensus mechanism used by a blockchain network can have a significant impact on its performance. Some mechanisms, like Proof of Work (PoW), can be more resource-intensive and slower than others, like Proof of Stake (PoS)
- Network latency: The time it takes for transactions to be propagated across the network can affect the overall performance of a blockchain. High network latency can lead to slower transaction processing times and increased risk of network congestion.

- Node count: The number of nodes present in a blockchain network can affect its performance. While adding more nodes can increase security and decentralization, it can also lead to increased network congestion and impede the processing of transactions.
- Mining difficulty: The mining difficulty level of a blockchain can affect its performance. Higher mining difficulty can increase the security of the network, but it can also lead to slower block confirmation times and higher costs for miners.
- Transactions per block: A blockchain network's performance is impacted by the amount of transactions that can be handled in each block. A higher transaction per block can lead to faster transaction processing times, but it can also increase the risk of network congestion.
- Hardware and software infrastructure: The underlying hardware and software infrastructure of a blockchain network can also affect its performance. A well-designed, high-performance infrastructure can help to ensure fast and efficient processing of transactions.

6.5 Experimental setting

We carried out extensive experiments on our proposed blockchain-enabled data exchange platform. The results of the baseline scheme and our scheme were obtained and compared. The table shows the tools used for building the proposed architecture. Hyperledger Fabric is an open-source blockchain framework that was developed by the Linux Foundation. The docker engine was utilized to create VMs on which each Hyperledger fabric was embedded in a docker image. Additionally, the Hyperledger caliper was integrated with our blockchain network to collect information about the blockchain network. An Artificial Neural Network (ANN) was used as the learn-to-predict module to predict transaction throughput and latency to find the optimal configurations for the network. The Hadoop HDFS file system was used to store the information and data exchange off-chain for better performance. A non-SQL database Mongo DB was used to store benchmark results from Hyperledger fabric for the prediction module. The configuration parameters used in this proposed model are presented in Table 6-1. These parameters were used to improve the performance of the proposed blockchain-enabled data exchange platform for the entities of the power system. See Appendix A for more details.

Table 6-1 Development environment of the proposed framework

Components	**Specification**
Docker Engine	20.10.17
Docker Composer	1.29.2
CPU	Intel Core i7-3.00GHz
Memory	16 GB
Operating System	Ubuntu 20.4
Node SDK	Node.js
Blockchain Platform	Hyperledger Fabric
Programming Language	JavaScript
DBM	MongoDB

6.6 Performance evaluation

For evaluating the performance, an open-source tool Hyperledger caliper is used for the proposed framework. The performance is measured in terms of network throughput and latency. The configuration parameters of the proposed framework are presented in Table 6-2. These configuration parameters are used to enhance the performance of the proposed blockchain network.

Table 6-2 Experimental setup parameters for a blockchain configuration

Parameter	**Value**	**Description**
Block Size	128,512 KB	Max block size
Block Interval	250, 300, 350 ms	Time to create a block
TSOs	1	Transmission System Operators

DSOs	5	Distribution System Operators
Internal Database	CouchDB	Ledger data storage
External Database	HDFS	Off-chain data storage
Ordering Service	PBFT	Transaction order

The participants of the network are TSOs and DSOs. Users who can submit transactions to the blockchain network are referred to as Participants. To evaluate the capabilities of the blockchain network, a simple smart contract is utilized. A new block is created every 250 ms, and the default block size is set to 100 transactions per block. The default ordering service operates in Raft mode and has only one ordering node. In this experiment, CouchDB is used as the default state database. To reduce inaccuracies caused by network congestion, the evaluation tests presented in this section were averaged over multiple rounds.

6.7 Experimental results

In this section, we evaluate the effectiveness of our method for optimizing the blockchain network by comparing it with a baseline scheme. We compare the results of using our learning-to-predict method with the results of using the baseline network. We use sample data to evaluate the network's transaction throughput. Figure 6-3 illustrates a comparison of the network's performance. The performance is tested using different send rates ranging from 30 to 400. It is observed that the transaction throughput increases linearly about the send rate until it reaches around 210 tps. Beyond this point, the increase in transaction throughput slows down and becomes steady. The transaction throughput of the network improves by 23.9% to 153.2 tps and 201.4 tps, respectively, when the transmit rate is set to 175 tps, indicating that our proposed method is effective in improving the performance of the blockchain network.

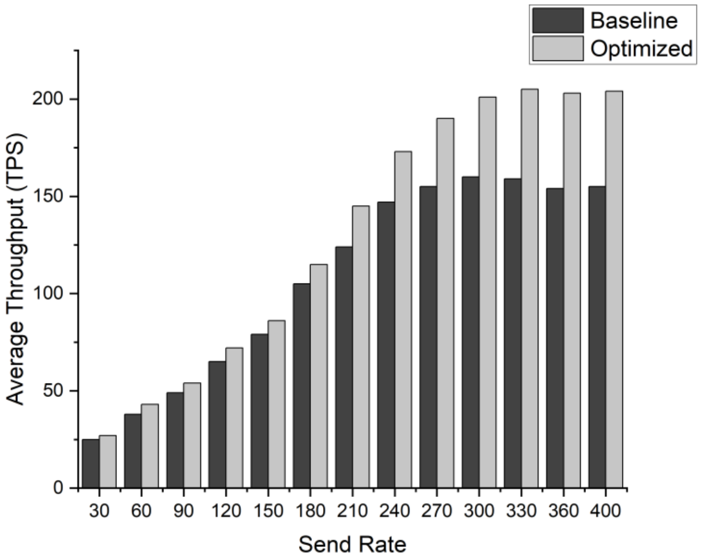


Figure 6-3 Average Transaction Throughput with one TSO and one DSO

In Figure 6-4, we can see a comparison of the network's performance when utilizing our proposed mechanism, which is based on learning to predict, the performance of the standard network, at varying transaction send rates. The transaction latency of the learning-to-prediction mechanism is 77 ms, while the baseline network has a latency of 102 ms. This demonstrates that the transaction latency is improved by 21% when the sending rate is set to 400 tps.

