Efficient Design of Decentralized Privacy and Trust in Distributed Digital Marketplaces

Roman-Valentyn Tkachuk

Department of Computer Science
Blekinge Institute of Technology

Doctoral Dissertation Series no. 2023:13

Efficient Design of Decentralized Privacy and Trust in Distributed Digital Marketplaces

Roman-Valentyn Tkachuk

Blekinge Institute of Technology Doctoral Dissertation Series
No: 2023:13

Efficient Design of Decentralized Privacy and Trust in Distributed Digital Marketplaces

Roman-Valentyn Tkachuk

Doctoral Dissertation in
Computer Science

Department of Computer Science
Blekinge Institute of Technology
SWEDEN

Department of Computer Science
Publisher: Blekinge Institute of Technology,
SE-371 79 Karlskrona, Sweden
Printed by MEDIA-TRYCK, Lund, Sweden 2023
ISBN 978-91-7295-465-6
ISSN 1653-2090

"I'm a great believer in luck, and I find the harder I work, the more I have of it."

Thomas Jefferson

Abstract

This thesis aims to advance the knowledge on the efficient design and evaluation of distributed marketplaces with an emphasis on trust and privacy. Distributed systems are an integral part of today's computing infrastructures, enabling multiple nodes to work towards a common goal. Although distributed, most of today's computational infrastructures are still built as centralized systems, which assume governance by a single organization. In the case of centralized marketplaces, the correct trade execution guarantees, *i. e.*, digital trust, and data privacy are provided centrally, containing all processes and operations within a single organization's boundaries. This puts the marketplace operator in a prime position to govern trade settlement conditions. Further, centralization may limit the rapid expansion of the marketplace in its respective domain due to restricted process interoperability and automation. Thus, a decentralized marketplace model can be adopted to enable governance distribution among multiple organizations, enhancing marketplace scalability and bringing automation and interoperability to the services of collaborating governor organizations. However, trust issues are raised if more than one organization has to govern the marketplace. In such a case, trust and privacy are decentralized, and control is distributed among all organizations that are part of the marketplace system. Thus, a decentralized marketplace requires a robust and secure digital trust-enabling mechanism while allowing organizations to process and store private data for further usage in trade settlements.

This thesis investigates distributed marketplaces where centralized and decentralized governance models are applied to use cases of Artificial Intelligence (AI) artifacts and renewable energy trading. It begins with a study of a marketplace for AI artifacts where multiple organizations collaborate on AI pipeline execution. The study defines a Secure Virtual Premise, which enables AI pipeline execution in a centralized marketplace governed by a trusted third party. The thesis continues with a survey of the telecommunication services marketplaces, where both centralized and decentralized governance models are discussed. In addition, the survey provides an in-depth investigation of blockchain technology as a main trust-enabling platform, providing distributed storage and data assurance to all processes in a decentralized marketplace. Having mapped the state-of-the-art, the research shifts towards an in-depth investigation of blockchain-based decentralized renewable energy marketplaces. The designed marketplace enables automation and digital trust of peer-to-peer (P2P) energy trade settlements in decentralized systems while preserving users' data privacy. Furthermore, the marketplace is aligned with the data and P2P energy trade regulations. The studies provide an in-depth requirements definition, system architecture, implementation, and performance evaluation of marketplaces based on two major blockchain platforms. The final study of this thesis provides improvements towards the renewable energy marketplace model aiming at an enhancement of digital trust, privacy, and scalability. Ultimately, such a marketplace should incentivize the widespread adoption of renewable energy sources, resulting in the decarbonization of electricity distribution systems.

Preface

This thesis includes five publications. The author of this thesis is the first author and main driver of all included publications. All studies described in publications have been defined under the guidance of the author's supervisors. Both supervisors contributed with suggestions and comments. Included publications contain the originally published contents with the formatting changed to match the styling of this document.

Paper I Roman-Valentyn Tkachuk, Dragos Ilie, Kurt Tutschku, "Towards a Secure Proxy-based Architecture for Collaborative AI Engineering," In *Proceedings of The Eighth International Symposium on Computing and Networking Workshops*, IEEE, Okinawa, Japan, pp. 1-7, Nov. 2020, DOI: 10.1109/CANDARW51189.2020.00077

Paper II Roman-Valentyn Tkachuk, Dragos Ilie, Kurt Tutschku, Remi Robert, "A Survey on Blockchain-based Telecommunication Services Marketplaces," *IEEE Transactions on Network and Service Management*, *vol. 19*, *no. 1*, IEEE, pp. 1-28, Oct. 2021, DOI: 10.1109/TNSM.2021.3123680

Paper III Roman-Valentyn Tkachuk, Dragos Ilie, Remi Robert, Victor Kebande, Kurt Tutschku "Towards Efficient Privacy and Trust in Decentralized Blockchain-Based Peer-to-Peer Renewable Energy Marketplace," *Sustainable Energy, Grids and Networks*, *vol. 35*, Elsevier, pp. 1-27, Aug. 2023, DOI: 10.1016/j.segan.2023.101146

Paper IV Roman-Valentyn Tkachuk, Dragos Ilie, Remi Robert, Victor Kebande, Kurt Tutschku "On the Performance and Scalability of Consensus Mechanisms in Privacy-Enabled Decentralized Renewable Energy Marketplace," *Annals of Telecommunications*, Springer, pp. 1-18, Sep. 2023, (*Invited paper, the extension of Paper VIII*), DOI: 10.1007/s12243-023-00973-8

Paper V Roman-Valentyn Tkachuk, Dragos Ilie, Remi Robert, Kurt Tutschku, Victor Kebande "On the Application of Enterprise Blockchains in Decentralized Renewable Energy Marketplaces," *To be submitted for journal publication.*

Other contributions that are related to this thesis but not included:

Paper VI Roman-Valentyn Tkachuk, Dragos Ilie, Kurt Tutschku, "Orchestrating Future Service Chains in the Next Generation of Clouds". In *Proceedings of 15th Swedish National Computer Networking Workshop*, Luleå, Sweden, pp. 1-5, Jun. 2019, URN: URN:NBN:SE:BTH-18785

Paper VII Roman-Valentyn Tkachuk, Dragos Ilie, Kurt Tutschku, "Building a Framework for Automated Security Testbeds in Cloud Infrastructures". In *Proceedings of 16th Swedish National Computer Networking Workshop*, Kristianstad, Sweden, pp. 1-4, May 2020, URN: URN:NBN:SE:BTH-19621

Paper VIII Roman-Valentyn Tkachuk, Dragos Ilie, Remi Robert, Victor Kebande, Kurt Tutschku, "On the Performance of Consensus Mechanisms in Privacy-Enabled Decentralized Peer-to-Peer Renewable Energy Marketplace". In *Conference on Innovation in*

Clouds, Internet and Networks, IEEE, Paris, France, pp. 1-8, Mar. 2023, (*Best Paper Award*), DOI: 10.1109/ICIN56760.2023.10073510

Tutorial I Roman-Valentyn Tkachuk, Dragos Ilie, Kurt Tutschku, "Decentralized Blockchain-based Telecommunication Services Marketplaces". Presented in *IEEE International Conference on Network Softwarization*, Tokyo, Japan, Jul. 2021

Report I Kurt Tutschku, Roman-Valentyn Tkachuk, "Revised Bonseyes system architecture and concepts – deliverable D1.3". In *H2020 project Bonseyes – AI Marketplace*, technical report, Nov. 2019.

Report II Kurt Tutschku, Roman-Valentyn Tkachuk, "Secure Virtual Premise architecture and workflow – deliverable D2.4". In *H2020 project Bonseyes – AI Marketplace*, technical report, Nov. 2019.

Funding

This thesis was partially funded by:

- Swedish Knowledge Foundation (KKS) under the project "*Symphony - Supply-and-Demand-based Service Exposure using Robust Distributed Concepts*" under grant agreement number 20190111.
- European Unions (EU) Horizon 2020 research and innovation program under grant agreement number 732204 (Bonseyes).
- Swiss State Secretariat for Education Research and Innovation (SERI) under contract number 16.0159.
- Swedish Agency for Economic and Regional Growth (Tillväxtverket) under the umbrella of European Structural and Investment Funds (ESI) covered by grant agreement number 20201213.

Disclaimer

Some contents of this doctoral dissertation were previously published in the author's licentiate dissertation [1]. The licentiate is a halfway point between master's and doctor's degrees.

Acknowledgements

I would like to thank the people who supported me throughout my research path and made this thesis possible. First and foremost, I would like to express my deepest gratitude to my scientific supervisors *Prof. Kurt Tutschku* and *Dr. Dragos Ilie*. Thank you for all the valuable feedback that you have given. Your mentorship has helped me to improve my competence and to grow both professionally and personally. You are outstanding researchers and wonderful human beings. It is a great honor to work with you.

Thanks to all my friends and colleagues who have helped with comments and discussions throughout all these years. Thank you for creating a positive and supportive work environment.

I would also like to express my gratitude to *Remi Robert* and *Johan Sjöberg* from Ericsson Research Stockholm for all the discussions, feedback, and support. Further, I would like to thank *Robbert Prinselaar* from Affärsverken Karlskrona for his cooperation and support.

My deepest gratitude to my family, who has always been there for me. Mom, Dad, Grandma†, you are the people who shaped my personality, and without your dedication and diligence, this thesis would not have been possible. I also would like to say thanks to my elder brother *Taras*, who has always been an excellent example to follow. Finally, I would like to thank my wife *Liubov* and daughters *Anna* and *Melania*. *Liubov*, your support is the reason I have embarked on this adventure, and I thank you for your patience and understanding. *Anna* and *Melania*, thank you for being the source of my inspiration and joy every single day. You will always be my greatest achievement.

October 2023, Karlskrona, Sweden
Roman-Valentyn Tkachuk

Contents

Acronyms

AC	Access Control
AM	Application Marketplace
BFT	Byzantine Fault Tolerant
BO	Blockchain Organization
BPS	Block Period Seconds
CA	Certificate Authority
CBAN	Communication Business Automation Network
CDR	Call Data Record
CFT	Crash Fault Tolerant
CLI	Command-Line Interface
CSP	Communication Service Provider
CSM	Cloud Services Marketplace
CSR	Certificate Sign Request
D2018/2001	Directive 2018/2001 of European Parliament
DAG	Direct Acyclic Graph
DApp	Decentralized Application
DER	Distributed Energy Resource
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
DoS	Denial of Service
DPoS	Delegated Proof of Stake
DSL	Domain Specific Language
EDS	Energy Distribution System
eID	Electronic Identification
EIP	Ethereum Improvement Proposal
EP	Energy Provider
EPCF	Endorsement Policy Configuration File
EU	European Union
FR	Functional Requirements
FT	Fungible Token
GDPR	General Data Protection Regulation
GO	Guarantee of Origin
HB	Hyperledger Besu
HC	Hyperledger Calipher
HF	Hyperledger Fabric
IAM	Identity and Access Management

IBFT	Istanbul BFT
IDS	Intrusion Detection System
IdM	Identity Management
IOPS	Input Operations Per Second
IoT	Internet of Things
IPS	Intrusion Prevention System
IPFS	InterPlanetary File System
kWh	Kilowatt-hours
MC	Marketplace Channel
MI	Marketplace Interface
MNO	Mobile Network Operator
MSP	Membership Service Provider
NFT	Non-Fungible Token
NR	Non-Functional Requirements
NRA	National Regulatory Authority
P2P	Peer-to-peer
PDC	Private Data Collection
PG	Privacy Group
PoA	Proof of Authority
PoW	Proof of Work
PoS	Proof of Stake
PBFT	Practical Byzantine Fault Tolerance
QBFT	Quorum BFT
QoS	Quality of Service
RES	Renewable Energy Source
RPS	Reads per Second
SC	Smart Contract
SCP	Small Cell Provider
SFC	Service Function Chain
SG	Smart Grid
SLA	Service Level Agreement
SSO	Single Sign-On
SSI	Self-Sovereign Identity
TL	Transaction Latency
TPS	Transactions per Second
TSM	Telecommunication Services Marketplace
TTP	Trusted Third-Party
VM	Virtual Machine
VNF	Virtual Network Function

Chapter One

Introduction

Distributed systems are an integral part of today's computing infrastructures, enabling the collaboration of multiple nodes to work towards a common goal. Further, distributed systems enable efficient system scalability and fault tolerance, allowing organizations to build more robust and highly available infrastructures. Being distributed, the system can be built with centralized governance, which implies that a single organization controls, *i. e. ,* governs, all processes within the infrastructure. In the use case of *centralized marketplaces* [2], the governor organization is a provider of guarantees, *i. e. ,* digital trust [3], that the operations executed on behalf of users are secure and not tampered with. Further, the marketplace provides data privacy [4], ensuring user data is processed and stored according to the agreements and regulations. Consequently, centralized marketplaces act as providers of identity and access management (IAM) [5], governance [6], and trade settlement [7] services centrally, containing all processes and operations within a single organization's boundaries. The first use case that this thesis investigates is an AI engineering marketplace where centralized governance is applied. Such a marketplace is discussed next.

AI Engineering Marketplace: Access to machine learning (ML) [8] algorithms implemented in open-source software development kits has lowered the bar for incorporating artificial intelligence (AI) [9] into general-purpose applications. However, building robust and efficient AI applications requires expert knowledge on how AI algorithms and data, *i. e. ,* AI artifacts, interact. The AI algorithm development, model training, and application benchmarking are executed in several stages as an AI pipeline. ML requires training data typically available in organizations with large data processing infrastructure. However, small-to-medium enterprises (SMEs) [10] may define efficient ML models, but must rely on collaborations with data-owning organizations to test their AI solutions in an AI pipeline. Hence, trusted AI engineering collaborations are needed to empower ML beyond large companies and conduct AI pipeline execution [11]. The AI marketplace can enable such a trusted collaboration, providing a platform for trading AI artifacts and executing AI pipelines. However, such an AI marketplace must meet the privacy, trust, and regulatory requirements for collaborative AI engineering. For large organizations, *i. e. ,* data owners, the data is the asset traded in the AI marketplace and represents financial value. Thus, data owners require guarantees that the data is not susceptible to tampering or exfiltration during AI pipeline execution by application developer SMEs. Further, SMEs do not want to share the code of ML models and AI applications with the data owners. Thus, data owners and SMEs require an organization, *i. e. ,* a trusted third party (TTP) [12], that can govern the marketplace, ensuring AI asset privacy and preventing tampering and exfiltration. Here, a centralized marketplace model can be applied, as TTP is required to act as an impartial intermediary in AI pipeline execution, enabling trust and AI asset privacy. Moreover, TTP ensures that AI assets can be used only within marketplace infrastructure and by having a valid license.

A TTP enables digital trust and privacy in a centrally governed marketplace. However, centralization may limit the rapid expansion of the marketplace in its respective domain due

to restricted processes interoperability and automation. Thus, a *decentralized marketplace* [13] model can be adopted to enable governance distribution among multiple organizations, enhancing marketplace scalability and bringing automation and interoperability to the services of collaborating governor organizations. However, *trust issues are raised if more than one organization has to govern the marketplace.* In the case of a *decentralized marketplace*, IAM, governance, and trade settlement processes execution is distributed among all organizations involved in its governance. Thus, a decentralized marketplace requires a robust and secure digital trust-enabling mechanism between collaborating organizations. Further, a decentralized marketplace system must enable collaborating organizations to process and store private data for further usage in trade settlements. The second use case that this thesis investigates is a peer-to-peer (P2P) renewable energy marketplace where a decentralized governance model is applied. Such a marketplace is discussed next.

Renewable Energy Marketplace: Energy distribution systems play a vital role in modern societies. However, the electricity production conducted by power plants that work on fossil fuels results in atmosphere carbonization. In order to make electricity generation cleaner, renewable energy sources (RES), *e. g.,* solar panels, were introduced as an alternative to fossil fuel [14]. Consequently, the introduction of RES opened opportunities for electricity prosumers, *i. e.,* producers/consumers, to become a part of the grid as a distributed energy resource (DER) [15]. This allows prosumers to consume energy as a conventional node but also produce and output energy to the grid [16]. Further, prosumers can also trade the produced electricity through the energy marketplace. However, today's energy markets face a number of challenges when it comes to management and operation. The first is the *inflexible pricing model* of today's marketplaces, where the prosumer is limited to selling the generated electricity to a single buyer without any other options [17], *i. e.,* typically prosumer's energy provider (EP). The second is *inaccurate green consumption information*, *i. e.,* buyers receive unreliable information about the sources of the electricity they consume. Due to the inflexibility of energy distribution systems, *e. g.,* unavailability of RES in close proximity to consumers, they still end up using the electricity produced by fossil fuel energy sources even when paid for renewable [18]. These limitations can be alleviated by introducing *peer-to-peer* (P2P) *electricity trading*, which is an automated sale process for renewable energy between market participants using a contract with pre-determined conditions [19]. A P2P energy trade settlement allows prosumers to trade electricity directly with each other, enabling them to control when, where, and for what price the electricity is bought or sold. The ultimate goal of P2P energy trading is the widespread adoption of RES, resulting in the decarbonization of the energy distribution systems [20].

Today's energy marketplaces are built as centralized systems. Thus, a TTP (typically prosumer's EP) acts as a guarantee that the conditions of a P2P energy trading contract are followed. However, *trust issues are raised when scaling the marketplace to more than one EP.* EPs want to keep their operations private to maintain a competitive advantage in the electricity market. This requires the introduction of an external TTP that can be trusted by all energy providers within the marketplace [21], *i. e.,* allowing individuals belonging to different energy providers to trade with each other. To remediate this limitation, a decentralized marketplace model can distribute governance over the marketplace operations to multiple EPs. However, all organizations require an efficient and robust digital trust-

enabling mechanism that guarantees that P2P trade settlement conditions are followed while maintaining EPs' data privacy. Such capabilities can be provided by blockchain technology [22]. Blockchain provides marketplace participants with distributed storage, *i. e.,* the ledger, and brings such benefits as provenance, accountability, and privacy to all data processed in a system. It also acts as a consensus-reaching platform, allowing initially non-trusting energy providers and prosumers to establish a trusted relationship and conduct P2P trade settlements without needing a single TTP acting as a middleman [12].

This thesis explores the distributed digital marketplace model and demonstrates the shift from centralized to decentralized privacy and trust. The included studies demonstrate that a centralized marketplace model is capable of providing necessary trust and privacy capabilities for specific use cases, *e. g.,* AI engineering. However, centralization may limit the rapid development and expansion of the marketplace in its respective domain. Thus, our investigations shift towards the decentralized marketplace model, demonstrating that it enables digital trust of operations executed between multiple organizations while preserving their data privacy.

1.1 Distributed Systems Design Problems

A distributed system design has to meet a number of requirements regarding compliance, trust, and privacy to be ready for further implementation and production deployment. Both governance models, *i. e.,* centralized and decentralized, have their system specifics resulting in different service complexity, privacy and trust implications, and operational overhead. Thus, a range of problems have to be addressed in distributed systems, where their complexity varies based on the chosen governance model.

1.1.1 Identity and Access Management (IAM)

Distributed systems require a robust and secure IAM mechanism. In a centralized governance model, the advantage is in the ability of a single organization to govern IAM processes and establish the perimeter defenses around identity information storage. As shown in **Paper I**, with a proper system security analysis and countermeasures placed, a centralized marketplace can provide an IAM mechanism and protect identity information. However, centralized IAM mechanisms represent a single point of failure and expose data stored in the identity storage in the case of a system breach. In contrast, decentralized IAM mechanisms distribute governance over identity verification and authorization processes to several organizations. In this case, the decentralized governance model removes the ability to control the IAM processes and store identity information centrally. While such an approach eliminates a single point of failure, it increases the complexity of IAM processes and identity information storage. The decentralized IAM requires a consensus, *i. e.,* agreement, mechanism, which provides guarantees to organizations that IAM processes are executed according to agreed conditions. Further, the decentralized IAM requires distributed identity information storage, which enables authorization interoperability and self-sovereignty. These requirements increase system complexity and operational overhead. As shown in **Paper III**, the decentralized marketplace requires an IAM system where all prosumers and EPs must be authorized for

energy trade execution.

1.1.2 Data Privacy

Data privacy is essential for any system. It provides guarantees that the user data is processed and stored in a secure fashion. Further, it relies on an established IAM mechanism, which enforces only authorized data access. Data privacy plays a key role in securing users' assets in distributed marketplaces where financial operations occur. In centralized marketplaces, privacy is implemented within a single organization's data storage. As **Paper I** demonstrates, data privacy can be implemented and enforced in controlled, centralized marketplace environments. However, in a decentralized marketplace system, data privacy must be implemented across multiple organizations. Further, a decentralized marketplace must provide mechanisms for private data disclosure and validity verification. When disclosing user data across nodes in a decentralized system, organizations need guarantees that private data is processed according to marketplace regulations. As demonstrated in **Papers III** and **IV**, depending on the approach used, systems can provide different validity and provenance guarantees on private data disclosure during marketplace trade settlement. Further, **Paper V** demonstrates that legal regulations can limit decentralized data privacy and may need to be adapted in the future.

1.1.3 Digital Trust

Digital trust, *i. e.,* guarantees, is a characteristic of distributed systems that defines the assurance that the data processing and general service execution is done according to specific security policies derived from rules and regulations. In the case of centralized marketplaces, digital trust is an inherent characteristic and is established by the user accepting the system operations. As demonstrated in **Paper I**, the digital trust can be established within a centrally governed, restricted environment controlled by the TTP. However, the system can be trusted only as long as TTP provides sufficient execution guarantees and does not turn rogue. Thus, the decentralized marketplace aims at distributing digital trust to several governing organizations. However, to build digital trust in a decentralized marketplace, organizations must have a mechanism to validate data processing and trade operations execution. As demonstrated in **Papers III**, **IV**, and **V**, such validation mechanism is achieved through a multi-step trade transaction life-cycle, where distributed marketplace utilizes consensus mechanism and predefined agreed-upon operation algorithm, *i. e.,* smart contract (SC) [23].

1.1.4 Scalability and Performance

Distributed systems are designed to scale the computational infrastructure aiming at the support of a large number of users [24]. The centralized governance model assumes a central point of interaction with a system. Such a central point may introduce a bottleneck due to the high volume of traffic, limiting the system's performance. Further, horizontal or vertical scaling may be expensive for a single organization that governs a centralized system. As demonstrated in **Paper I**, the centralized marketplace can be designed to overcome

the congestion of a central point of interaction at a high cost of excessive computational infrastructure deployment. Thus, a decentralized system distributes the financial burden of scalability among several organizations, enabling more flexible horizontal system expansion. Further, in a decentralized system, each organization can process requests independently, distributing the load to multiple nodes and eliminating a centralized bottleneck. A decentralized system provides high fault tolerance, *i. e.,* in case one organization is down, the system can still operate. However, considering the need for decentralized trust, the process of consensus reaching can represent a considerable system bottleneck. As demonstrated in **Papers III**, **IV**, and **V**, a detailed system design process is needed to provide sufficient performance and scalability within a decentralized system.

1.1.5 Regulatory Compliance

Depending on the use case, a marketplace system may be subject to regulatory compliance [19, 25]. Regulatory documents within a marketplace system define the roles and responsibilities of all system actors. Such regulations are essential to establish countermeasures that deter illegal actions. Further, regulations enforcement establishes trust for parts of the system that cannot be controlled digitally, *e. g.,* electricity generation and transmission through the energy grid. As demonstrated in **Paper II**, a centralized marketplace is subject to multiple regulatory constraints due to excessive responsibility undertaken by a governor organization. This requires the governor to provide data privacy and trusted execution guarantees centrally, which have to be maintained for marketplace users during trade settlement. In case of a system breach, the governor bears full responsibility for users' data and assets. In a decentralized marketplace, regulatory responsibility for trusted execution is distributed between multiple organizations governing the marketplace. This requires establishing a mechanism that takes into account the area of responsibility of each involved organization. As demonstrated in **Paper III**, regulatory responsibility can be undertaken by specific organizations within a decentralized marketplace, which may result in partial system centralization. Further, **Paper V** demonstrates that partial centralization can be avoided by updating the regulation to the needs of a decentralized marketplace.

1.1.6 Sustainability

Considering the scale of distributed systems, the sustainability of a long-term operation must be taken into account [26]. A large number of computational resources in a distributed system may raise an issue of excessive energy consumption which leads to harmful environmental effects. In such a case, the distributed system has to be designed with the vision of resource efficiency, minimizing energy consumption. Further, the system operations must be evaluated for optimal performance, *e. g.,* fine-tuning algorithms and data access patterns. For a centralized system, control over the infrastructure allows for maintaining resource efficiency and optimal performance. As demonstrated in **Paper I**, the centralized marketplace can allocate the resources needed for collaborative AI engineering while providing guarantees on trusted execution and optimizing the performance of all operations. However, in decentralized systems, trust is established through a consensus-reaching process, which may be a computationally expensive operation. It may lead to excessive energy consumption

by a decentralized system, which results in atmosphere carbonization. Thus, decentralized systems require efficient techniques for trusted operations execution which provide optimized performance and minimize energy consumption.

1.2 Research Aim

The aim of this thesis is *to advance the knowledge on the efficient design and evaluation of distributed marketplaces with an emphasis on trust and privacy.* The trust and privacy characteristics are intrinsic to centralized marketplaces. However, in the case of decentralized marketplaces, trust and privacy are dispersed over several governing entities. Thus, this requires new techniques to build decentralized trust and ensure data privacy in such multi-actor systems. In order to address the identified aim, this thesis defines and addresses the following two objectives:

1. *To design and evaluate the centralized digital marketplace with an emphasis on digital trust and data privacy.*

   This objective is fulfilled in **Paper I**, where we investigate the centralized digital marketplace application in the context of collaborative AI engineering. The main constraint of the AI marketplace is the necessity to protect the AI assets of collaborating organizations during AI pipeline execution. We define a Secure Virtual Premise (SVP) deployed in a centralized marketplace governed by a TTP. SVP enables AI pipeline execution while protecting the AI assets from exfiltration.

2. *To design and evaluate the decentralized digital marketplace based on blockchain technology with an emphasis on digital trust and data privacy.*

   This objective is fulfilled in **Papers II**, **III**, **IV**, and **V**, where the concepts of decentralized privacy and trust are investigated in the context of the renewable energy marketplace. In particular, the papers investigate the usage of blockchain as a main trust-enabling technology in digital marketplaces. The blockchain provides guarantees that P2P renewable energy trade settlements are executed according to the automated contract while enabling the privacy of data according to the requirements of marketplace governing organizations.

1.3 Thesis Outline

The outline of the included studies and their relation to the thesis aim and objectives is provided in Figure 1.1. This thesis is divided into two parts. In the first part of the thesis, in order to address the first objective, we focus on the centralized marketplace model. We define the requirements and architecture of SVP in the context of collaborative AI engineering, where trust and privacy requirements are posed due to sensitive data processing in the system and intellectual property rights (IPR) [27] enforcement (**Paper I**). Further, in order to address the second objective, we move to the second part of the thesis. First, we describe the shift from the centralized to the decentralized marketplace model by surveying state-of-the-art and providing an in-depth investigation of the advantages and limitations of these two

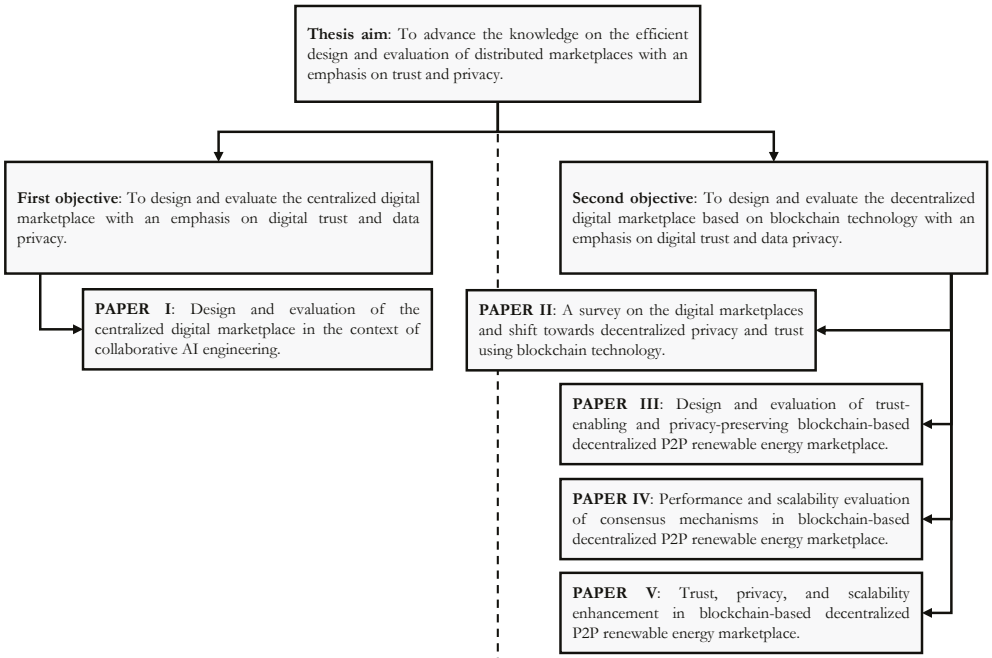


Figure 1.1: Papers which are included in the thesis. *The relation to the thesis aim and respective research objectives is shown. In addition, each research objective outlines a part of this thesis.*

governance types. In addition, we outline the usage of blockchain in the marketplace model and its ability to enable decentralized trust and privacy. We define decentralized marketplace core services and survey blockchain-based proposals on their design and implementation (**Paper II**). Next, we investigate a blockchain-based decentralized system model in the context of renewable energy marketplaces. We define the P2P renewable energy marketplace requirements in compliance with the European Union (EU) [19] electricity trade regulations. Further, we design a decentralized renewable energy marketplace based on Hyperledger Fabric (HF) [28] enterprise blockchain platform. We provide a complete architecture definition and analyze energy trade settlement privacy and trusted execution guarantees. Moreover, we conduct system performance and scalability evaluation and analyze HF's Crash-Fault Tolerant (CFT) [29] consensus mechanism (**Paper III**). Next, we investigate the performance and scalability of Byzantine Fault Tolerant (BFT) [30] consensus mechanisms in the P2P renewable energy marketplace based on Hyperledger Besu (HB) [31] enterprise blockchain platform. The advantage of BFT consensus over CFT is the ability to withstand malicious blockchain nodes. We design the marketplace according to the identified requirements, describe system architecture, and conduct the performance and scalability evaluation (**Paper IV**). In the final paper of this thesis, we address the limitations of trust and scalability of the HF-based renewable energy marketplace by proposing improvements toward the consensus mechanism and system architecture. We propose an improvement of P2P energy trade regulations which enhances organizations' data privacy and makes the marketplace more

flexible. Finally, we address marketplace scalability through a new multilayered energy marketplace model (**Paper V**).

1.4 Disposition

The remainder of the thesis is organized as follows:

Chapter 2: Background

This chapter describes the technologies and concepts used in the studies included in this thesis. It starts with a more general view of distributed system architecture and a discussion of its advantages and limitations. Further, we discuss data privacy and digital trust characteristics and how they are achieved in centralized and decentralized governance models. Finally, we discuss distributed marketplace design principles and outline the related work on collaborative AI and renewable energy market proposals.

Chapter 3: Research Questions and Methodology

We continue with the discussion of the research methodology used to conduct our investigations. In this chapter, we introduce the research problem and identify research questions. Further, we outline the information retrieval process and describe the research methodology used for use cases investigation. Finally, we discuss research validity threats and their remediation.

Chapter 4: Results

This chapter presents results of this thesis. We answer each research question with the respective papers included in this thesis. In addition, we outline the lessons learned during conducted investigations and pinpoint the advantages and limitations of decentralized systems design.

Chapter 5: Conclusions and Future Work

Finally, this chapter provides a summary of the thesis and discusses future research directions.

Chapter Two

Background

In this chapter, the required background knowledge in *distributed systems* and *digital marketplaces* [32] is introduced. Figure 2.1 depicts the structure and relationships between background concepts and papers included in this thesis. The main focus of this thesis is the investigation of the efficient design of privacy and trust in distributed digital marketplaces. To this end, we investigate ways to enable privacy and trust in the context of *centralized* and *decentralized governance* models [33]. This thesis investigates the marketplace design as a complete system, taking into consideration application, security, performance, and regulatory characteristics. The innovation of the included papers comes out of investigating design decisions that combine these characteristics to make the resulting system applicable in a real-world context.