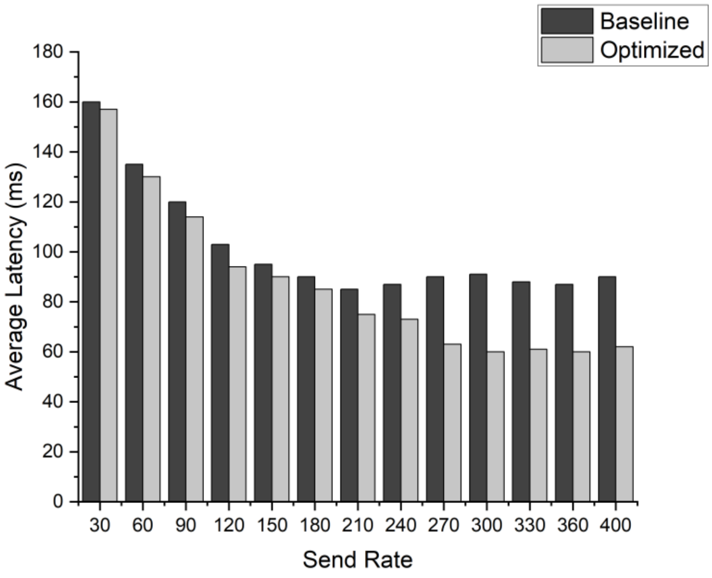


Figure 6-4 Average Transaction Latency with one TSO and one DSO

Figure 6-5 and 6-6 compares the network latency and throughput of the proposed blockchain performance improvement mechanism based on learn-to-predict with the baseline network with one TSO and two DSOs at varying transaction send rates between 30 and 400 tps. In this experiment, it can be seen that transaction throughput scaled linearly with a send rate up to about 210 tps. The transaction throughput of the learning-to-predict mechanism and the baseline were 140 and 179.3 tps, respectively, when the send rate was set to 400 tps, representing a 21.9% increase in transaction throughput). Similarly, when the send rate was adjusted to 400 tps, the transaction latency of the learning-to-predict mechanism and the baseline were 115 and 91 ms, respectively, representing a 20.8% reduction in transaction latency.

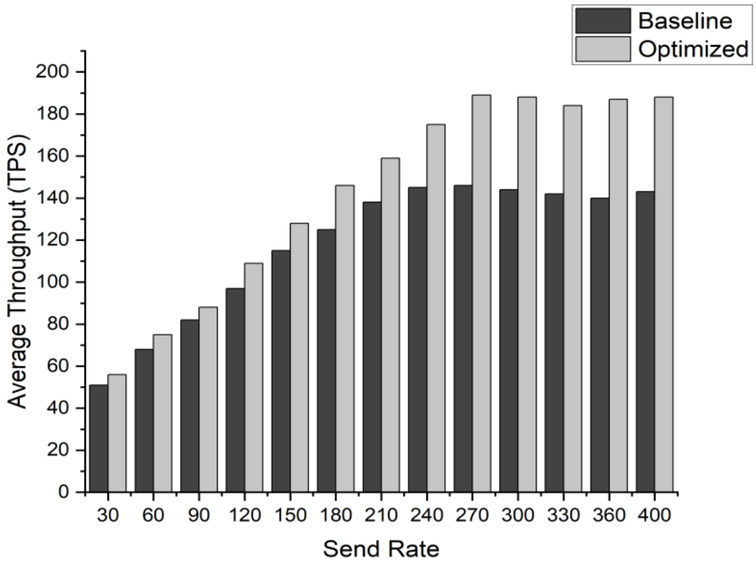


Figure 6-5 Average Transaction Throughput with one TSO and two DSO

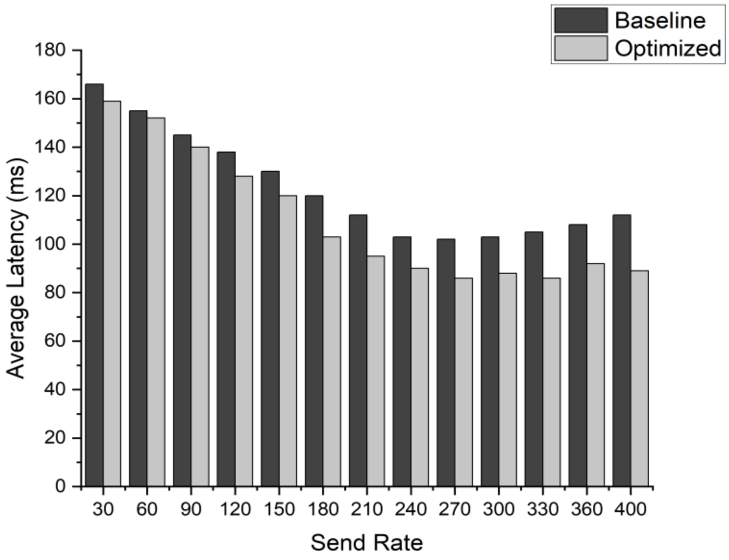


Figure 6-6 Average Transaction Throughput with one TSO and two DSO

Figure 6-7 compares the network's performance when using our proposed mechanism based on learning to predict, with the performance of the baseline network, at different transaction send rates to evaluate transaction throughput with one TSO and five DSOs (range from 30 to 400 tps). There is a linear increase in the transaction throughput with an increasing send rate until it reaches approximately 210 tps. As the send rate surpasses this threshold, the increase in transaction throughput slows significantly, and the system reaches a saturation point. The transaction throughput for the learning-to-predict mechanism was 221.5 tps, and for the baseline network, it was 263.4 tps. This means that when the send rate is set to 400 tps, the proposed mechanism leads to an 18.9% increase in transaction throughput as compared to the baseline network.

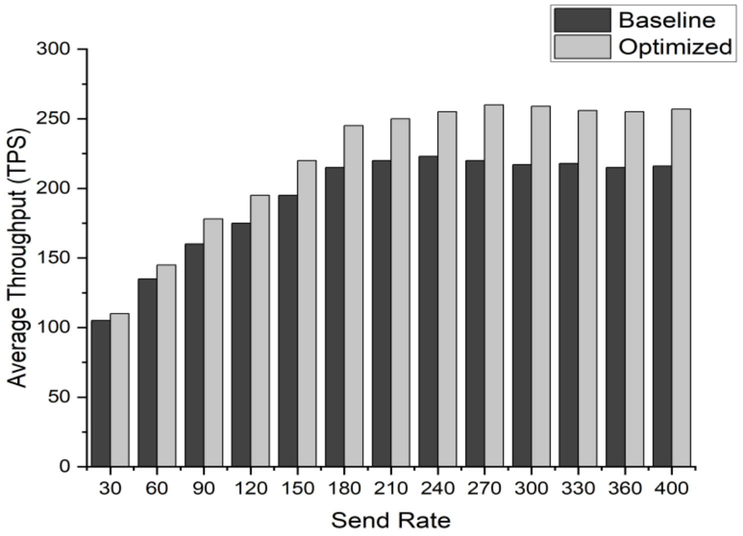


Figure 6-7 Average Transaction Throughput with one TSO and five DSO

Figure 6-8 compares the network's performance when using our proposed transaction traffic control mechanism based on learning to predict the performance of the baseline network, at different transaction send rates to evaluate transaction latency with one TSO and five DSOs (range from 30 to 400 tps). It can be seen from Figure 6-6 that when the send rate is set to 400 tps, the transaction latency of the learning-to-predict mechanism is 1200 ms, and the baseline network has a transaction latency of 790 ms, respectively. This means that the proposed mechanism leads to a 34.1% reduction in transaction latency as compared to the baseline network.

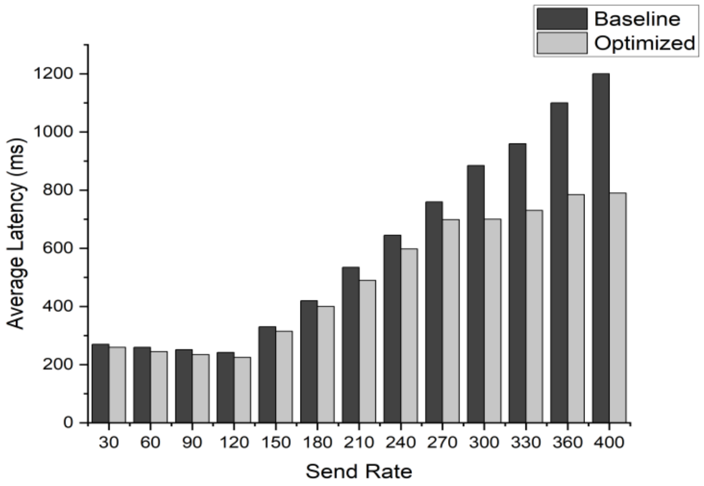


Figure 6-8 Average Transaction Throughput with one TSO and five DSO

The results of the experiments in this section demonstrate that the proposed strategy is effective for sharing data and information between different entities of the power system. We implemented a case study of a blockchain network for data exchange within the power system, using a permission-based blockchain network called Hyperledger Fabric. This study introduces a learn-to-predict mechanism for improving the performance of the blockchain network. The results of the case study indicate that the system performance is improved by increasing the network throughput and reducing the latency.

6.7 Chapter Summary

This chapter presented a performance improvement mechanism for blockchain based information and data exchange for power system entities used to predict the ANN model. Various experiments were performed to assess the effectiveness of the proposed approach. The experimental results indicate that the proposed approach is suitable for data exchange within power systems, and the performance of the blockchain network is augmented when combined with the learn-to-predict model. The results indicate that the overall throughput is improved with the increasing send rate and the network latency is reduced.

Chapter 7 Conclusion and Future Research

7.1 Conclusion

This part concludes the thesis by reviewing its most important contributions and discoveries. The following are the summaries of the three primary works included in this thesis:

Cloud-based data exchange platform: The newly designed cloud platform is intended to facilitate data exchange among various actors in the power sector such as TSOs, DSOs, consumers, grid users, producers, and aggregators. This will increase interoperability among a wide range of actors in the energy sector. The use of cloud computing in this platform has many benefits, such as scalability and reliability. In comparison to existing data exchange platforms in the energy sector, the new cloud platform is equipped to handle a larger number of services, include more actors, and transfer a larger volume of data. Additionally, the platform is evaluated for scalability and reliability to ensure that it is suitable for its intended purpose of priority information, data needs, and interfaces. This evaluation is based on a series of experiments, as well as an analysis of service deployments and VM cluster architecture.

Blockchain-enabled secure and transparent data exchange: We have proposed a system for information exchange that prioritizes trustworthiness, transparency, and accountability for data breaches. This system combines the use of blockchain technology and big data tools such as HDFS. The decentralized consensus framework of blockchain technology is particularly useful for verifiable data transactions. This proposed platform can enhance TSO-DSO interoperability, resulting in better system performance in terms of both supply reliability and congestion control. The blockchain-based approach eliminates the need to rely on a single entity for controlling access to encrypted data and utilizes chaincodes to store and evaluate access control policies on the chain. The experiments indicate that the proposed platform scales linearly with the number of nodes. Analysis shows that the platform can use a private blockchain with minimal overhead for communication between various power system entities. The experimental results also demonstrate the proposed platform's feasibility for data and information sharing in a power system.

Blockchain performance optimization: Performance optimization is becoming increasingly crucial in the development of new blockchain technology as it is being integrated into various applications. In this thesis, an ANN-based learn-to-predict model was proposed in Chapter 6 for performance optimization of the blockchain platform. The method was used to predict the network's throughput and latency, allowing the network administrator to adjust the network's settings to achieve high throughput and low latency. The results indicate that the proposed model was successful in improving the performance of blockchain-enabled information and data exchange, and able to adapt to the dynamic nature of power systems.

7.2 Future research

This thesis has made significant contributions to the analysis of various ICT tools for information and data exchange within power systems. However, there are still some research gaps that need further exploration. One such area is the scalability of the platform, which could be demonstrated through a comprehensive demonstration of additional services on a VM cluster that includes multiple VMs. Another potential area of research is the optimal allocation of VMs, services, and market participants. Future research could also focus on developing a variety of applications such as load forecasting, electric vehicle charging planning, and power system optimization that can be added to the TSO-DSO use cases on the designed data exchange cloud platform. This is because the platform is designed to facilitate the exchange and storage of a variety of data types.

This study has shown that blockchain technology can be effectively employed for exchanging information and data among entities in a power system. However, there are several areas for future research to improve upon this. One area of improvement would be to compare the performance of Hyperledger Fabric with other enterprise blockchain platforms in terms of throughput, scalability, and latency. Another area of exploration could be to assess the scalability of the system in a configuration where each node is installed on a separate physical or VM, both horizontally and vertically.