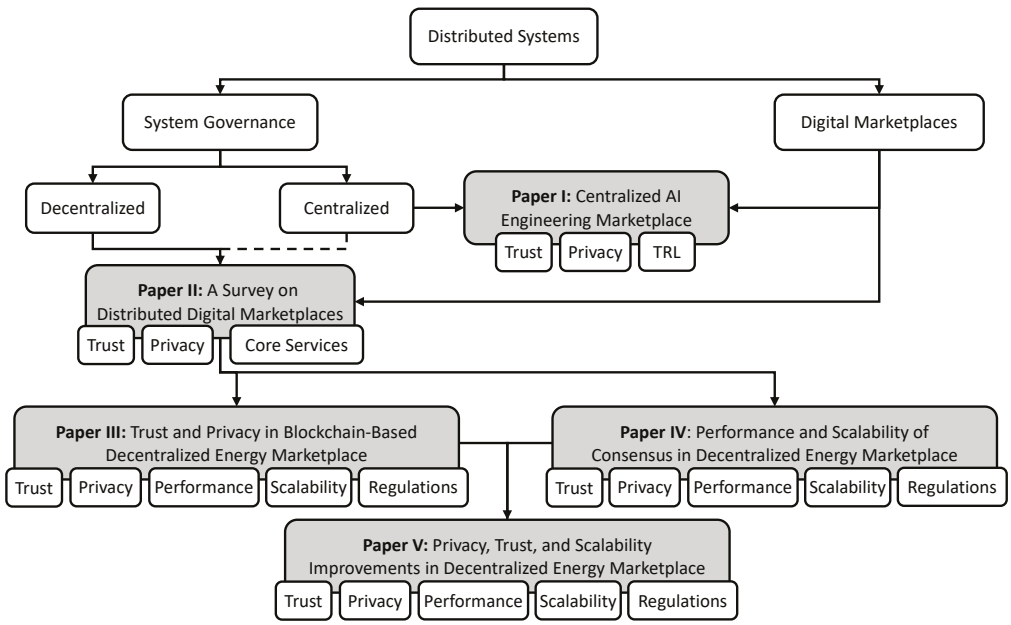


Figure 2.1: *Background knowledge map for the thesis. Boxes with a grey background indicate the topics of the included papers.*

The central topic of each included paper is depicted in Figure 2.1 in boxes with a grey background. Further, each grey box contains system characteristics investigated in the respective paper, *i. e.* , small boxes under each paper outline. Finally, each paper is connected to the fundamental concepts we employ to conduct the investigation. Some concepts are not discussed in this chapter due to extended discussions provided in the studies included in this

thesis. The reader is referred to **Paper I** for a discussion on *Collaborative AI Engineering*, **Paper II** on *Blockchains*, and **Paper III** on *Renewable Energy Marketplaces*.

The remainder of this chapter is structured as follows. A discussion on the concepts of distributed systems and digital marketplaces is provided, followed by the related work on distributed marketplaces.

2.1 Distributed Systems

There are a number of architectures that were defined to address the needs of computing systems. The distributed system architecture and its centralized and decentralized governance models are depicted in Figure 2.2. The *Distributed* part of the figure indicates all nodes dedicated to application infrastructure, where white circles depict *system nodes*. Further, the distributed infrastructure scheme is used to demonstrate *Centralized* and *Decentralized* governance models. The dashed circles in centralized and decentralized depictions denote a perimeter(s) that is/are governed by an organization(s). Dark circles depict *system interaction nodes*, and grey circles depict *user nodes*. The user nodes do not run any computing infrastructure, instead utilizing the interaction nodes to access services provided by the governing organization(s). Further, the distributed system architecture is discussed from the point of view of governance over the computing infrastructure as well as *privacy* and *digital trust* assurance within the services that comprise it. In addition, different governance types have advantages and limitations for asset trade execution within distributed marketplaces. Next, the distributed architecture is discussed in detail, along with centralized and decentralized governance types.

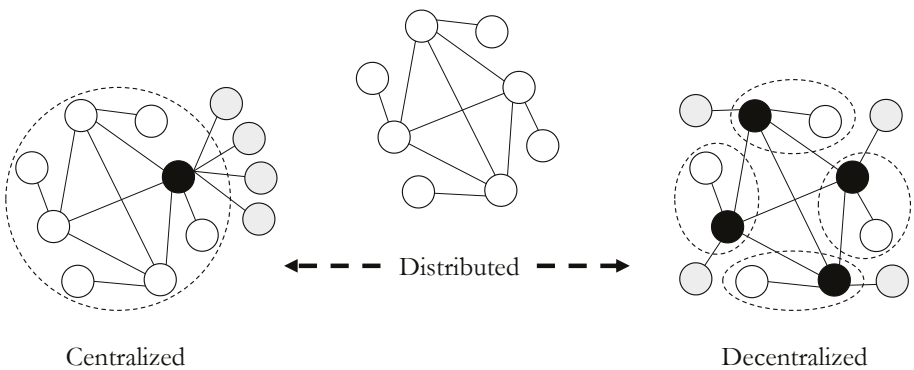


Figure 2.2: Distributed system governance models. *(Grey circles depict user nodes, dark circles depict system interaction nodes, and white circles depict system nodes. Dashed circles denote the perimeter governed by an organization.*

2.1.1 Distributed Architecture

The systems with *distributed architecture* comprise a network of nodes that are interconnected with each other, creating an infrastructure that works towards a common goal. Such a system allows for the distribution of computing tasks across multiple nodes, improving

horizontal scalability, fault tolerance, and performance. Distributed architecture refers to the *dissemination of computing resources*, in contrast to the monolith approach.

The inherent characteristic of distributed systems is *scalability*. The distribution allows for horizontal scalability, *i. e.*, adding additional nodes, and enables an increased number of users the system can support. Further, services can be loosely coupled and scaled independently to be distributed over multiple computing nodes. Communication in a distributed network is performed via messages, *e. g.*, data packets, that are exchanged between the nodes. These messages contain the data and instructions that define network behavior. The communication can be implemented via various inter-process communication (IPC) [34] mechanisms, *e. g.*, message queues [35] or remote procedure calls (RPC) [36].A message queue assumes that the communication is established through concrete data packets, *i. e.*, messages, which are stored in the node's buffer, *i. e.*, queue, to be further processed. RPC assumes that nodes have active services running where functions can be triggered remotely. These approaches may also be used as a part of a peer-to-peer (P2P) [37] model of communication, where all nodes communicate directly with each other, acting either as a client, *i. e.*, when requesting information, and a server, *i. e.*, when serving information, at the same time. In general, each node is referred to as a peer.

Another distributed systems' characteristic is *fault tolerance*. One of the ways to enable fault tolerance is redundancy through horizontal infrastructure scaling. Redundancy makes the service more resilient to failures by deploying a copy that can be used when a primary instance fails. Further, through the employment of load balancing, the distributed system can automate service recovery and prevent outages in case one of the instances fails. In addition, the overall system fault tolerance can be increased through computation distribution over multiple nodes, preventing individual resources from overloading and improving system performance. The advanced techniques to ensure fault tolerance in distributed systems include self-healing and self-reconfiguration, which involve the usage of health monitoring to identify infrastructure failures.

Further, the data in a system may be distributed and shared among multiple data storages. In such a case, *data consistency* and *synchronization* have to be maintained to ensure the integrity of system services operation. For distributed data storage, a *consensus mechanism* is required to keep data synchronized across multiple nodes. A consensus mechanism defines a data synchronization algorithm and employs techniques such as digital signatures during the asynchronous exchange to ensure data integrity. The existing consensus mechanisms are developed for the purposes of the infrastructure they are designed for, *e. g.*, blockchain-based systems, *c. f.*, **Paper II**.

Distributed systems require careful consideration of *performance* characteristics. Throughput, *i. e.*, number of operations per second, affects the number of concurrent users that can interact with system services. Further, latency, *i. e.*, the time needed to execute an operation, affects the Quality of Service (QoS) [38] provided by the distributed system. Our investigations demonstrate that various factors such as consensus mechanisms, node hardware, and system size affect resulting performance, *c. f.*, **Papers III** and **IV**. Thus, careful design has to be defined where topology, hardware, and communications mechanisms are chosen according to the needs of the resulting distributed system.

Further, the management of a distributed system poses several challenges regarding nodes'

configuration management. Configuration management is an essential process to ensure correct updates of the node software. Distributed systems tend to suffer from configuration drift, *i. e.,* node configurations start to differ after several updates. Thus, Infrastructure as Code (IaC) [39] tools are commonly used to automate distributed systems management through machine-readable configuration files, ensuring correct deployment, scaling, and updates.

A major challenge in distributed systems is ensuring *data privacy* and *digital trust*. Privacy refers to data that belongs to an individual and may contain personally identifiable information (PII) [40]. PII is often subject to regulations, *e. g.,* General Data Protection Regulation (GDPR) [25], where the system has to enable certain extended functionality, *e. g.,* GDPR's Article 17 denotes the user's right of complete erasure from the system. Techniques like encryption and digital signatures enable data integrity and privacy at transit and at rest. Further, a robust IAM mechanism ensures data privacy by enforcing only authorized data access within the system. Depending on the system governance, *i. e.,* centralized or decentralized, ensuring privacy may require the deployment of specific techniques to enable private data integrity checks, *e. g.,* consensus mechanism execution in the decentralized private data processing. Further, distributed systems must ensure *digital trust*, *i. e.,* guarantees of correct process execution, which is the characteristic that defines the degree of assurance to the users that services operate according to the established conditions, *e. g.,* trade settlement execution within a marketplace. Digital trust is highly reliant on the type of distributed system governance, *i. e.,* centralized or decentralized, and requires the implementation of specific protocols to provide guarantees of correct service execution, *e. g.,* blockchain transaction life-cycle involving techniques like encryption and digital signatures. Thus, depending on the system governance model, distributed systems require specific design decisions to enable digital trust and data privacy. Further, the centralized and decentralized governance models toward distributed system design are discussed.

2.1.2 Centralized Governance

The distributed systems with a *centralized governance* (further *centralized systems*) rely on a single organization, *i. e.,* the governor, that controls the entire distributed infrastructure, *c. f.,* *Centralized* in Figure 2.2. The governor is responsible for the implementation of system services and their exposure to users. The centralized systems define an interaction node, *i. e.,* central node, that exposes services to users. The central node can be implemented as multiple physical computers, *i. e.,* load balanced cluster of servers appearing as one. In this type of governance, a bilateral exchange of information is executed between the central node and users that utilize the system. Nowadays, the majority of enterprises build their systems with centralized governance.

Due to the presence of a central node, the centralized system *scalability* may be limited considering the number of system interaction points. However, there are ways to mitigate these limitations within a distributed infrastructure. The centralized system may have a sophisticated load-balancing solution, which comprises multiple central nodes appearing as one interaction point, increasing the number of concurrent users. Further, the centralized system may be instructed to deploy dedicated computing resources for specific users and redirect them towards newly instantiated nodes, *c. f.,* **Paper I**.

In terms of *fault tolerance*, the central node represents a single point of system failure, *i. e. ,* if it fails, the entire distributed system is rendered unavailable for users. Thus, the central node is often subject to Denial of Service (DoS) [41] attacks, where it is flooded with superfluous requests with the aim of saturating the available capacity causing valid requests to be dropped. The advantage of centralized systems is the ability to implement perimeter-based defenses over the computing infrastructure. Such mechanisms as firewalls, intrusion detection systems (IDS), and intrusion protection systems (IPS) [42] can be set up within centrally governed distributed infrastructure, protecting the system's perimeter against breaches [43]. Further, the central node may have a sophisticated load-balancing solution that redirects DoS traffic to a honeypot and passes through only valid user requests.

The *performance* of the centralized system is directly related to the throughput and latency that are supported by the central node and the slowest system service, *i. e. ,* bottleneck. The horizontal scaling of the central node and efficient load balancing may further improve system performance. In addition, centralized governance results in better control over the distributed system, ensuring optimal resource utilization across the entire infrastructure. Further, centrally governed systems enable comprehensive system monitoring to identify performance bottlenecks.

The governor organization implements and coordinates all activities within the system. Thus, it acts as *data privacy* and *digital trust* provider to the users. Data privacy is ensured by the IAM mechanism, which is provided centrally by the governing organization. Further, the communication between users and the central node can be based on digital signatures and encryption to further enhance data privacy and integrity. Next, the governor acts as the only decision-maker, as all data processing and storing functionality is concealed within its boundaries. In this way, digital trust becomes an inherent system characteristic, and all clients have to accept that the governor performs data processing in a secure and privacy-preserving way before starting to use any services, *c. f. ,* **Paper I**. The centralized system's digital trust can also be enhanced by regulatory compliance, implementation of legal frameworks and contracts, independent audits, and third-party verification.

2.1.3 Decentralized Governance

In distributed systems with *decentralized governance* (further *decentralized systems*), there is no single organization that controls the entire infrastructure. Instead, multiple interconnected organizations govern parts of the distributed infrastructure, *c. f. , Decentralized* in Figure 2.2. Each organization governs its own part of the distributed system, which has an interaction node that accepts requests from clients. Each interaction node processes client requests and, if required, communicates system changes to other organizations.

Separately, each organization's governance zone within a decentralized system inherits characteristics of centralized governance in terms of scalability, fault tolerance, and performance. Further, each organization separately acts as a data privacy and digital trust provider to the users utilizing its system interaction node. However, to enable the interaction between the organizations and establish coherent decentralized governance, a new approach has to be employed.

It is assumed that organizations involved in the governance of decentralized system services do not necessarily trust each other, requiring an assurance that the data which is

processed on a foreign part of the system is not being manipulated with malicious intent. Thus, in order to support decentralized services execution, a *digital trust*-enabling mechanism is required, which ensures compliant data processing. First, as services are executed in a decentralized environment, they require distributed storage which enables organizations to maintain an updated system state. A robust and secure *consensus mechanism* [44] enables data synchronization and consistency in distributed storage. Further, decentralized services are defined as automated[1] functions, which all organizations have to approve before they are deployed in the infrastructure. The consensus mechanism is used to ensure data correctness and compliant data processing during automated services execution, building digital trust among organizations in a decentralized system, *c. f. ,* **Papers II**, **III**, **IV**.

Organizations can enable *data privacy* in the part of a decentralized system they govern, ensuring users' data is stored and processed securely. However, if private data has to be used in decentralized services, guarantees of compliant data processing have to be provided to ensure consistency and integrity. For example, in blockchain technology, such guarantees can be provided by computing the proof, *e. g. ,* encrypted transaction hash, that the private data modification has happened. Further, the proof is disseminated via a consensus mechanism to organizations not involved in private data processing. Having the proof saved in the distributed storage, organizations are able to verify the validity of private data if it is disclosed for the purposes of specific decentralized service execution, *e. g. ,* asset trade within a marketplace, *c. f. ,* **Papers III**, **IV**, and **V**.

In terms of *fault tolerance*, each organization can organize the defense of its governance area, *i. e. ,* protecting the system interaction point. However, inter-organization communication requires additional fault tolerance mechanisms deployment in a decentralized system. Mainly, it concerns distributed data storage and decentralized services execution. Thus, a consensus mechanism, as a main digital trust-enabling entity, must be fault tolerant. In this thesis, we investigate two types of consensus mechanisms used for decentralized enterprise applications, *i. e. ,* CFT and BFT. CFT consensus mechanisms are protected from node failures, *i. e. ,* if less than 50% of the nodes fail, the network can operate successfully. If consensus is BFT, it is both CFT and can operate in the presence of adversaries, *e. g. ,* nodes that manipulate the data synchronization process, *c. f. ,* **Papers III**, **IV**, **V**.

In terms of *performance*, the main limitation is the system process that requires the most computations. To provide sufficient guarantees, consensus mechanism execution is a multi-step process that involves techniques such as digital signatures and encryption. This represents a significant computational strain on the decentralized system and results in a performance bottleneck. Further, as our studies demonstrate, the improved security of BFT consensus mechanisms may come at the cost of decreased performance compared to CFT ones. The exhaustive performance evaluation and fine-tuning of decentralized system parameters may contribute to improving service throughput and latency, *c. f. ,* **Papers III**, **IV**, and **V**.