In addition, this thesis has implemented a Node.js server in conjunction with the Fabric network to improve the overall speed of the system. However, the use of a separate server (Node.js) to process requests increases the system overhead, which may not be desirable for the computing device due to its limited resource capacity. Therefore, it may be necessary to investigate ways to improve the performance of the network without placing excessive strain on the system's resources in future research.

In this research, parameters of a blockchain, such as block size and block interval time, were adjusted using a learn-to-predict model. However, in theory, this prediction model would continually interact with the network, which would slow down the network and reduce the speed at which transactions can be processed. Future research could explore how to strike a balance between low interaction and high performance, to optimize the performance of the blockchain network.

References

[1] B Bloomberg, J., 2018. Digitization, digitalization, and digital transformation: confuse them at your peril. *Forbes. Retrieved on August*, *28*, p.2019.

[2] Narayan, A., Klaes, M., Babazadeh, D., Lehnhoff, S. and Rehtanz, C., 2019, May. The first approach for a multi-dimensional state classification for ict-reliant energy systems. In *International ETG-Congress 2019; ETG Symposium* (pp. 1-6). VDE.

[3] Bruinenberg, J., Colton, L., Darmois, E., Dorn, J., Doyle, J., Elloumi, O., Englert, H., Forbes, R., Heiles, J., Hermans, P. and Kuhnert, J., 2012. CEN-CENELEC-ETSI smart grid co-ordination group smart grid reference architecture. *CEN, CENELEC, ETSI, Tech. Rep*, *23*, p.24.

[4] VDE, “The German Roadmap E-Energy/Smart Grids 2.0.” DKE – German Commission for Electrical, Electronic and Information Technologies of DIN and VDE, Tech. Rep., 2012.

[5] Edmunds, C., Galloway, S., Elders, I., Bukhsh, W. and Telford, R., 2020. Design of a DSO-TSO balancing market coordination scheme for decentralised energy. *IET Generation, Transmission & Distribution*, *14*(5), pp.707-718.

[6] Saint-Pierre, A. and Mancarella, P., 2016. Active distribution system management: A dual-horizon scheduling framework for DSO/TSO interface under uncertainty. *IEEE Transactions on Smart Grid*, *8*(5), pp.2186-2197.

[7] Suljanović, N., Souvent, A., Taylor, G., Radi, M., Cantenot, J., Lambert, E. and Morais, H., 2019, June. Design of interoperable communication architecture for tso-dso data exchange. In *2019 IEEE Milan PowerTech* (pp. 1-6). IEEE.

[8] ENTSO-E. "Data management in the EU electricity networks ". 2016 [Online]. Available: https://www.entsoe.eu/news-events/announcements/announcements-archive/Pages/News/Data- management-in-the-EU-electricity-networks.aspx

[9] Clark, T. and Jones, R., 1999, June. Organisational interoperability maturity model for C2. In Proceedings of the 1999 command and control research and technology symposium (Vol. 29).

[10] GridWise Architecture Council, “The GridWise® Interoperability Context-Setting Framework is a work of the GridWise Architecture Council,” March 2008.

[11] G Kezunovic, M., Grijalva, S., Dutta, P. and Roy, A., 2012, January. The fundamental concept of unified generalized model and data representation for new applications in the future grid. In 2012 45th Hawaii International Conference on System Sciences (pp. 2096-2103). IEEE.

[12] Alves, R., Reis, F. and Liang, C., 2015. TSOs and DSOs collaboration: The need for data exchange. vol. Trivent En, no. Deregulated Electricity Market Issues in South Eastern Europe, pp.55-62.

[13] European Commission. "EU energy in figures—Statistical pocketbook 2017."

(2017). [Online] Available: https://ec.europa.eu/energy/sites/ener/files/documents/pocketbook_energy_2017_web.pdf

[14] Howard, M.W., 2014. VII. The Integrated Grid: Realizing the Full Value of Central and Distributed Energy Resources. *The ICER Chronicle Edition 2, July 2014*, *1*, p.32.

[15] Burtin, A. and Silva, V., 2015. Technical and economic analysis of the European electricity system with 60% RES.

[16] Gerard, H., Puente, E.I.R. and Six, D., 2018. Coordination between transmission and distribution system operators in the electricity sector: A conceptual framework. *Utilities Policy*, *50*, pp.40-48.

[17] European Commission. "Regulation of the European Parliament and of the Council on the internal market for electricity. " 2015. [Online]. Available: https://www.entsoe.eu/Documents/Publications/Position%20papers%20and%20reports/1503 03_ENTSO-E_Position_Paper_TSO-DSO_interaction.pdf.: http://eurlex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32015R0848&from=EN

[18] Khajeh, H., Laaksonen, H., Gazafroudi, A.S. and Shafie-khah, M., 2019. Towards flexibility trading at TSO-DSO-customer levels: A review. Energies, 13(1), p.165.

[19] Grenard, S. "TSO-DSO interaction in power systems with high RES penetration ". 2015 [Online]. Available: https://www.ofgem.gov.uk/ofgem-publications/107804

[20] CIGRE/CIRED. "Planning Criteria for Future Transmission Networks in the Presence of a Greater Variability of Power Exchange with Distribution Systems “. CIGRE/CIRED JWG C1.29. 2017

[21] Zegers, A. and Brunner, H., 2014. TSO-DSO interaction: An Overview of current interaction between transmission and distribution system operators and an assessment of their cooperation in Smart Grids. International Smart Grid Action Network (ISAGN) Discussion Paper Annex, 6, pp.2-32.

[22] European Commission. "System Operation Guideline ". 2016 [Online]. Draft version Available:https://ec.europa.eu/energy/sites/ener/files/documents/SystemOperationGuideline%20final%28pr ovisional%2904052016.pdf

[23] ENTSO-E. " All TSOs' proposal for the key organizational requirements, roles and responsibilities in relation to data exchange ". 2017 [Online]. Available: https://consultations.entsoe.eu/system-operations/korrr/consult_view/.

[24] EURELECTRIC-- Networks Committee. "The role of distribution system operators (DSOs) as information hubs." 2010. [Online]. Available: http://www.eurelectric.org/media/44143/role_of_dsos_as_information_hubs_final_draft_10-06-10- 2010-200-0001-01-e.pdf.

[25] ENTSO-E. " All TSOs' proposal for the key organizational requirements, roles and responsibilities in relation to data exchange ". 2017 [Online]. Available: https://consultations.entsoe.eu/system-operations/korrr/consult_view/.

[26] Rivero, E., D. Six, and H. Gerard. "Assessment of future market architectures and regulatory frameworks for network integration of DRES–the future roles of DSOs." evolvDSO project, Deliverable 1 (2015). [Online]. Available:http://www.evolvdso.eu.

[27] CEER. "Position Paper on the Future DSO and TSO Relationship "(2016) [Online]. Available: http://www.aemc.gov.au/getattachment/de49d815-7a03-46b4 893007511353f5fa/SandC-Electric-Company.aspx

[28] CEDEC "Position paper on A European Electricity Market Design Fit for the Energy Transition (2015)." [Online]. Available: http://www.cedec.com/files/default/a-european electricity-market-design-fitfor-the-energy-transition-cedec-position-paper.pdf.

[29] European Commission. "Consultation on a new Energy Market Design." 2015 [Online]. Available:https://ec.europa.eu/energy/en/consultations/public-consultation-new-energy- market-design.

[30] ENTSO-E. "Towards smarter grids: Developing TSO and DSO roles and interactions for the benefit of consumers." 2015 [Online]. Available: https://www.entsoe.eu/Documents/Publications/Position%20papers%20and%20reports/15 030 3_ENTSO-E Position Paper TSO-DSO interaction.pdf

[31] Eurelectric. "Response to EC consultation on a new Energy Market Design."2015 [Online]. Available: https://ec.europa.eu/energy.

[32] A. Guerrero, J. I. García Delgado, A. Personal Vázquez, E. Luque Rodríguez, J. León de Mora, "Heterogenous data source integration for smart grid ecosystem", 2017.

[33] Goel, Sanjay, Stephen F. Bush, and David Bakken. "IEEE vision for smart grid communications: 2030 and beyond." IEEE Standard Association (2013): 1-390.

[34] ENTSO-E, "Response to EC consultation on a new Energy Market Design." 2015 [Online]. Available: https://www.entsoe.eu/Documents/News/151012_Response%20to%20EC%20Consultation%20o n%20Market%20Design.pdf

[35] ENTSO-E, "Network Code Overview,"2015 [Online].Available : https://www.entsoe.eu/majorprojects/network-code development/Pages/default.aspx

[36] European Commission. "Commission Regulation (EU) 2016/631 of 14 April 2016". 2016 [Online]. Available: https://publications.europa.eu/en/publication-detail//publication/1267e3d1-0c3f-11e6-ba9a-01aa75ed71a1/language-en

[37] European Commission. "Commission Regulation (EU) 2017/1485 of 2 August 2017". 2017 [Online]. Available: http://eur-

lex.europa.eu/legalcontent/EN/TXT/?uri=CELEX:32017R1485.

[38] European Commission. "Commission Regulation (EU) 2017/2196 of 24 November 2017". 2017 [Online]. Available: http://eur-lex.europa.eu/legalcontent/EN/TXT/?uri=CELEX:32017R1485.

[39] European Commission, “Draft Regulation establishing a guideline on electricity transmission system operation – the System Operation Guideline.” 2016 [Online] Available: https://ec.europa.eu/energy/sites/ener/files/documents/SystemOperationGuideline%20final%28 provisional%2904052016.pdf

[40] European Commission.” Commission Regulation (EU) 2015/1222 of 24 July 2015”. [Online]. Available: http://eurlex.europa.eu/legalcontent/EN/TXT/?uri=uriserv%3AOJ.L_.2016.259.01.0042.01.E NG

[41] European Commission.” Commission Regulation (EU) 2017/2195 of 23 November 2017”. 2017 [Online]. Available: http://eurlex.europa.eu/legalcontent/EN/TXT/?uri=uriserv%3AOJ.L_.2016.259.01.0042.01.E NG

[42] Goel, Sanjay, Stephen F. Bush, and David Bakken. "IEEE vision for smart grid communications: 2030 and beyond." IEEE Standard Association (2013): 1-390.

[43] Bretzke, S., Ponce de Leon, M., Musa, A., Toma, L., Murphy, C., Gurumurthy, S. "RESERVE Deliverable D1.3: ICT Requirements". 2017 [Online].Available:http://www.reserve.eu/files/reserve/Content/Deliverables/D1.3.pdf

[44] Uslar, M., Specht, M., Rohjans, S., Trefke, J., and Gonzalez,J.M. “The Common Information Model CIM IEC 61968/61970 and 62325 - A Practical Introduction to the CIM “. Springer-Verlag Berlin Heidelberg.2017.

[45] Everden, R. “TOGAF vs ArchiMate – What Are the Differences? “. 2016 [Online]. Available:http://blog.goodelearning.com/togaf/togaf-vs-archimate-what-are-the-differences/

[46] Papavasiliou, Anthony, and Ilyes Mezghani. "Coordination schemes for the integration of transmission and distribution system operations." 2018 Power Systems Computation Conference (PSCC). IEEE, 2018.

[47] ENTSO-E. “MADES Communication Standard“.2014 [Online]. Available: https://www.entsoe.eu/Documents/EDI/Library/depreciated/503_mades-v1r1.pdf.

[48] TDX-ASSIST Project. “Deliverable D1.1, State of the Art - TSO-DSO Interoperability,” Horizon 2020, the EU Framework Programme for Research & Innovation, 2017. [Online]. Available: http://www.tdx-assist.eu/index.php

[49] Herhalt, J., and K. Cochrane. "Exploring the cloud: A global study of governments’ adoption of cloud." Klynveld Peat Marwick Goerdeler (KPMG) (2012): 1-46.

[50] Garrison, Gary, Sanghyun Kim, and Robin L. Wakefield. "Success factors for deploying cloud computing." Communications of the ACM 55.9 (2012): 62-68.

[51] Yang, Haibo, and Mary Tate. "A descriptive literature review and classification of cloud computing research." Communications of the Association for Information Systems 31.1 (2012): 2.

[52] Doug Hyde. “A survey on the security of virtual machines”. 2009. [Online]. Available: https://www.cse.wustl.edu/~jain/cse571-09/ftp/vmsec/.

[53] Perez-Botero, D., Szefer, J. and Lee, R.B., 2013, May. Characterizing hypervisor vulnerabilities in cloud computing servers. In *Proceedings of the 2013 international workshop on Security in cloud computing* (pp. 3-10).