The decentralized systems *scalability* may be limited due to the computations needed to execute a consensus mechanism. The increasing amount of organizations in a decentralized system increases the computations needed to maintain data consistency in a distributed

[1] Automation is referred to as an ability to define a concrete process block that can be executed as a transaction and result in a new system state, *i. e. ,* modifying or creating data in distributed data storage.

storage. Further, the more organizations there are, the more computations are needed for the decentralized service function execution to provide necessary guarantees of digital trust. As our studies demonstrate, achieving sufficient decentralized systems' scalability characteristics requires careful design considerations, *c. f. ,* **Papers III**, **IV**, and **V**.

One of the possibilities is to use *blockchain technology* as the data privacy and digital trust-enabling mechanism in decentralized systems. It provides distributed data storage, *i. e. ,* immutable ledger, and brings such benefits as provenance, consistency, and accountability to all data processed within a decentralized system. It also permits the implementation of decentralized services as it enables the deployment and execution of automated functions, *i. e. ,* SC. Such decentralized services may represent a part of the system's functionality or support required security and compliance mechanisms, *e. g. ,* IAM, data assurance, and governance. In order for the SC to be deployed in the decentralized blockchain-based system, all collaborating parties have to agree on the correctness of the workflow defined in it. Once deployed, a SC is stored on an immutable ledger, where organizations can trigger its execution. The SC acts as a guarantee of precisely what code is being executed as well as on what data it is operating [45]. In addition, some blockchains enable private transaction execution, providing guarantees of compliant data processing to the organization in a decentralized system.

2.2 Digital Marketplaces

Digital marketplaces are a widely accepted concept for the formation of business opportunities. In general, digital marketplaces are defined to meet the requirements of *supply and demand* concepts. The digital marketplace model defined in this thesis is shown in Figure 2.3. It enables the supply of traded assets to the target audience and allows for increasing seller revenue, trade process flexibility, and expansion of the respective domain. Further, the marketplace system must provide *core services* that enable users, *i. e. ,* buyers and sellers, to conduct trade [46].

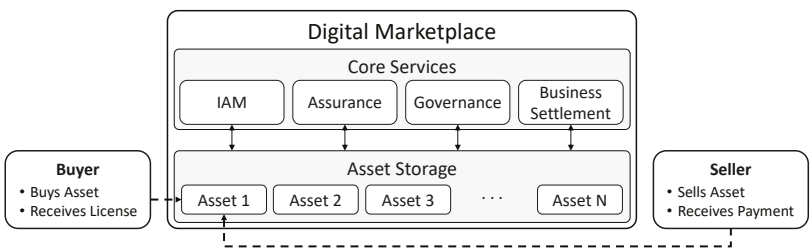


Figure 2.3: *Digital Marketplace Model.*

IAM [47–49] safeguards the marketplace from unauthorized access. The marketplace acts as a provider of user identity and the associated authentication and authorization mechanisms. The IAM ensures that users execute only actions that they are authorized for. Further, The IAM enables data privacy, ensuring only authorized access to users' private data. The IAM service may be based on digital signatures and associated access control

(AC) mechanisms defining users' rights based on a specific pattern, *e. g.,* role-based access control (RBAC) [50].

Assurance [51–53] builds confidence users have in the marketplace services. It begins with the guarantees, *i. e.,* digital trust, of marketplace services execution by ensuring their reliability and integrity. Further, the assurance can be enhanced by the compliance of the marketplace system with related regulations, laws, and standards, *e. g.,* GDPR or D2018/2001. The dispute resolution system can reinforce assurance by ensuring fairness in conflict resolution. Next, service transparency can enhance assurance by demonstrating processes and policies openly to the marketplace users. Finally, marketplace data security can improve assurance by enabling privacy and protection for data in transit and at rest, *e. g.,* by employing encryption and AC.

Governance [54–56] ensures compliance with the internal marketplace regulations. The regulations define policies and mechanisms that enable efficient marketplace operation. In addition, governance outlines the decision-making process, defining the rights and responsibilities of organizations involved in the marketplace operation. Further, governance ensures the implementation of legal documents to ensure compliance with the regulations within the marketplace domain. Finally, governance ensures the accountability of parties operating within the marketplace, establishing the mechanisms to enforce penalties in the case of violations. Efficient governance builds trust among the governing organization(s) and users and contributes to the expansion and reputation of the marketplace system.

Business settlement [57–59] ensures correct financial transaction execution and the distribution of assets and funds between the involved parties, *e. g.,* buyer and seller in Figure 2.3. The business settlement service ensures that the trade contract is executed according to the agreed-upon conditions and all involved parties get their respective reward. The *licensing approach* [60] is used to regulate the business settlement within the marketplace. Licensing defines the trade conditions between the seller and buyer in a document, *i. e., license*, which is signed when the business settlement is executed. The licensing approach makes the business settlement transparent and assures users of the bought asset ownership rights. Finally, after the license is signed, the seller receives a *payment*, which is charged by the marketplace on his/her behalf.

The marketplace core services provide the minimum necessary functionality for the digital marketplace to operate. Further, the functionality may be expanded based on the requirements of the specific use case. The deployment and operation of marketplace core services depend on the system governance model, *i. e.,* centralized or decentralized. In a centralized marketplace model, the core services are provided by a single governor organization, *i. e.,* trusted third party (TTP), which acts as an intermediary in trade transactions between buyers and sellers. However, being an intermediary in a business settlement, the governor can dictate rules and trade transaction payment distribution. Consequently, it may lead to a value distribution imbalance between the marketplace and sellers. Thus, a decentralized marketplace model can distribute governance, bringing competition to the organizations running the infrastructure. Driven by business interest, governor organizations create transparent market rules ensuring accountability and compliance within a decentralized marketplace. For a detailed discussion of the particular decentralized marketplace use cases, the reader is referred to **Paper II**.

2.3 Related Work

The area of digital marketplaces generates research interest in academia and industry due to the benefits provided for respective domains' accelerated development and expansion, *e. g.*, AI algorithm development or energy trading. The centralized marketplaces provide ownership-preserving capabilities by ensuring asset protection within an infrastructure controlled by TTP. Further, the widespread adoption of blockchain technology opened new opportunities in designing decentralized P2P renewable energy marketplaces. The proposals on AI and energy marketplaces are discussed next.

2.3.1 Collaborative AI Marketplaces

The proposals on collaborative AI engineering are investigated with the application of different marketplace governance models, *i. e.*, centralized and decentralized. The authors of [61] propose a blockchain-based collaborative ML model training in a decentralized marketplace using HF private permissioned platform. The marketplace incentivizes all system actors to record their actions within the ML model training process on the distributed ledger to provide verifiable evidence of correct execution. Authors claim their marketplace can train up to 70 ML models per second on a 12-peer blockchain network. In [62], the authors outline guidelines for the AI diffusion method based on a decentralized blockchain-enabled marketplace model. The authors consider multiple aspects of the AI engineering marketplace, including technical, economic, and regulatory requirements. In addition, the authors provide a comparison of today's AI marketplaces in operation. Authors claim that the majority of today's marketplaces are built as centrally governed systems. The authors of [63] propose a decentralized P2P ML model training platform. The platform ensures data privacy and protects against the disclosure of information. The authors claim their approach outperforms previous work while providing ML model quality under privacy constraints. In [64], the authors propose a new decentralized deep learning algorithm based on Leader-Follower topology and a differential privacy model. The algorithm aims at optimizing the system's performance during private data processing. The authors claim that their algorithm outperforms other state-of-the-art decentralized learning competitors by reducing the computation by 30% while increasing the final accuracy result.

2.3.2 Renewable Energy Marketplaces

Several proposals have emerged in the field of decentralized blockchain-based energy marketplaces. In [65], the authors propose a platform for blockchain-based P2P energy trading. This platform aims to provide prosumers with the ability to trade electricity after the generation period and not before, as it is done in energy markets today. According to the authors, blockchain technology enables bilateral energy trading after the generation period by enabling prosumers to trade electricity in P2P mode without the need for TTP. Authors of [66] propose a hybrid blockchain-based P2P energy marketplace that enables electricity trading without a need for TTP. The marketplace aims at enabling an efficient energy market while reducing costs and improving energy distribution strategies. In [67], the authors propose a blockchain-based energy trading platform that utilizes ML with the

aim of enhancing the distribution of the energy generated by the DER. In addition, ML is used to analyze the electricity production data and propose better energy production strategies. The authors claim that the proposed trading platform enables the construction of efficient energy distribution strategies while maintaining QoS. The authors of [68] define a decentralized P2P energy marketplace architecture that provides a unified electricity trade settlement model. The marketplace is based on HB private permissioned blockchain platform. The authors conduct a performance evaluation of an implemented marketplace with HB's Istanbul BFT (IBFT) consensus mechanism. Further, they compare IBFT-based system results with Ethereum's Proof of Work (PoW) and Clique consensus mechanisms. The authors claim that their system demonstrates nearly double throughput compared to existing PoW-based systems. Further, the proposed model provides lower latency and optimizes the energy trading process, *i. e. ,* reduces the number of transactions needed to trade electricity. In [69], the authors propose a new blockchain-based P2P energy trading platform to address the challenge of decentralized electricity production and trading. The multilayered platform consists of the market and blockchain layers. The market layer represents an auction that allows prosumers to bid on energy trading proposals. According to the authors, an auction model results in an efficient market solution that preserves users' privacy and allows inter-temporal market product trading. The blockchain layer is used for SC implementation and the automation of decentralized marketplace functions. The authors conduct a platform evaluation that demonstrates the increased efficiency of energy trading via blockchain-based settlements. The authors of [70] conduct an in-depth investigation on the role of blockchain technology in microgrids. According to the authors, blockchain technology enables potential solutions for electrification in the transportation, building, and industrial sectors. Further, blockchain-based microgrids enhance the electrification opportunities for remote areas, *e. g. ,* islands, towards a green networking ecosystem. The authors claim that their study may serve as a comprehensive reference for modern microgrids, *i. e. ,* their control and communication technology with the integration of blockchain services for the sustainable energy supply chain. In [71], the authors propose a P2P energy marketplace based on novel strategies for bilateral electricity trade, *i. e. ,* supply and demand matching based on a distance between producer and consumer. Energy trade strategies evaluation demonstrates more efficient energy distribution while reducing the energy price. For further reading on the developments in blockchain-based energy marketplaces see [72].

The topic of energy grid management has also gained traction in recent years, investigating topics such as efficient energy distribution strategies, demand-supply matching, and balancing electricity prices. Authors of [73] propose a blockchain-based virtual power plant (VPP) management platform which addresses energy aggregation flexibility and operation of community microgrids. Further, the proposed platform enables P2P electricity trading between market participants. Authors claim that the proposed platform can successfully enable small-scale VPPs deployment and operation and promote grid decarbonization. In [74], authors present a novel framework aimed at the optimization of DERs aggregators capacity using mixed non-linear programming. The analysis of the demand data within the developed framework improves the optimization of energy distribution accuracy, resulting in a significantly increased prosumer revenue. According to the authors, the energy trade revenue increases up to 29.8% in comparison to the currently used heuristic approach.

Chapter Three

Research Questions and Methodology

The main focus of this thesis is to advance the knowledge of the efficient design and evaluation of distributed marketplaces with an emphasis on trust and privacy. The main subject of this thesis is *Computer Science* (CS) [75]. The main CS area discussed in this thesis is *Distributed Systems*. From the viewpoint of distributed systems, we focus on the investigation of trust and data privacy design within centralized and decentralized system architectures to enable digital marketplace services. Further, we focus on decentralized systems' performance and scalability characteristics for marketplaces built on blockchain technology. Finally, designed marketplaces are compliant with the official regulations on asset trading and data protection.

The main challenge in distributed systems investigation is that different service architectures require the application of specific approaches to enable data privacy and trusted execution. In a centralized architecture, trust and privacy are governed by a single organization to enable sensitive data processing and asset safety, *e. g.*, AI marketplace. Thus, associated security features may require the definition of a specific system design that protects the system and confidential data from unauthorized manipulation. In contrast, in a decentralized architecture, multiple non-trusting parties govern the system and processes, *e. g.*, renewable energy marketplace. In order to enable trust and privacy within a decentralized system, a consensus mechanism is required to allow robust and trusted process execution for collaborating parties.

This chapter will outline and relate the research methods applied in this thesis to the specific research questions. The mapping of the research methods to the detailed findings will be done in Chapter 4.

3.1 Research Questions

The main question of this thesis can be formulated as: *How to design distributed digital marketplaces in the context of systems with specific trust, privacy, and regulatory requirements?* To address this question, four more specific research questions are defined. Next, these research questions are discussed in the context of studies that address them.

RQ1. *How can trust and privacy be achieved in a collaborative distributed AI engineering marketplace?*

The process of collaborative AI engineering requires the implementation of specific security requirements due to sensitive data processing in the system and IPR enforcement. This RQ aims to identify the advantages and limitations of the centralized marketplace model and its capabilities in enabling data privacy and trusted execution via a TTP. In **Paper I**, we define an execution environment for centralized marketplace design, which allows for addressing the requirements of AI engineering. We define marketplace design, provide in-depth security analysis, and conduct the system evaluation.

RQ2. *How can blockchains be used to integrate trust and privacy in decentralized digital telecommunication marketplaces?*

In decentralized systems, the blockchain is one of the key trust-enabling technologies. This RQ aims to identify the blockchain technology advantages and limitations in the context of digital marketplaces. Further, it aims at identifying research gaps and mapping state-of-the-art in the blockchain-based marketplace investigation. Thus, in **Paper II**, we explore how blockchain technology can be incorporated into digital telecommunication marketplaces to enable decentralized trust and privacy. First, we outline the concept of the marketplace and outline today's centralized marketplace design. Further, we provide a survey on blockchain-based marketplaces, as well as outline core services that enable decentralized privacy and trust. We demonstrate that blockchain technology enables decentralized marketplaces while bringing such benefits as immutability, accountability, and privacy to all data processed within a digital marketplace.

RQ3. *What are the performance and scalability characteristics of decentralized blockchain-based renewable energy marketplaces with trust, privacy, and regulatory requirements?*

Performance and scalability are key metrics that allow evaluation of the distributed system efficiency. This RQ aims to investigate the performance and scalability of the decentralized blockchain-based marketplaces designed for facilitating renewable energy transactions while considering privacy and regulatory requirements. Performance refers to the ability of the blockchain-based marketplace to execute trade settlement transactions. Scalability addresses the system's ability to support an increasing number of transactions as the marketplace expands. The RQ3 is addressed in two research papers, where the following sub-research questions investigate different blockchain-based renewable energy marketplace designs.

RQ3.1. *What are the performance and scalability characteristics of Hyperledger Fabric-based renewable energy marketplace with trust, privacy, and regulatory requirements?*

In **Paper III**, we investigate a decentralized renewable energy marketplace based on HF blockchain. We conduct the performance and scalability evaluation and outline lessons learned out of blockchain-based marketplace implementation.

RQ3.2. *What are the performance and scalability characteristics of Hyperledger Besu-based renewable energy marketplace with trust, privacy, and regulatory requirements?*

In **Paper IV**, we investigate a decentralized renewable energy marketplace based on HB blockchain. HB enables the execution of a special type of consensus that protects the blockchain network from malicious nodes. We conduct the performance and scalability evaluation and compare the results to the HF-based marketplace.

RQ4. *How can current regulations and blockchain-based systems be improved to provide enhanced privacy and scalability for renewable energy marketplaces?*

Investigated renewable energy marketplace is subject to regulatory compliance. The P2P energy trade is a governmentally regulated process where marketplace actors' rights and responsibilities are clearly defined. Further, the P2P energy trade process itself has to be executed via an automated contract with the usage of renewable electricity certificates, *i. e.*, guarantees of origin (GOs). **Pepers III** and **IV** demonstrate that current renewable energy trade regulations make the blockchain-based marketplace partially centralized around the governmental regulator due to its involvement in every P2P energy trade transaction. Further, a portion of prosumer private data has to be shared at all times with the governmental regulator, which limits the marketplace data privacy scheme. This RQ aims to enhance privacy and decentralization of the blockchain-based energy marketplace. In **Paper V**, we propose changes to current energy trade regulations. The updated regulation enables system design changes, improving marketplace data privacy and decentralization.

3.2 Research Methods

In order to address the defined research questions, this thesis uses several research methodologies to derive scientific results. For **Paper II**, we use the *literature survey* (LS) [76, 77] research method, which is a process of information collection with the aim of gaining a deeper understanding of a subject under investigation. In our survey, we map state-of-the-art academic and industrial results on blockchain-based marketplaces with an emphasis on privacy and trust. In **Papers I**, **III**, **IV**, and **V**, we use the experimental compute science (ECS) [78, 79] research methodology, which utilizes quantitative methods relying on numerical data collection, *e. g.*, performance or scalability data. In detail, we apply ECS to several *use cases* where we investigate distributed systems and their design to enable efficient privacy and trust. Further, we discuss the LS and ECS research methods in detail and describe steps to achieve scientific results.