[54] Yang, Y., Jiang, H., Zhang, G., Wang, X., Lv, Y., Li, X., Fdida, S. and Xie, G., 2021. S2H: Hypervisor as a setter within Virtualized Network I/O for VM isolation on cloud platform. *Computer Networks*, *201*, p.108577.

[55] Riddle, A.R. and Chung, S.M., 2015, June. A survey on the security of hypervisors in cloud computing. In *2015 IEEE 35th International Conference on Distributed Computing Systems Workshops* (pp. 100-104). IEEE.

[56] Aalam, Z., Kumar, V. and Gour, S., 2021, August. A review paper on hypervisor and virtual machine security. In *Journal of Physics: Conference Series* (Vol. 1950, No. 1, p. 012027). IOP Publishing.

[57] Kazim, M., Masood, R., Shibli, M.A. and Abbasi, A.G., 2013, September. Security aspects of virtualization in cloud computing. In *IFIP International Conference on Computer Information Systems and Industrial Management* (pp. 229-240). Springer, Berlin, Heidelberg.

[58] Taubmann, B. and Reiser, H.P., 2020, June. Towards hypervisor support for enhancing the performance of virtual machine introspection. In *IFIP International Conference on Distributed Applications and Interoperable Systems* (pp. 41-54). Springer, Cham.

[59] Coppolino, L., D’Antonio, S., Mazzeo, G. and Romano, L., 2017. Cloud security: Emerging threats and current solutions. *Computers & Electrical Engineering*, *59*, pp.126-140.

[60] Chien-Yu Liu, Meng-Ru Shie, Yi-Fang Lee, Yu-Chun Lin, and Kuan-Chou Lai. “Vertical/Horizontal Resource Scaling Mechanism for Federated Clouds,” in 2014 International Conference on Information Science & Applications (ICISA), Seoul, South Korea, May 2014.

[61] Nandgaonkar, S.V. and Raut, A.B., 2014. A comprehensive study on cloud computing. *International Journal of Computer Science and Mobile Computing, a Monthly Journal of Computer Science and Information Technology*, *3*, pp.733-738.

[62] Momeni, H. and Mabhoot, N., 2021. An Energy-aware Real-time Task Scheduling Approach in a Cloud Computing Environment. *Journal of AI and Data Mining*, *9*(2), pp.213-226.

[63] Mazumdar, S., Seybold, D., Kritikos, K. and Verginadis, Y., 2019. A survey on data storage and placement methodologies for cloud-big data ecosystem. *Journal of Big Data*, *6*(1), pp.1-37.

[64] Venkatesh, A. and Eastaff, M.S., 2018. A study of data storage security issues in cloud computing. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *3*(1), pp.1741-1745.

[65] Sadeeq, M.M., Abdulkareem, N.M., Zeebaree, S.R., Ahmed, D.M., Sami, A.S. and Zebari, R.R., 2021. IoT and Cloud computing issues, challenges and opportunities: A review. *Qubahan Academic Journal*, *1*(2), pp.1-7.

[66] Cai, C. and Chen, C., 2021. Optimization of human resource file information decision support system based on cloud computing. *Complexity*, *2021*.

[67] Attaran, M. and Woods, J., 2019. Cloud computing technology: improving small business performance using the Internet. Journal of Small Business & Entrepreneurship, 31(6), pp.495-519.

[68] Iansiti, M., & Lakhani, K. R. (2017). The truth about blockchain. Harvard Business Review, 95(1), 118-127.

[69] Yang, R., Wakefield, R., Lyu, S., Jayasuriya, S., Han, F., Yi, X., Yang, X., Amarasinghe, G. and Chen, S., 2020. Public and private blockchain in construction business process and information integration. Automation in construction, 118, p.103276.

[70] Wang, W., Hoang, D.T., Hu, P., Xiong, Z., Niyato, D., Wang, P., Wen, Y. and Kim, D.I., 2019. A survey on consensus mechanisms and mining strategy management in blockchain networks. Ieee Access, 7, pp.22328-22370.

[71] Shalaby, S., Abdellatif, A.A., Al-Ali, A., Mohamed, A., Erbad, A. and Guizani, M., 2020, February. Performance evaluation of hyperledger fabric. In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT) (pp. 608-613). IEEE.

[72] Vranken, H., 2017. Sustainability of bitcoin and blockchains. Current opinion in environmental sustainability, 28, pp.1-9.

[73] Dannen, C., 2017. Introducing Ethereum and solidity (Vol. 1, pp. 159-160). Berkeley: Apress.

[74] Brown, R.G., Carlyle, J., Grigg, I. and Hearn, M., 2016. Corda: an introduction. R3 CEV, August, 1(15), p.14.

[75] Baliga, A., Subhod, I., Kamat, P. and Chatterjee, S., 2018. Performance evaluation of the quorum blockchain platform. arXiv preprint arXiv:1809.03421.

[76] Iftekhar, A., Cui, X., Tao, Q. and Zheng, C., 2021. Hyperledger fabric access control system for internet of things layer in blockchain-based applications. Entropy, 23(8), p.1054.

[77] Beckert, B., Herda, M., Kirsten, M. and Schiffl, J., 2018. Formal specification

and verification of Hyperledger Fabric chaincode. In 3rd Symposium on Distributed Ledger Technology (SDLT-2018) co-located with ICFEM (pp. 44-48).

[78] Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R. and Wang, F.Y., 2018, June. An overview of smart contract: architecture, applications, and future trends. In 2018 IEEE Intelligent Vehicles Symposium (IV) (pp. 108-113). IEEE.

[79] Chen, X., Li, X., Zhang, Q., Shi, Z. and Guan, Y., 2020, March. Formalizing the Transaction Flow Process of Hyperledger Fabric. In International Conference on Formal Engineering Methods (pp. 233-250). Springer, Cham.

[80] Foschini, L., Gavagna, A., Martuscelli, G. and Montanari, R., 2020, June. HyperLedger fabric blockchain: chaincode performance analysis. In ICC 2020-2020 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

[81] Androulaki, E., De Caro, A., Neugschwandtner, M. and Sorniotti, A., 2019, July. Endorsement in hyperledger fabric. In 2019 IEEE International Conference on Blockchain (Blockchain) (pp. 510-519). IEEE.

[82] Baliga, A., Solanki, N., Verekar, S., Pednekar, A., Kamat, P. and Chatterjee, S., 2018, June. Performance characterization of hyperledger fabric. In 2018 Crypto Valley conference on blockchain technology (CVCBT) (pp. 65-74). IEEE.

[83] Hewa, T., Ylianttila, M. and Liyanage, M., 2021. Survey on blockchain based smart contracts: Applications, opportunities and challenges. Journal of Network and Computer Applications, 177, p.102857.

[84] Nguyen, G.T. and Kim, K., 2018. A survey about consensus algorithms used in blockchain. Journal of Information processing systems, 14(1), pp.101-128.

[85] Nakamoto, S., 2008. Re: Bitcoin P2P e-cash paper. The Cryptography Mailing List, pp.1-2.

[86] Fullmer, D. and Morse, A.S., 2018, December. Analysis of difficulty control in bitcoin and proof-of-work blockchains. In 2018 IEEE Conference on Decision and Control (CDC) (pp. 5988-5992). IEEE.

[87] Zhao, W., Yang, S., Luo, X. and Zhou, J., 2021, March. On PeerCoin Proof of Stake for Blockchain Consensus. In 2021 The 3rd International Conference on Blockchain Technology (pp. 129-134).

[88] Sukhwani, H., Martínez, J.M., Chang, X., Trivedi, K.S. and Rindos, A., 2017, September. Performance modeling of PBFT consensus process for permissioned blockchain network (hyperledger fabric). In 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS) (pp. 253-255). IEEE.

[89] Lei, K., Zhang, Q., Xu, L. and Qi, Z., 2018, December. Reputation-based byzantine fault-tolerance for consortium blockchain. In 2018 IEEE 24th international conference on parallel and distributed systems (ICPADS) (pp. 604-611). IEEE.

[90] Di Silvestre, M.L., Gallo, P., Guerrero, J.M., Musca, R., Sanseverino, E.R.,

Sciumè, G., Vásquez, J.C. and Zizzo, G., 2020. Blockchain for power systems: Current trends and future applications. Renewable and Sustainable Energy Reviews, 119, p.109585.

[91] Livingston, D., Sivaram, V., Freeman, M. and Fiege, M., 2018. Applying blockchain technology to electric power systems.

[92] Amjad, M., Taylor, G., Lai, C.S., Huang, Z. and Li, M., 2022, August. A Novel Blockchain Based Approach to Exchanging Information and Data in Power Systems. In 2022 57th International Universities Power Engineering Conference (UPEC) (pp. 1-6). IEEE.

[93] Radi, M., Taylor, G., Cantenot, J., Lambert, E. and Suljanovic, N., 2021. Developing Enhanced TSO-DSO Information and Data Exchange Based on a Novel Use Case Methodology. Frontiers in Energy Research, 9, p.259.

[94] Radi, M., Taylor, G., Uslar, M., Köhlke, J. and Suljanovic, N., 2019, September. Bidirectional power and data flow via enhanced portal based TSO-DSO Coordination. In 2019 54th International Universities Power Engineering Conference (UPEC) (pp. 1-5). IEEE.

[95] Song, Yaqi, Shunren LIU, and Yongli ZHU. "Cloud storage of power equipment state data sampled with high speed." Electric Power Automation Equipment 33.10 (2013): 150-156.

[96] Amjad, M., Taylor, G., Li, M. and Huang, Z., 2021, December. A Critical Evaluation of Cloud Computing Techniques for TSO and DSO Information and Data Exchange. In 2021 11th International Conference on Power and Energy Systems (ICPES) (pp. 481-485). IEEE.

[97] Cloudera.“ClouderaWhitepaper”.2019.[Online].Available:https://www.cloudera.com/about/enterprise-data-cloud.html

[98] Cloudera. “Cloudera Documentation”. 2019. [Online]. Available:https://docs.cloudera.com/documentation/enterprise/6/6.3/topics/security.html

[99] Hong, A., Xiao, W. and Ge, J., 2021, May. Big Data Analysis System Based on Cloudera Distribution Hadoop. In 2021 7th IEEE Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS) (pp. 169-173). IEEE.

[100] Devi, A., 2017. Design of Mechanism for Enhancement the Security of Hadoop Processing Tool" Hive" With VMWARE Platform. International Journal of Advanced Research in Computer Science, 8(5).

[101] Srivastava, K. and Kumar, A., 2011. A new approach of cloud: Computing infrastructure on demand. Trends in Information Management, 7(2).

[102] Bartol, J., Kodek, T., Souvent, A., Oliveira, F., Lambert, E., Petrovič, N. and Suljanović, N., 2019, November. Utilization of ECCo SP for secured and reliable

information exchange between system operators. In 2019 27th Telecommunications Forum (TELFOR) (pp. 1-4). IEEE.

[103] TDX-ASSIST Project. “Deliverable 3.2, Definition of use cases with regard to levels of portal access, of WP3,” Horizon 2020, the EU Framework Programme for Research & Innovation,2017.

[104] Amjad, M., Taylor, G., Lai, C.S., Huang, Z. and Li, M., 2022, August. A Novel Blockchain Based Approach to Exchanging Information and Data in Power Systems. In 2022 57th International Universities Power Engineering Conference (UPEC) (pp. 1-6). IEEE.

[105] Aleksieva, V., Valchanov, H. and Huliyan, A., 2020, June. Implementation of smart-contract, based on hyperledger fabric blockchain. In 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA) (pp. 1-4). IEEE.

[106] Wang, C. and Chu, X., 2020, July. Performance characterization and bottleneck analysis of hyperledger fabric. In 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS) (pp. 1281-1286). IEEE.

[107] “Hyperledger Caliper.” Available online: https://hyperledger.github.io/caliper/v0.3.2/fabric-config/. [Accessed: 12-June2021].

[108] Eyal, Ittay, et al. "Bitcoin-ng: A scalable blockchain protocol." 13th {USENIX} symposium on networked systems design and implementation ({NSDI} 16). 2016.

[109] Kim, Soohyeong, Yongseok Kwon, and Sunghyun Cho. “A survey of scalability solutions on blockchain.” 2018 International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2018.

[110] Swan, M., 2015. Blockchain: Blueprint for a new economy. " O'Reilly Media, Inc.".