3.2.1 Literature Survey (LS)

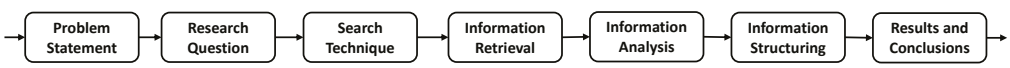


Figure 3.1: Literature survey methodology design used in Paper II.

The LS research method aims to identify and evaluate relevant sources, both from academia and industry. The outcome of a LS research method is a structured knowledge of the current state-of-the-art and gained insights that contribute to the research question answering. For our LS, we define methodological steps shown in Figure 3.1. Each methodological step defines

concrete boundaries of work on a specific task that we want to achieve. Further, each LS methodology step is discussed in detail.

1. **Problem Statement:** In this step, we describe the problems and related research gaps the survey aims to address. The problem statement provides a clear overview of the research subject and investigated areas. This methodological step also highlights the significance and relevance of the identified problem. Finally, the problem statement describes the broader implications of the survey to demonstrate its importance within the research subject.

2. **Research Question:** In this step, we identify the specific inquiry that helps to address the identified problem. The research question must be focused and explore gaps in the existing literature. It also should align the overall purpose of the survey with the selection and analysis of relevant literature. The research question serves as a foundation for the next steps of the LS research method.

3. **Search Technique:** In this step, we define a systematic approach to searching for relevant literature. It includes determining appropriate keywords and search terms related to the research topic. Further, it involves selecting relevant databases and sources to search for the relevant information. The search technique step involves using advanced strings to refine search queries and exploit advanced search features of bibliographical databases.

4. **Information Retrieval:** In this step, we access and retrieve the academic and industrial literature based on the search technique identified. This step involves accessing relevant databases or online platforms and retrieving the selected literature. It may involve downloading open-access articles or obtaining permission for restricted materials.

5. **Information Analysis:** In this step, we review and analyze the retrieved literature. This step includes the evaluation of the selected literature and the identification of common themes. Further, it involves identifying gaps or problematic areas within retrieved literature. The goal of this step is to derive insights and knowledge from the retrieved literature that can contribute to the general knowledge on the subject and address the identified research question.

6. **Information Structuring:** In this step, we organize and structure the extracted information from the analyzed literature. It includes categorizing and grouping relevant findings and organizing them into concepts based on their relationships. The goal is to provide a coherent structure to the gathered knowledge and identify the main arguments within the analyzed literature, enhancing the overall understanding of the research topic.

7. **Results and Conclusions:** In this step, we summarize the findings and draw overall conclusions from the analyzed literature. We summarize the key results and trends identified during the analysis. Based on the identified studies, we draw conclusions that address the identified research question. This step is crucial for LS, presenting a comprehensive overview of the state-of-the-art on the research area.

This methodology follows the *spiral approach* [80] towards the final research results retrieval. Throughout the execution of the above methodological steps, one can return to the research question step to include constraints and concepts identified during information analysis. The research question may be broadened to include more concepts or narrowed down to concentrate on more specific characteristics. Such research question refinement can be executed consequently to make the retrieved information sufficient and applicable for continuous research results production. When the results are obtained, such a methodology can be used for a new research question where the LS method is applicable.

3.2.2 Experimental Computer Science (ECS)

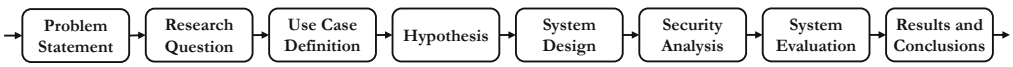


Figure 3.2: Experimental Computer Science methodology design used in Papers I, III, IV, and V.

The ECS research method involves designing and conducting experiments to investigate specific research questions in CS. This method evaluates algorithms and systems within a controlled experiment environment with the aim of measuring a specific variable. For our ECS, we define methodological steps shown in Figure 3.2. Further, each ECS methodology step is discussed in detail.

1. **Problem Statement:** In this step, we define the research problem and objectives the experiment aims to address. This step describes the specific challenge that the experiment investigates. The problem statement provides the rationale for conducting the experiment and highlights the relevance of the research problem. It serves as a foundation for the experiment design and helps to select investigated variables.

2. **Research Question:** In this step, we formulate the specific inquiry that helps to address the identified problem. The research question provides a direction for the experiment and influences the design, implementation, and evaluation steps. The research question serves as a basis for determining experimental conditions and defining the independent, *i. e.,* controlled, and dependent, *i. e.,* measured, variables.

3. **Use Case Definition:** In this step, we define the details of the investigated use case and set experiment boundaries. This step describes the system requirements, regulations, and context, *i. e.,* a use case, in which the experiment will take place. The use case definition helps in setting experimental conditions and workflows. Further, this step ensures that the experiment is applicable to real-world situations, allowing for its practical application.

4. **Hypothesis:** In this step, we formulate a testable statement about the expected outcomes of the experiment. The hypothesis is based on existing knowledge from the

use case definition step and serves as a prediction that can be validated through the experiment. It has to be falsifiable and establish concrete boundaries for a results evaluation framework.

5. **System Design:** In this step, we describe a detailed architecture of the system under investigation. The design specifies the system's functionality and services. It also involves selecting appropriate technologies, *e. g. ,* blockchain, to implement the defined functionality. This step aims to create a design that is used in the implementation phase of the experiment, ensuring that the system is aligned with the use case definition.

6. **Security Analysis:** In this step, we assess the security aspects of a system under investigation. This step identifies potential system security threats. The security analysis aims at identifying threat countermeasures to enhance system security. This step is crucial for ensuring the system's integrity, confidentiality, and availability and evaluating resilience against potential attacks or breaches [81].

7. **System Evaluation:** In this step, we implement the system according to the defined design and conduct experiments to collect data. The implemented system is evaluated under predefined experimental conditions to test the hypothesis. The implementation and evaluation step provides empirical evidence that helps to answer the research question.

8. **Results and Conclusions:** In this step, we analyze the data collected during the experiment to draw conclusions. Based on the analysis of the results, conclusions are drawn regarding the validity of the hypothesis and the implications for the research objectives. The results and conclusions provide valuable insights into the field of CS, contributing to general knowledge and the investigated system improvements.

This methodology follows the spiral approach toward the final results retrieval and validation. It assumes one can return to the research question if previously unknown constraints or concepts were identified during the system design step. The research question refinement can be executed consequently to improve use-case system design. When the implementation is evaluated and results are verified, such a research methodology can be used for a new research question where the ECS method is applicable. Next, the research methodology of the included papers is described in detail.

3.2.3 Security Evaluation

The evaluation of security features defined within a computational system plays an essential role in the correctness of its operation and protection against unauthorized access. The overall security of the system cannot be described with a constant value, as it is a continuously changing characteristic, which has to be repeatedly evaluated and maintained [82]. In **Papers I** and **III**, the security analysis plays a vital role in the research methodology process as it allows to address and implement required security and privacy mechanisms. Further, to provide appropriate scientific reasoning and justify the inclusion of proposed security mechanisms, a system-wide security evaluation is required.

There are a number of quantitative methods to evaluate and verify a particular security protocol within a computational system. For such purposes, we can use symbolical verification methods [83], *i. e.,* formal model, or computational approaches [84], *e. g.,* reduction techniques. However, the application of quantitative methods becomes problematic when a system-wide security evaluation is needed, as it combines a variety of security mechanisms implemented within the system. This restricts the evaluation by the quantitative methods and requires the application of a differentiated approach.

The system's architecture is designed to address the requirements of specific use cases, where system-user and intra-system interactions are considered. One of the approaches to evaluate the overall system's security is the analysis of such interactions from the point of view of the adversary, which intends to disrupt systems operation or obtain unauthorized access. This allows confirmation of implemented security mechanisms enforcement if the adversary is prevented from harming the system's operation or gaining unauthorized access.

With these considerations, in **Paper I**, we conduct a qualitative security analysis to perform a system-wide security evaluation. This method is executed according to defined methodological steps. First, we describe a preliminary architecture that does not contain any security mechanisms. Based on the preliminary architecture, we define the capabilities of an adversary that attacks the system. This step is done to create the boundary for the security analysis stage, which enables the security threats definition and contributes towards understating which security features are being considered. Further, a security analysis is conducted, where the required security mechanisms are chosen based on the identified threats. These security mechanisms are applied to the preliminary architecture, and the system's implementation is performed. Finally, a set of practical use cases is executed on a live system, where both intended functionality and malicious exploits are executed. When the results of executed actions are obtained, they allow evaluation of the system's protection against disruption and unauthorized access. If all malicious exploits fail to breach the system's security mechanisms while intended functionality executes successfully, we can judge that for this specific case study architecture, the trustworthy system security state is achieved. If at least one security protocol is breached, the architecture is considered unreliable, and the research methodology process has to be repeated from the security analysis step.

In **Paper I**, the use case of collaborative AI marketplace poses specific trust and privacy requirements due to sensitive data processing and IPR enforcement. During practical use-cases execution on the implemented system, defined security mechanisms were successfully enforced, protecting the system from adversaries, while the legitimate users were able to conduct the intended functionality of AI pipeline execution without interruptions. Thus, we were able to justify the inclusion of defined security mechanisms and complete the system-wide security evaluation.

In **Paper III**, we conducted a security analysis from perspectives of data privacy and trusted execution in a decentralized renewable energy marketplace. The blockchain network and marketplace interface (MI) act as a security scheme that protects the prosumers' assets, *e. g.,* personal wallets. We design security countermeasures to protect the assets at risk from any outside or inside adversary. The outside adversaries do not have authorized system access. Thus, they may try to intercept the traffic and likely tamper with the data exchanged between the prosumer and MI. As a countermeasure, the traffic between the prosumer and

MI has to be encrypted and authenticated, *i. e.,* protected from spoofing and tampering. For inside adversaries, we use blockchain technology as a security scheme, utilizing its IAM and data privacy features. The blockchain network's consensus mechanism and SCs protect transactions from spoofing, tampering, or repudiation. Further, private information access is possible only for authorized parties.

3.2.4 Information Retrieval and Analysis

This section outlines the way in which the required literature is retrieved and analyzed for LS described in **Paper II**. Also, it describes the information retrieval and analysis processes used for use cases defined in **Papers I**, **III**, and **IV**. **Paper V** works on the improvement based on the use cases investigated in **Papers III** and **IV**. Next, each included paper is discussed individually.

In **Paper II**, the information retrieval process is a part of LS methodology and conducted using the exhaustive *database search* [85]. The sources of information are bibliographic databases *Scopus*[1], *Web of Science*[2], *IEEE Xplore*[3], *ACM Digital Library*[4] and *Google Scholar*[5]. This set of databases is chosen to diversify the retrieved results and provide state-of-the-art research in the field of blockchain-based digital marketplaces. The analysis of the retrieved information targets the identification of the most relevant works. In addition, the analysis aims to present the information in a structured and comprehensive way, as far as a large amount of academic and industrial works have been identified and presented. The analysis resulted in a comprehensive literature survey where state-of-the-art research progress on blockchain-based digital marketplaces is reflected, as well as the core decentralized system services are described.

In **Papers I**, **III**, **IV**, and **V**, the information retrieval process is a part of ECS methodology, where the preliminary information for a use-case description is obtained to define requirements. In **Paper I**, the preliminary information is retrieved via related works investigation and communication with H2020 Bonseyes[6] project partners. Bonseyes partners contributed to the requirements definition process and finalization of preliminary architecture. In **Papers III**, **IV**, and **V**, the preliminary information is retrieved via related works, official regulatory documents, and communication with our Symphony[7] project partners. Symphony partners contributed to the energy marketplace use case definition process and regulatory compliance enforcement.

3.3 Validity Threats

The included papers are subject to validity threats that should be considered when performing an evaluation and validation of implemented systems. These validity threats should also be

[1] https://www.scopus.com
[2] https://clarivate.com/webofsciencegroup/solutions/web-of-science
[3] https://ieeexplore.ieee.org
[4] https://dl.acm.org/
[5] https://scholar.google.com
[6] https://www.bonseyes.eu/
[7] https://www.bth.se/eng/about-bth/departments/dida/symphony_kks/

considered in the case of research results application in the context of industrial production systems.

3.3.1 Construct Validity

The *construct validity* indicates the degree to which the research results reflect the theory or concepts they are based on [86]. In this case, the main threat to construct validity is the correctness of design and the reliability of the implementations provided in **Papers I**, **III**, **IV**, and **V**. In these studies, the described evaluation and validation results are executed on the test systems implemented with the use of specific open-source technologies, which are used by the industry, and considered to be reliable and secure.

To address the possibility of construct validity threats, in our studies, we provide implementation details, where we indicate the names of used technologies and their interaction within the designed system. In addition, we describe an entire methodology process depicted in Figure 3.2 and discuss how a system design can be obtained. In particular, for the system design, we describe the system functionality, services, and technologies. Following such a methodology, one can define a design for any system under consideration.

3.3.2 External Validity

Another threat to the research results is the *external validity*. It refers to the generalization of conclusions drawn in the included papers [86]. As a LS, **Paper II** investigates the academic and industrial proposals that have already been peer-reviewed, validated, or presented. However, there is a risk that the validation of results described in **Papers I**, **III**, **IV**, and **V** may contain external validity threats. Since the system design and the security analysis is performed for specific use cases of AI and renewable energy marketplaces, it may be challenging to generalize the obtained results to any use case in research fields different from the original one.

To decrease the effect of external validity threats, we base our use-cases design on well-defined and adopted technologies such as cloud infrastructures, blockchains, containerization, and public key infrastructure (PKI). These technologies are applied in multiple research branches where they serve as key components that enable the systems' distribution, privacy, and trust. In addition, we detail the process of use-cases system design definition and implementation, which enables the generalization of the methodological concepts to other research fields.

3.3.3 Internal Validity

The *internal validity* indicates whether the performed evaluation procedures and received results establish a trustworthy causal relationship and are not influenced by some other external uncontrolled variables or factors [86]. Internal validity allows the elimination of any other interpretations of retrieved results. Thus, it confirms the causal relationship of the used research method.

As a LS, **Paper II** maintains internal validity as described results and conclusions are based on the already published information. However, the case studies described in **Papers**

I, **III**, **IV**, and **V** may contain internal validity threats due to the system evaluation where multiple factors, *e. g.,* computing hardware or network bandwidth, influence the results of experiments execution. Thus, there is a risk that external factors may influence the evaluation results. To mitigate the threats to internal validity, we outline the boundaries of evaluation by isolating independent and dependent variables in a controlled experimental environment. This ensures a causal relationship between the independent and dependent variables and allows treating the final result as an aggregate of all causal relationships, which represent the system-wide performance and scalability.

3.3.4 Conclusion

The studies in this thesis investigate real-world use cases, where we take industrial systems requirements as a basis for the defined marketplace designs. The main threat to the validity is the specifics in the design and requirements for real-world use cases, which may limit the obtained results and their applicability in a general context, *i. e.,* other research domains. To address such validity threats and limitations, we have taken several measures. First, in our studies, we outline all steps performed to retrieve the respective research results. The design, implementation, and evaluation steps are described in detail to provide needed insight. This provides the means to reproduce the research results and, if needed, adapt the research methodology process to the needs of the use cases in other research domains. Second, multiple representatives from both industry and academia have taken part in the process of a qualitative security evaluation in **Paper I**. In this way, diverse opinions on system security, trust, and privacy enforcement were collected from all parties involved in the requirements definition to increase the validity of evaluation results and draw conclusions. Further, performance evaluation results obtained in **Papers III**, **IV**, and **V** play a crucial role in assessing the efficiency of the proposed solutions. By quantitatively measuring factors such as performance and scalability, we provide objective insights into systems' behavior under different scenarios. This enables academia and industry representatives to assess the practical applicability of the proposed system designs in their respective domains.

Chapter Four

Results

This chapter discusses the results obtained during the research process. The descriptions of contributions follow the methodology described in Chapters 3.2.1 and 3.2.2. Here, a condensed version of the text provided in the papers is used to point out the main issues addressed and contributions provided. Further, we test the defined hypotheses and answer the respective research questions in the context of each included paper.