[111] Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K., 2016. Where is current research on blockchain technology?—a systematic review. PloS one, 11(10), p.e0163477.

[112] W. Xin et al., “On Scaling and Accelerating Decentralized Private Blockchains,” Proc. 2017 IEEE 3rd Int’l. Conf. Big Data Security on Cloud, IEEE Int’l. Conf. High Performance and Smart Computing, and IEEE Int’l. Conf. intelligent Data and Security, 2017, pp. 267–71.

[113] M. Liu et al., “Performance Optimization for Blockchain-Enabled Industrial Internet of Things (IIoT) Systems: A Deep Reinforcement Learning Approach,” IEEE Trans. Industrial Informatics, vol. 15, no. 6, 2019, pp. 3559–70.

[114] Zhou, Q., Huang, H., Zheng, Z. and Bian, J., 2020. Solutions to scalability of blockchain: A survey. Ieee Access, 8, pp.16440-16455.

[115] Mazlan, A.A., Daud, S.M., Sam, S.M., Abas, H., Rasid, S.Z.A. and Yusof, M.F., 2020. Scalability challenges in healthcare blockchain system—a systematic

review. IEEE Access, 8, pp.23663-23673.

[116] Sanka, A.I.; Cheung, R.C. A systematic review of blockchain scalability: Issues, solutions, analysis and future research. J. Netw. Comput. Appl. 2021, 195, 103232.

[117] Xie, J.; Yu, F.R.; Huang, T.; Xie, R.; Liu, J.; Liu, Y. A Survey on the Scalability of Blockchain Systems. IEEE Netw. 2019, 33, 166–173.

[118] Alrehaili, A.; Namoun, A. A Comparative Analysis of Scalability Issues within Blockchain-based Solutions in the Internet of Things. Environments 2021, 12, 480–490.

[119] Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Mohamed, N.A. and Arshad, H., 2018. State-of-the-art in artificial neural network applications: A survey. Heliyon, 4(11), p.e00938.

[120] Nabavi, S.A., Aslani, A., Zaidan, M.A., Zandi, M., Mohammadi, S. and Hossein Motlagh, N., 2020. Machine learning modeling for energy consumption of residential and commercial sectors. Energies, 13(19), p.5171.

Appendix A

Software Environment Description:

1. Docker Engine (Version 20.10.17):

Overview: Docker Engine is a containerization platform that allows you to package and run applications and their dependencies in isolated containers. It provides a consistent and reproducible environment for your experiments.

Significance: Docker containers are lightweight, efficient, and can be easily deployed across different environments. They ensure that your blockchain setup and dependencies are consistent and isolated from the host system.

2. Docker Compose (Version 1.29.2):

Overview: Docker Compose is a tool for defining and running multi-container Docker applications. It allows you to define the services, networks, and volumes required for your application in a single Compose file.

Significance: Docker Compose simplifies the management of complex multi-container applications, making it easier to set up and orchestrate your Hyperledger Fabric blockchain network and related components.

3. CPU: Intel Core i7-3.00GHz:

Overview: The Intel Core i7 processor is a high-performance CPU known for its processing power and efficiency. It plays a crucial role in the overall performance of your experiments.

Significance: The powerful Intel Core i7 CPU ensures that your blockchain network can handle computationally intensive tasks efficiently, contributing to the overall performance of your experiments.

4. Memory: 16 GB:

Overview: The 16 GB of RAM (Random Access Memory) provides the server with a sufficient amount of memory for running applications and handling data efficiently.

Significance: Ample memory is essential for managing the blockchain network, databases, and any other software components simultaneously, ensuring smooth and responsive performance.

5. Operating System: Ubuntu 22.02:

Overview: Ubuntu 22.02 LTS is a popular Linux distribution known for its stability and long-term support. It provides a reliable environment for running various software components.

Significance: Ubuntu offers a secure and well-supported platform for hosting your Hyperledger Fabric blockchain network, Node.js applications, and MongoDB database.

6. Node SDK (Node.js):

Overview: Node.js is an open-source JavaScript runtime environment that allows you to execute JavaScript code outside of a web browser. It's commonly used for building server-side applications.

Significance: Node.js, along with the Node SDK, is crucial for interacting with and developing applications for the Hyperledger Fabric blockchain network. It enables the creation of blockchain smart contracts and client applications.

7. Blockchain Platform: Hyperledger Fabric:

Overview: Hyperledger Fabric is a blockchain framework designed for building permissioned blockchain networks. It provides a modular and extensible architecture for developing enterprise-grade blockchain applications.

Significance: Hyperledger Fabric serves as the foundation of your blockchain network, providing the necessary infrastructure and tools for creating, managing, and executing blockchain transactions.

8. Programming Language: JavaScript:

Overview: JavaScript is a widely used programming language known for its versatility. In your context, it's used for developing blockchain applications and interacting with the Hyperledger Fabric network.

Significance: JavaScript allows you to write smart contracts, applications, and scripts to interact with the blockchain. It's commonly used in web-based blockchain applications.

9. Database Management System (DBMS): MongoDB:

Overview: MongoDB is a NoSQL database known for its flexibility and scalability. It's suitable for storing unstructured or semi-structured data, making it relevant for blockchain use cases.

Significance: MongoDB likely serves as the database for storing off-chain data related to your blockchain network. It complements the blockchain ledger with versatile data storage capabilities.

Server Specifications:

Processor (CPU):

Manufacturer: Intel

Number of Cores: 14

Clock Speed: 1.7GHz

Cache Size: 35.75 MB

Memory (RAM):

Total Installed Memory: 128 GB

Memory Type: DDR4

Memory Speed: 2400 MHz

Number of Modules: 8

Total Memory Capacity: 256 GB (Expandable)

Storage Devices:

Hard Disk Drives (HDDs):

Number of HDDs: 2

Capacity of Each HDD: 2 TB

RPM (if applicable): 7200 RPM

Solid State Drives (SSDs):

Number of SSDs: 1

Capacity of Each SSD: 512 GB

Graphics Processing Unit (GPU):

Manufacturer: NVIDIA

Model: GeForce RTX 2080 Ti

VRAM: 11 GB GDDR6

Type (e.g., Ethernet, Wi-Fi): Ethernet

Data Transfer Speed (for each NIC): 1 Gbps

Operating System:

Name: Ubuntu Server

Version: 22.02

Architecture (32-bit/64-bit): 64-bit