4.1 (Paper I) Towards a Secure Proxy-based Architecture for Collaborative AI Engineering

Problem Statement: The AI marketplace can enable a trusted collaboration, providing a platform for trading AI artifacts and executing AI pipelines. However, such an AI marketplace must meet the privacy and trust requirements of collaborative AI engineering. Considering this problem statement, the next research question has been defined:

Research Question: *How can trust and privacy be achieved in a collaborative distributed AI engineering marketplace? (RQ1)*

Use Case Definition: Collaborative AI engineering poses trust requirements towards AI algorithm development, model training, and application benchmarking, *i. e.,* AI pipeline execution. Further, the AI pipeline execution process poses requirements towards AI models and algorithms, *i. e.,* AI assets, in terms of their privacy and IRPs. These requirements can be addressed via a centralized marketplace model, where a TTP provides a trusted AI pipeline execution environment. This paper proposes to address the privacy and trust requirements for collaborative AI engineering by designing the *Secure Virtual Premise* (SVP), which is a proxy-based PaaS execution environment. The SVP can be executed on-demand, *i. e.,* dynamically when artifacts are instantiated, on a dedicated computational infrastructure controlled by the TTP, where it shields the AI artifacts and pipelines from adversaries. Further, we define the main hypothesis of this study:

Hypothesis: *Secure Virtual Premise architecture enables trust and asset privacy in a collaborative AI engineering marketplace.*

System Design: The considered approach for engineering AI solutions [87] is conducted via the use of AI artifacts and ML pipelines, called *AI pipelines*. These pipelines are used to structure and eventually automate ML workflows. They consist of several modular steps to generate an AI application or to benchmark its accuracy.

The key element of collaborative AI engineering is the marketplace (MP) for AI artifacts. The MP enables organizations to trade AI artifacts and agree on terms and conditions for collaborations and artifact usage, *i. e.,* AI assets licensing. However, while the MP is an open platform, the SVP must be a closed, protected, and controlled space where the AI engineering tasks are executed. The SVP utilizes distributed computing and storage resources controlled by the MP, *i. e.,* TTP, and provides them to the organizations. Only eligible

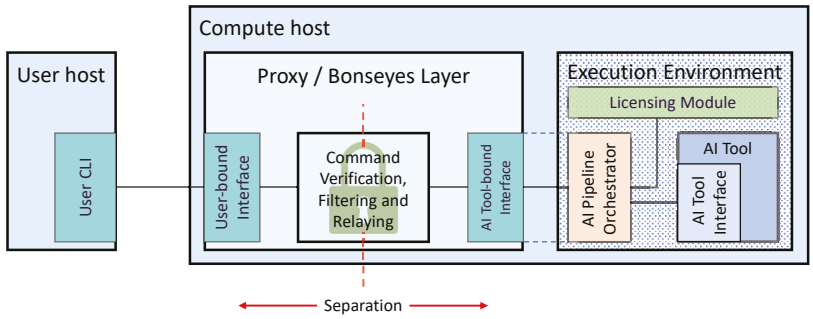


Figure 4.1: *Security and Separation by Bonseyes Layer.*

users should access a specific SVP. Resources and users on the inside, *i. e.*, on premise, are typically trustworthy and compliant, and everything on the outside is untrusted and potentially malicious. From a security perspective, the AI artifacts are the assets at risk, and the SVP is the security scheme that provides privacy and trusted AI pipeline execution. A successful attack on this scheme will grant adversary access to the asset, such that the asset can be exfiltrated or configured to engage in potentially malicious behavior.

Proxies provide security by applying the security concepts of separation and verification. They implement services such as IAM, communication encryption, and network isolation. The SVP architecture is based on the *sidecar proxy* concept [88, 89]. Sidecar proxies are attached to parent services and provide them with supporting features, *c. f.*, *Proxy / Bonseyes Layer (BL)* in Fig. 4.1. The SVP's BL shields the artifact from misuse, is implemented next to the artifact as a sidecar proxy, and is started on-demand on a dedicated computing infrastructure of the MP.

The system design is depicted in Figure 4.2. The SVP consists of the controller and the compute hosts. The controller host is used to instantiate the compute host's AI pipeline execution environment. Further, the controller host enforces marketplace license checking to ensure compliant usage of AI assets. Finally, the communication with the execution environment and execution of the AI pipeline is conducted through the BL.

Security Analysis: We identify the main attack paths targeting the respective APIs on the Controller and Compute hosts shown in Figure 4.2. These attack paths may lead to unauthorized access, privilege escalation, and interference with AI artifacts and pipelines. To address these threats, we utilize the STRIDE [90] approach to define the threat profile. Based on the threat profile, we propose several security requirements. First, we use X.509 digital certificates for user identity and authentication to prevent unauthorized access. Further, we utilize HTTPS with certificate-based mutual authentication for the controller and compute host communication. Next, we implement role-based access control for SVP API calls to ensure trusted AI pipeline execution. Finally, we implement the import volume, which allows the safe deposit of AI assets within the SVP and their further usage for pipeline execution. This way, the SVP preserves AI assets' privacy and prevents exfiltration.

System Evaluation: Several real-world AI engineering cases evaluated the SVP implementation. These use cases permit judging the SVP's privacy and trust and demonstrate its *Technology Readiness Level* (TRL) [91]. In the case of SVP, the TRL indicates its ability to

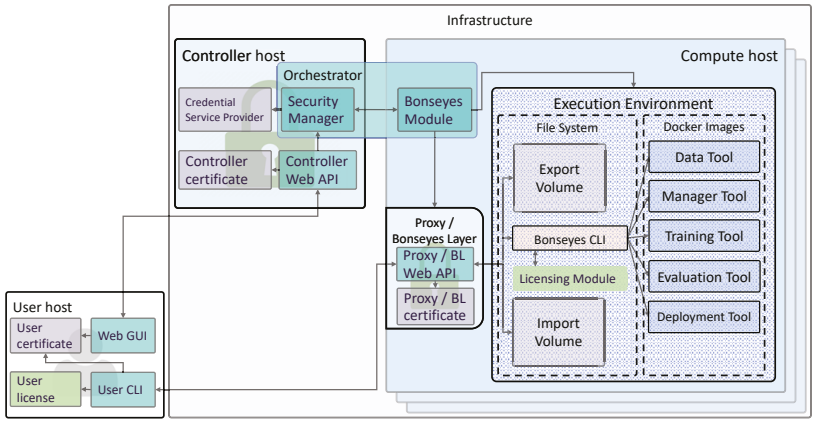


Figure 4.2: *The architecture of the SVP.*

execute AI pipelines while shielding AI assets from adversaries. The use cases were carried out on the computing infrastructures of stakeholders of the collaborative AI engineering process. The stakeholders in the use cases were assigned one of the following roles: *legitimate marketplace TTP, legitimate SVP user, rogue SVP user*, and *outside adversary*. Throughout these use cases, a legitimate SVP user and marketplace TTP were always able to execute the AI pipelines and exchange the artifacts. However, a rogue SVP user and an outside adversary were experiencing the enforcement of security mechanisms.

In the first use case, an outside adversary was given the task of penetrating into the SVP and interfering with the process of AI pipeline execution. When an adversary attempted a connection, it was blocked by the X.509 digital certificate authentication. In parallel, the embedded logging system recorded unsuccessful connection attempts for further analysis.

Another use case introduced a rogue SVP user on the inside of SVP's perimeter, having malicious goals to 1) obtain the AI assets stored in the SVP and 2) execute an AI benchmarking tool with an expired license. The Proxy / BL enforces for regular users a store-only mode for the Import Volume and lets only privileged users obtain access to the data inside. To complete the first malicious goal, the rogue user must authenticate using the digital certificate. Using the identity from the certificate, the RBAC mechanism determined that regular user privileges apply and thus blocked read access to the AI assets. To complete the second goal, the rogue user modified the license, encoded in plain-text JSON format, to extend the license expiration date. The modified license was uploaded as part of the pipeline construction process to the SVP, along with the original license signature. During license verification, the SVP first verified the license signature. Since the license's content was changed, the signature verification failed. Thus, the execution of the benchmarking tool was blocked.

Results and Conclusions: The use cases confirmed that the AI and security requirements were appropriately addressed. Further, the evaluation confirmed the hypothesis of this study, demonstrating that the security capabilities of the SVP provide sufficient privacy and trust to enable collaborative AI engineering. Finally, the SVP implementation demonstrated TRL 4, *i. e.*, technology validated in the lab. To our knowledge, the SVP is the first PaaS that meets the needs of distributed and collaborative AI engineering.

4.2 (Paper II) A Survey on Blockchain-based Telecommunication Services Marketplaces

Problem Statement: Centralized digital marketplaces are governed by a TTP, which acts as a trust and privacy provider for the market participants and their transactions. However, centralization may limit the expansion of the marketplace in its respective domain due to restricted processes interoperability and automation. Thus, a *decentralized marketplace* model can be adopted to enable governance distribution among multiple organizations. However, scaling the marketplace to multiple governors and eliminating a single TTP requires providing the marketplace actors with core services to perform the business transaction in a decentralized and trusted manner. Blockchain technology may enable decentralized business transaction execution.

Research Question: *How can blockchains be used to integrate trust and privacy in decentralized digital telecommunication marketplaces? (RQ2)*

Search Technique: In this study, we survey blockchain-based digital marketplaces using the LS research method. The search technique process is conducted using the exhaustive *database search* in combination with the *snowballing* method. To create the search strings, a number of keywords were used: *telecommunication, marketplace, blockchain, service, identity management, assurance, governance, business settlement.* Also, multiple variations of these words were constructed, *i. e.*, plural forms and different word combinations. The sources of information are bibliographic databases *Scopus*, *Web of Science*, *IEEE Xplore*, *ACM Digital Library*, and *Google Scholar*. This set of databases is chosen to diversify the retrieved results and provide state-of-the-art research in the field of blockchain-based digital marketplaces.

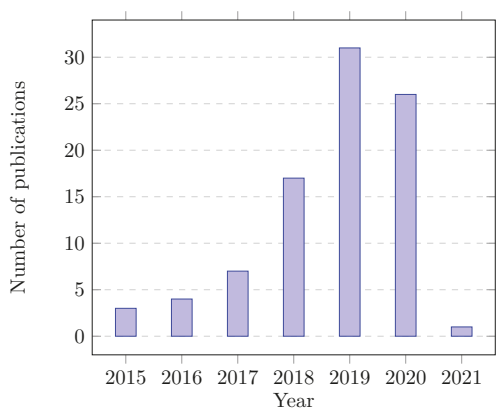


Figure 4.3: *Number of related publications per publication year.*

Information Retrieval: The main search criteria for related publications is the presence of a discussion on blockchain technology's ability to integrate privacy and trust in the context of decentralized telecommunication services marketplaces (TSMs). According to the search we conducted, there are a limited number of proposals that specifically target TSMs based on blockchains. As a consequence, our search was extended to works that discuss the advantages and disadvantages of blockchain-enabled trust and privacy in the context of

digital marketplaces at large. Fig. 4.3 presents a number of publications on blockchain-based marketplaces per publication year. Further, to correctly define the blockchain-based marketplace core services, we include the main standardization activities in TSMs. Finally, the works which investigate digital marketplaces' core blockchain-based services were included as well.

Information Analysis: One of the aims of this paper is to demonstrate the shift from a centralized to a decentralized marketplace model. Thus, we outline the generalized marketplace model, applicable to centralized and decentralized governance, and outline the core services which provide trust and privacy. In general, digital marketplaces are defined to meet the requirements of the concepts of *supply and demand*. The marketplace allows products to be supplied to customers with increased speed and stimulates the popularity and expansion of the software. The core services include IAM, data assurance, governance, and business settlement.

Further, we provide a comprehensive description of today's centralized marketplace types that are in operation, *i. e. ,* application and cloud services marketplaces. They follow different asset delivery models which provide business capabilities for both individual customers and organizations. Application marketplaces (AM) [92] help to sell and deliver an application, *e. g. ,* smartphone app, at an increased speed, *i. e. ,* for developers, and convenience, *i. e. ,* for customers. Further, cloud services marketplaces (CSM) [93, 94] allow the construction of service function chains (SFC) [95] on the infrastructure governed by a TTP. Organizations may require SFC to execute complex workflows which can be developed based on the cloud services provided by CSM. The SVP described in **Paper I** is a variation of SFC where an AI pipeline is constructed from cloud services running on an infrastructure governed by a TTP.

Centralization may limit the rapid expansion of the marketplace due to restricted business settlement interoperability and automation. Thus, a decentralized marketplace model was proposed to enable business settlement automation through governance distribution among multiple organizations. However, scaling the marketplace to multiple governors and eliminating a single TTP requires providing the marketplace actors with services to perform the business transaction in a decentralized and trusted manner. Blockchain technology is selected as the main trust-enabling technology to support decentralized services and business settlement automation. To describe the shift from a centralized model, we survey the proposals for decentralized blockchain-based marketplaces. All proposals are first analyzed and assigned to their respective domain, *e. g. ,* energy, cloud, IoT, or Telecom. Further, marketplace proposals are divided based on the blockchain technology used to implement them, *e. g. ,* Hyperledger Fabric, Ethereum, or IOTA [96]. Next, we analyze and discuss the decentralized blockchain-based marketplace standardization activities, with the aim of identifying core services needed for automated and trusted trade settlement execution, *i. e. ,* IAM, assurance, governance, and business settlement [97].

Information Structuring: The information on included works that investigate blockchain-based marketplaces is structured in Table 7.1, supported by extensive written explanations, *c. f. ,* Section 7.3.5. The included works are listed in chronological order along with their *application area* and *blockchain platform*. In addition, Section 7.3.7 provides an overview of standardization activities on TSMs. However, these standardization activities are also applicable in the context of decentralized blockchain-based marketplaces at large. Further,

Table 4.1: *Advantages of decentralized blockchain-based marketplaces vs. features of centralized marketplaces*

Core Service	Features of Centralized Marketplaces	Advantages by Decentralized Blockchain-based Marketplace
IAM	• A centralized authority manages a single identity database setup. • Centralized authority represents a single point of failure. • Restricts interoperability and reusability of digital identity.	• A decentralized network of nodes managing identity information. • Enables reusability and interoperability of identity. • Enables users full control over the identity information.
Assurance	• Data assurance is provided by the centralized authority. • Inability to inspect the marketplace by external parties. • Possibility of violations by the marketplace.	• Data assurance is provided by the immutability and transparency of the blockchain. • Blockchain's immutability ensures that the data the smart contract operates on is valid. • Transparency of operations for collaborating parties.
Governance	• Governance is performed solely by the marketplace operator. • All decisions are made within the centralized authority.	• Decentralization of governance within the system actors. • Increased automation, democratization, and time-efficiency of the governance activities.
Business Settlement	• Business settlement is executed and controlled by a central authority, which acts as a trusted third party. • May result in value distribution imbalance, as the marketplace may dictate billing rules and payments distribution.	• Smart contracts eliminate the need for a trusted third party. • Ability to verify the validity of the smart contract data. • Fair value distribution due to trusted, transparent, and automated business settlement.

Section 7.4 provides an in-depth overview of marketplace core services and blockchain applicability in their context. For better readability, Table 4.1 is included here to provide a summary of the features of centralized and decentralized marketplaces based on their core services. Finally, Section 7.5 discusses future research directions and ways to make marketplace core services more democratic and robust.

Results and Conclusions: The retrieved and analyzed proposals can be summarized in several blockchain application directions in the context of decentralized marketplaces. First, it is the reduction of the need for a TTP in the business settlement process to increase the fairness of value distribution. Eventually, the TTP may not be eliminated entirely since the marketplace is often subject to governmental regulations, *e. g.*, P2P energy trade marketplace. However, blockchain technology enables trusted and automated business transaction execution between marketplaces participants, where different trading concepts can be applied and enhanced, *e. g.*, auction bidding or fixed price selling. Next, blockchain enables the decentralized execution of marketplace core services, *i. e.*, IAM, assurance, and governance. Further, blockchain's immutable data storage provides auditing information that helps marketplace governors to dynamically resolve legal disputes, *e. g.*, non-sufficient network performance according to service level agreement. Finally, blockchain technology enables private transaction execution, preserving business-critical information privacy for collaborating organizations.

4.3 (Paper III) Towards Efficient Privacy and Trust in Decentralized Blockchain-Based Peer-to-Peer Renewable Energy Marketplace

Problem Statement: Today's renewable energy marketplaces are built as centralized systems, and TTP, *i. e.,* typically prosumer's EP, acts as a guarantee of P2P energy trade settlement execution. However, centralization is a limiting factor for marketplace expansion as it restricts the energy trade process automation and interoperability between multiple EPs. A decentralized electricity marketplace allows scaling the marketplace to more than one EP and enables P2P energy trade automation. However, a decentralized marketplace needs a trust-enabling mechanism that guarantees that P2P energy trade settlement conditions are followed while maintaining EPs' and prosumers' data privacy. Such capabilities may be provided by Hyperledger Fabric (HF) private permissioned blockchain.

Research Question: *What are the performance and scalability characteristics of Hyperledger Fabric-based renewable energy marketplace with trust, privacy, and regulatory requirements? (RQ3.1)*

Use Case Design: The renewable energy marketplace requirements were defined in collaboration with an EP that has DERs as a part of the electricity grid. Further, the energy marketplace actors and requirements were formulated in compliance with the regulations described in Directive 2018/2001 (D2018/2001) of the European Parliament [19]. D2018/2001 regulates the issuing, trading, and consumption of the Guarantee of Origin (GO), which is a document providing proof that a given quantity of energy was produced from renewable sources. As the marketplace works directly with the country's energy distribution system, the government oversees any operations on energy trading and transportation due to national security concerns. Thus, to be compliant with D2018/2001, our marketplace includes the *regulator* actor, which acts as a governmental representative and oversees the marketplace operations. Considering the data privacy requirements within the energy marketplace, we utilize HF private permissioned blockchain platform due to its trust-enabling and privacy-preserving capabilities. Further, we identify the main hypothesis of this study:

Hypothesis: *Hyperledger Fabric-based energy marketplace permits trusted energy trade execution and prosumer data privacy via endorse-order-validate transaction life-cycle and CFT consensus while proving sufficient throughput for a medium-sized energy community,* i. e., *up to 250 000 prosumers.*

System Design: Correct marketplace operation means all the system actors must have guarantees that the trade settlements are executed following market rules and regulations, maintaining data provenance and preventing tampering. HF provides marketplace participants with distributed storage, *i. e.,* the *ledger*, and brings such benefits as provenance and accountability to all data processed in a system. All transactions in HF are executed within a *blockchain channel*, which establishes a connection between the ledger participants. In HF, there are two types of nodes: *peers* and *orderers.* Further, each node performs a specific task: *endorsement* (peers), *ordering* (orderers), or *validation* (peers).

The trust that all transactions in the marketplace are following the predefined rules is provided by the *consensus mechanism*, *tamper resistance*, and *trusted execution capabilities* of HF. All marketplace functions are expressed as a *smart contract* (SC) that is audited by all the blockchain organizations, *i. e.,* EPs and regulators, and is stored in the ledger.

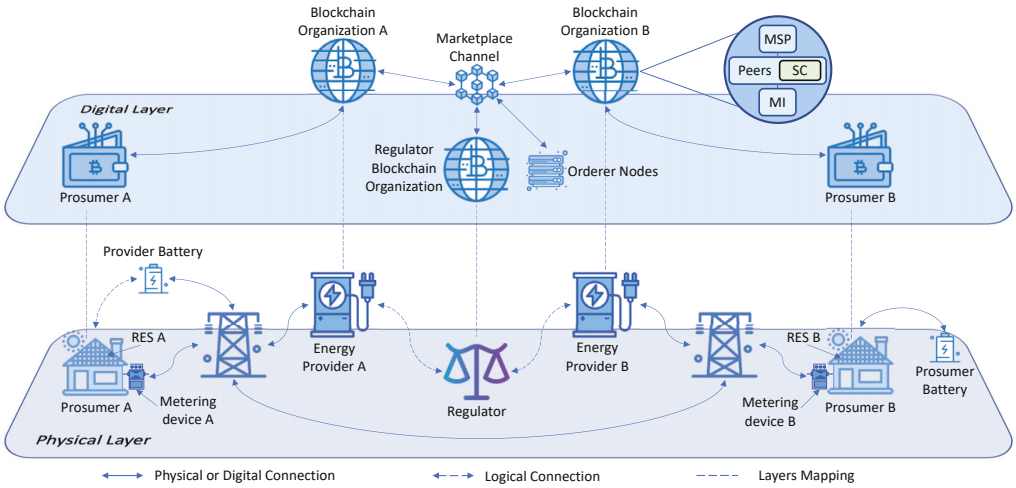


Figure 4.4: *P2P HF-based energy marketplace architecture.*

As a result, there is a clear consensus regarding the rules, expressed as programming code, that the transactions need to follow. Every action that the participants can take in the marketplace is implemented solely through the execution of this SC. Hence, relying on the guarantees provided by HF that the execution of the SC can be trusted, every marketplace transaction can be trusted to follow the rules.

In HF, the consensus mechanism is divided into two layers. The first layer denotes the *endorse-order-validate* transaction life-cycle. This life cycle provides the guarantees discussed above, as well as trusted SC execution. The second layer of the consensus mechanism is concentrated on the transaction ordering process. Hyperledger Fabric currently supports only *RAFT* protocol [98] to order transactions. RAFT is *crash fault tolerant* (CFT), *i. e.*, protecting only from orderer node failures. Thus, RAFT consensus needs to be executed by one or several trusted organizations within the blockchain network.

The architecture of the energy marketplace is depicted in Figure 4.4. It consists of two layers: *physical* and *digital*. The physical layer is the energy grid, where generated electricity is distributed. The digital layer is the communication network between the energy providers and prosumers where trading of the electricity, *i. e.*, virtual kWh, takes place. In order to correctly map both marketplace layers, each physical layer actor has to have representation in the digital layer, *c. f.*, Figure 4.4. EPs and the regulator act a separate *blockchain organization* (BO). The BO operates a number of peer nodes, which are the primary guarantors of valid transaction execution and require the most computational power. All peers are interconnected, forming the *marketplace channel* (MC) that establishes a connection between collaborating BOs. Further, each energy provider BO has a *marketplace interface* (MI). In addition, all BOs have a dedicated *membership service provider* (MSP) which generates cryptographic identity information for the prosumers who join the marketplace. Prosumers do not act as BO, *i. e.*, have no peers, relying on the EPs to endorse and validate trade settlement transactions on their behalf. They also have a *metering device* installed in their households, which monitors the RES-generated electricity and sends the data to the

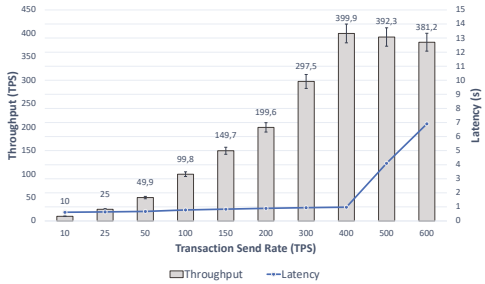
energy provider. Further, the regulator also maintains the *orderer nodes* responsible for the order of transactions in the block. As orderer nodes are an integral part of the MC, the entity that operates them can manipulate any transactions issued within the blockchain. *We assume that the regulator, being the governmental body, is not interested in any malicious activity and ensures the validity of all transaction ordering processes.*

The marketplace system supports a number of functions to enable the energy trade process. The *electricity generation registration* function registers data from the prosumer metering device in the prosumer's account in the blockchain. Further, the *GO management* function enables GOs' issuing and consumption. Next, the marketplace enables an energy *ordering system*, where prosumers can create buy and sell electricity orders. Finally, the marketplace enables a *P2P energy trading* process, where prosumers can trade with each other utilizing the computing infrastructure of their respective EP.

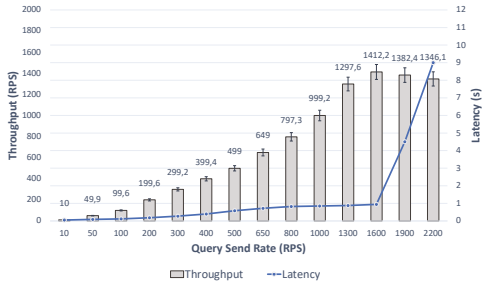
Security Analysis: From a security perspective, prosumer private data, GOs, and the trade transaction process are assets at risk. We identify the main attack paths targeting blockchain data handling and process execution. In order to comprise system security requirements, comprehensive threat modeling is executed. The identified threats include manipulation of the prosumer registered electricity, issuing fake GOs, and obtaining access to private data. To address these threats, we utilize the STRIDE approach to define the threat profile. Based on the threat profile, we propose several security requirements. First, it must be impossible to spoof, tamper, or repudiate any transaction executed within the marketplace. Further, no private information disclosure must be possible to an outsider and members of the blockchain. Next, within the smart contract, functions must be restricted based on the user's role. Finally, the marketplace system must alert blockchain organizations in case any modification of the configuration is outside of the proper governance model.

The blockchain-based marketplace works as a decentralized system where data privacy, IAM, and system governance are distributed between several BOs. Thus, attacks aimed at spoofing the execution of transactions with tampered data require gaining access to all BOs to perform a full endorse-order-validate life cycle, which is unlikely. Further, the regulator, being the governmental body, prevents transaction repudiation by controlling orderer nodes. However, if the aim is to spoof the identity of peer nodes, an improperly implemented MI may become the most vulnerable point of the system. Next, HF enables private data processing through the private data collection (PDC) feature. Thus, HF ensures that only members of PDC have access to the private data. Further, HF's SCs allow restricting function execution to a specific role, *e. g. ,* GO issuing restricted to the regulator BO. Finally, the marketplace endorsement policy is specified in the endorsement policy configuration file (EPCF), which describes the BOs that have to endorse a transaction for it to be valid. The EPCF is stored in the ledger, and all transactions are executed according to the endorsement policy defined in it. Thus, in the case of a non-agreed change in the configuration by any of the BOs, MC does not allow endorsement of transactions from the BO-violator.

System Evaluation: The evaluation aims to measure the performance of marketplace energy trade transaction execution. For evaluation purposes, we developed the SC tailored to the energy marketplace needs. This study considers transaction throughput and latency as performance metrics. The *throughput* is the number of successful transactions (TPS) or reads (RPS) executed per second. The *latency* is the time it takes to finalize transaction

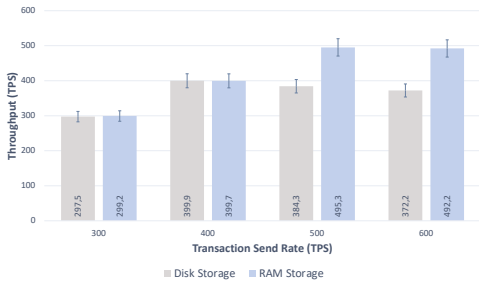


(a) *Write throughput. Block Size = 500Tx.*

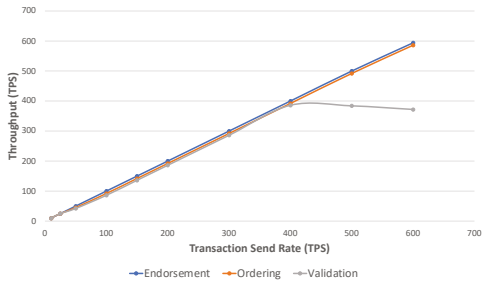


(b) *Read throughput. Query Size = 4KBytes.*

Figure 4.5: *HF-based energy marketplace throughput and latency.*



(a) *Transaction throughput depending on disk type.*



(b) *Validation step scalability.*

Figure 4.6: *HF-based energy marketplace bottleneck investigation.*

execution and write it to the ledger or return a reply with the query result. Further, a number of configurable metrics were manipulated within HF to investigate the throughput and latency. The first configurable metric is the HF *block configuration*, *e. g.*, block size or timeout. Further, we investigate the influence of different storage types on transaction throughput, *e. g.*, Disk or RAM. Finally, we investigate the scalability of orderer nodes and endorse-order-validate transaction life cycle steps.

The performance evaluation was executed on the SC's sell electricity function, *c. f.*, Algorithm 8, since it has the highest computational load. The successful sell electricity function execution corresponds to one endorse-order-validate transaction life-cycle. The performance evaluation result of write ledger transactions are described in Figure 4.5a. The sell energy transaction is private for two trading EPs and the regulator. The maximum throughput of approximately 400 TPS is achieved with a block size of 500 transactions and a timeout of 1 second. To read from the ledger, the system does not need to execute the endorse-order-validate life cycle. Instead, the data is read from the local peer database. The read transaction throughput result is shown in Figure 4.5b. The query was constructed to read 4KB of data, *i. e.*, size of the energy trade transaction. The results demonstrate a maximum throughput of approximately 1412 RPS.

In order to identify the ledger write throughput bottleneck, the implemented solution has been tested with a RAM-based disk. In this approach, the HF docker containers were stored

in the RAM disk only. The usage of RAM disk increased the maximum write throughput by 25% (from 400 to 500), *c. f.,* Figure 4.6a. Such behavior leads to the conclusion that the disk input operations per second limit the performance of the HF-based marketplace. However, such limitation is dependent on the hardware setup, which the marketplace BOs can regulate.

Further, the scalability of the HF-based marketplace was investigated with varying numbers of orderer nodes. In the main experimental setup, there are three orderer nodes. The decrease to a single orderer node does not demonstrate any write throughput increase, indicating that the RAFT consensus execution is not the performance bottleneck. Furthermore, the increase to 10 orderer nodes demonstrates HF's orderers' ability to scale and does not affect write throughput significantly, *i. e.,* remains at approximately 400 TPS. Finally, to investigate the impact of different transaction life-cycle phases, *i. e.,* endorse-order-validate, each phase throughput was measured individually, *c. f.,* Figure 4.6b. The results indicate that the *validate* phase represents the main and major performance bottleneck of the entire transaction life-cycle, indicating queue buildup in the validation phase, which limits the maximum transaction throughput.

We define a formula to calculate an estimated number of prosumers for the maximum measured throughput of 400 TPS, *c. f.,* Eq.(8.1). The T_{max} parameter is the system's maximum throughput, *i. e.,* 400 TPS in our case. Further, the m, o, t, and g parameters correspond to daily metering device updates, order creation, trade transaction execution, and GO issuing. Dividing the T_{max} by the sum of m, o, t, g yields an approximate number of prosumers that are able to operate within a marketplace. The throughput of 400 TPS allows the execution of 34 560 000 transactions per 24 hours. Further, setting the m affects the rest of the parameters. If the m parameter is set to 24, *i. e.,* hourly metering device update, it gives the prosumer the possibility to trade generated electricity 24 times during the day. Hence, the worst-case scenario is that the prosumer trades electricity every time the metering device update happens. For each trade to be executed, the order must be placed and GO issued. In such a case, the final number of daily prosumer transactions equals 96 (24+24+24+24). Thus, 34 560 000 / 96 = 360 000 prosumers.

Results and Conclusions: The designed and evaluated system demonstrates that the HF-based energy marketplace enables sufficient guarantees for trade execution while preserving EPs' and prosumers' data privacy. The performance evaluation shows that HF can support a maximum throughput of 400 TPS. However, maximum throughput depends on the system design decision, *e. g.,* block configuration and nodes hardware. Further, the designed marketplace supports 360 000 prosumers, which meets the requirements of medium-sized communities and confirms our hypothesis. The validation phase requires a considerable amount of computational capabilities creating a performance bottleneck and limiting throughput at 400 TPS for our marketplace design. Thus, optimization is required to increase the throughput and scalability of HF overall. Further, since the HF is limited in throughput, the ledger only has to save the data, which is critically needed to establish digital trust between system actors. In the case of the P2P energy marketplace, it is the GO and prosumer wallet records. Finally, the regulator's involvement in the energy trade settlement process results in a partial centralization around the governmental authority.

4.4 (Paper IV) On the Performance and Scalability of Consensus Mechanisms in Privacy-Enabled Decentralized Renewable Energy Marketplace

(This paper is an extension of our study described in [99], and provides extended discussions on system architecture, implementation, performance evaluation results, and a summary.)

Problem Statement: CFT consensus used in HF's orderer nodes does not provide any protection from malicious nodes. In contrast, BFT consensus provides the same level of protection as CFT and, in addition, can operate in the presence of adversaries, *e. g.*, nodes that manipulate transactions and try to disrupt the blockchain network operation. The Hyperledger Besu (HB) [31] blockchain platform supports BFT consensus and acts as a trust-enabling mechanism that guarantees that P2P energy trade settlement conditions are followed while maintaining EPs' and prosumers' data privacy.

Research Question: *What are the performance and scalability characteristics of Hyperledger Besu-based renewable energy marketplace with trust, privacy, and regulatory requirements? (RQ3.2)*

Use Case Design: The use case requirements were defined in the context of **Paper III**. The requirements towards P2P energy trade settlement and GO issuing and consumption are aligned with D2018/2001. The BFT consensus mechanism's ability to operate in the presence of adversaries is a compelling feature that may help to reduce the partial centralization of CFT, *i. e.*, reliance on a trusted organization within a blockchain network. However, the increased computations of BFT may come at the cost of decreased performance in comparison to CFT. Thus, this paper designs the HB-based P2P energy marketplace to investigate the performance and scalability characteristics of trade settlement execution with BFT consensus mechanism. Further, we identify the main hypothesis of this study:

Hypothesis: *Hyperledger Besu-based energy marketplace enables trusted energy trade execution and prosumer data privacy via BFT consensus while proving sufficient throughput for a medium-sized energy community,* i. e., *up to 250 000 prosumers.*

System Design: *Hyperledger Besu* (HB) [31] is representative of a private permissioned blockchain platform based on an open-source Ethereum [100] client. HB implements the *Enterprise Ethereum Alliance Protocol* to enable such functionality as private transactions, IAM, and permissioning. In the HB network, the *validator* nodes order, execute and verify transactions in the blockchain network. All transactions in the HB network are initiated by *user accounts*, representing a public and private key pair that can be generated off-chain. The smart contract (SC) defines functions that a user account can call to operate on the data in the ledger. First, the SC has to be installed in the blockchain network. Once installed, it serves as a predefined trade settlement contract where fixed, agreed-upon rules are enforced during every execution. In HB, private data is stored in transactions disclosed only to a subset of network participants, *i. e.*, privacy group (PG), while the rest of the network does not have access to the contents. Further, the rest of the network does not know the list of nodes that belong to PG. The private transactions in HB are handled by the *Tessera* private transaction manager. As the Ethereum blockchain was not designed to work with private transactions, the *Tessera* private transaction manager was adopted in HB. It is built as a separate entity and complements the implementation of the Ethereum Enterprise Client.

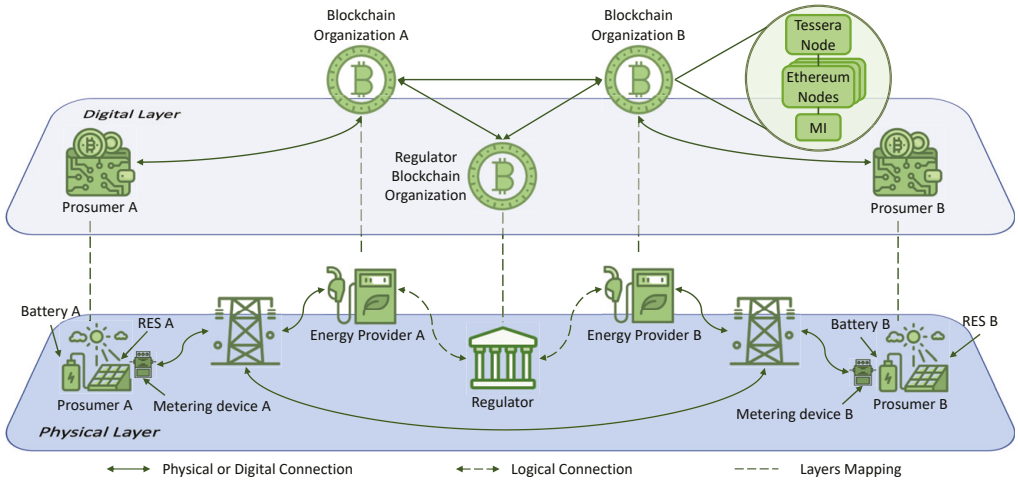


Figure 4.7: *P2P HB-based energy marketplace architecture.*

The consensus mechanisms supported by the HB are PoW (Ethash), Proof of Stake (PoS), and Proof of Authority (PoA) (Clique, IBFT 2.0, and QBFT). This study concentrates on PoA consensus mechanisms used in private HB networks. Within the available PoA consensus mechanisms, some are identified as Byzantine Fault Tolerant (BFT), *i. e.*, IBFT 2.0 and QBFT, and/or Crash Fault Tolerant (CFT), *i. e.*, Clique. CFT consensus mechanisms are protected only from node failure, *i. e.*, if less than 50% of the nodes fail, the network can operate successfully. BFT consensus provides the same level of protection as CFT and in addition can operate in the presence of adversaries, *e. g.*, nodes that manipulate transactions and try to disrupt the blockchain network operation. However, there are limitations to the BFT consensus mechanisms in terms of the number of adversaries, *i. e.*, consensus is jeopardized if more than *1/3* of the nodes collude. In practice, when the blockchain user account initiates the transaction, it must wait until the moment the *2m+1* responses are received, where m is the maximum number of allowed failed or malicious nodes. When *2m+1* responses are successfully received, the consensus is achieved and the state of the network is updated. The improved security of BFT consensus mechanisms may come at the cost of decreased performance compared to CFT ones.

The HB-based P2P energy marketplace is depicted in Figure 4.7. Each EP and regulator are represented within the marketplace as a *blockchain organization* (BO). Each BO must operate at least one validator node. The validator nodes are the main guarantors of valid transaction execution and require the most computational power. Further, each BO has a dedicated *Tessera* node to enable private transaction execution. In addition, each BO has a marketplace interface (MI) that prosumers use to conduct P2P trade settlements. Finally, prosumers are represented as user accounts. Since they do not operate validators, they need to trust their EP's BO to execute transactions on their behalf.

Security Analysis: HB enjoys similar transaction execution guarantees and fulfills the requirements defined in security analysis conducted in **Paper III**. Further, due to BFT consensus, there is no need to rely on a trusted party to run the ordering process.

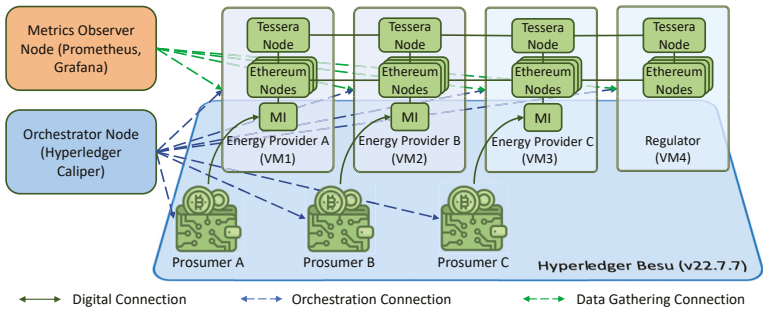
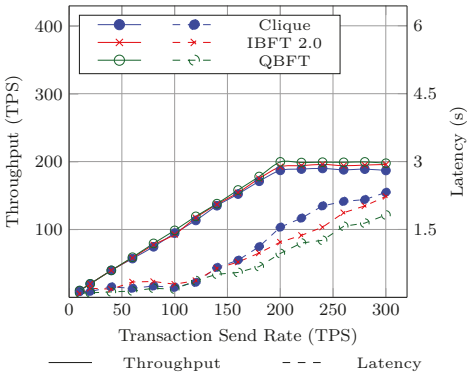


Figure 4.8: *Implemented HB-based energy marketplace.*

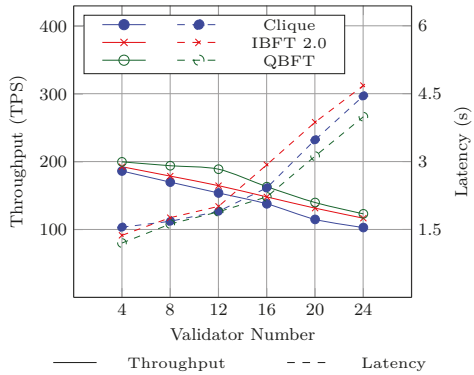
However, the non-native private transaction handling with Tessera comes at the cost of their auditability. The Tessera nodes distribute private transactions to the members of PG. However, the rest of the nodes outside of PG only receive the record, *i. e.*, the proof, confirming that the private transaction was executed. Such an approach results in a limitation where the blockchain network participants outside of PG cannot verify the validity of the private transaction data. This is a result of the inability of non-PG members of the HB network to verify the correctness of private SC deployment and transaction execution.

System Evaluation: The main aim of this study is to measure the performance of private transaction execution HB's BFT consensus with the SC tailored to the energy marketplace needs. The implemented test infrastructure is shown in Figure 4.8. Several performance metrics are considered in this study. The *throughput* is the number of successful transactions (TPS) or reads (RPS) executed per second in the blockchain network. The *latency* is the time it takes to finalize transaction execution and write it to the ledger or return a reply with the query result. The *scalability* is the behavior of the network with an increasing number of nodes involved in consensus execution. Further, a number of configurable metrics were manipulated within HB to investigate the throughput, latency, and scalability. The *Block Period Seconds* (BPS) parameter defines the time validators accept transactions to add to the new block. When the BPS time is up, the block is cut and embedded into the ledger. Further, scalability is investigated by changing the number of validator nodes and PG size.

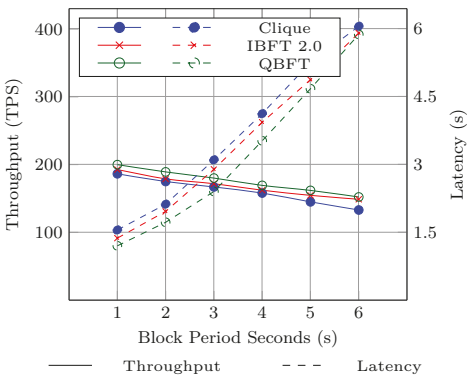
The read results demonstrate a maximum throughput of 1440 RPS a similar result to the HF-based marketplace investigated in **Paper III**. Further, a sell electricity function, *c. f.*, Algorithm 11, was executed to test maximum write TPS due to having the highest computational complexity. To write a transaction to the ledger, a respective consensus mechanism, *i. e.*, Clique, IBFT 2.0, or QBFT, must be executed. First, we test the baseline HB configuration, which included the minimum necessary setup to operate, *i. e.*, four validators, BPS = 1s. The PG size is 3, *i. e.*, two EPs and the regulator. The write throughput measurement results are shown in Figure 4.9a. All consensus mechanisms show a similar performance of approximately 200 TPS. The baseline test demonstrates the maximum sustainable network load with private transactions of around 200 TPS. Thus, further tests are conducted with a fixed send rate of 200 TPS. Next, the maximum TPS with a varying BPS was investigated, *c. f.*, Figure 4.9c. The results demonstrate that the BPS affects the maximum throughput of the HB network, *i. e.*, the BPS increase results in a steady



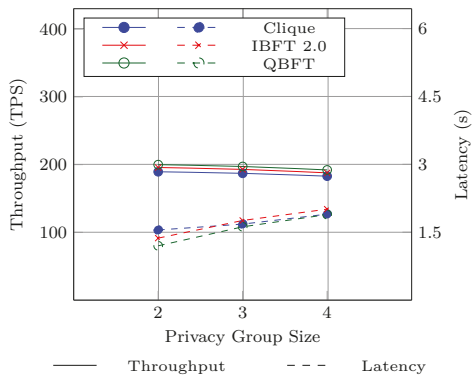
(a) *Transaction Throughput and Latency.*



(b) *Varying number of* Validator Nodes.



(c) *Varying* Block Period Seconds.



(d) *Varying* PG Size.

Figure 4.9: *Hyperledger Besu performance and scalability evaluation results.*

throughput decrease. Further, the latency rises significantly, *e. g.* , up to approximately 6s latency for BPS = 6s. Here all investigated consensus mechanisms show similar performance under varying BPS, where QBFT is the best performer.

The results of the validator scalability investigation are shown in Figure 4.9b. Here, the number of validator nodes was changed from 4 to 24 with a step of 4. Results demonstrate that the number of validator nodes significantly affects the maximum network throughput. It represents a significant performance bottleneck, resulting in approximately 42% throughput reduction with 24 validators. Here, all investigated consensus mechanisms demonstrate similar performance, with QBFT having the highest TPS and the lowest latency. In addition, QBFT demonstrates the best scalability by maintaining 190-200 TPS up to 12 validators. The results of PG size scalability are shown in Figure 4.9d. The investigated PG sizes are under four nodes because each BO can operate only one Tessera node, *i. e.* , test infrastructure limitation. The PG size increase does not result in a significant throughput decrease.

The measured HB throughput of 200 TPS can be used to estimate the maximum number

of prosumers the energy marketplace can support, Eq.(9.1). The same metrics as in **Paper III** are used to estimate this marketplace model. The only difference is that as trade settlement is executed in two stages, *i. e.,* HB cannot modify private and public blockchain data in the same transaction, it's every execution requires two blockchain transactions. The throughput of 200 TPS implies that the energy marketplace can execute 17 280 000 transactions in 24 hours. In the previously defined scenario in **Paper III**, *i. e.,* prosumers trade their generated electricity 24 times a day, the total number of daily prosumer transactions equals 120 (24+24+24*2+24). Therefore, the maximum number of prosumers supported by the marketplace is 17 280 000 / 120 = 144 000, corresponding to a small-to-medium-size energy community.

Results and Conclusions: The designed and evaluated system demonstrates that the HB-based energy marketplace enables sufficient guarantees for trade execution while preserving EPs' and prosumers' data privacy. Further, performance evaluation results of an HB-based marketplace private transaction execution with three main PoA consensus mechanisms, *i. e.,* Clique, IBFT 2.0, and QBFT, demonstrate a throughput of approximately 200 TPS with baseline configuration. Thus, the designed marketplace supports only 144 000 prosumers, which does not meet the requirements of medium-sized communities and falsifies our hypothesis. Further, HB's best-performing QBFT consensus mechanism demonstrates lower throughput than the HF blockchain platform. This is a side effect of BFT and, thus, increased computations of QBFT. In contrast, HF executes the RAFT consensus mechanism, which is CFT, *i. e.,* more centralized and vulnerable to collusion between malicious nodes. However, the inherent centralization around the regulator mitigates this issue, making HF better suited for such a use case.

4.5 (Paper V) On the Application of Enterprise Blockchains in Decentralized Renewable Energy Marketplaces

Problem Statement: The studies in **Papers III** and **IV** defined HF and HB-based decentralized energy marketplaces compliant with D2018/2001. However, while the compliance led to potential real-world applicability, it resulted in a partial centralization of the marketplace around the regulator. Considering this limitation, changes in marketplace design and D2018/2001 are needed to improve the flexibility and scalability of the decentralized blockchain-based energy trade.

Research Question: *How can current regulations and blockchain-based systems be improved to provide enhanced security and privacy for renewable energy marketplaces? (RQ4)*

Use Case Design: Papers III and **IV** elaborated on the differences in HF and HB consensus mechanisms. Further, the HF demonstrated higher throughput in comparison to HB. However, HF supports only CFT consensus, which is not protected from malicious nodes in the blockchain network. Thus, HF may benefit from BFT consensus in its ordering process, protecting it from malicious nodes. BFT consensus protects the ordering process, eliminating the need for a trusted party to run the orderer nodes. Consequently, it opens opportunities for changes in marketplace design and possible amendments for D2018/2001. Thus, we identify the main hypothesis of this study:

Hypothesis: *The use of the BFT consensus mechanism enables assigning the regulator role to the EP and, thus, improves trust and privacy characteristics of the Hyperledger Fabric-based marketplace.*

System Design: To address the limitation of CFT consensus, we propose an improvement towards the HF ordering process by utilizing the BFT-SMART (BFTS) [101] consensus mechanism. BFTS implements the Practical BFT (PBFT), following the three-phase commit process, *i. e.,* pre-prepare, prepare, and commit. A new block is disseminated to all orderers with a *pre-prepare* message. Then, orderers broadcast *prepare* message. When receiving *prepared* replies from *2/3+1* of orderers, the leader orderer broadcasts *commit message*. When *commit* message is received by *2/3+1* of orderers, the new block is written to the ledger. BFTS protects the HF's ordering process from malicious nodes, *i. e.,* up to 1/3 of all orderers.

The BFT consensus enables enhancements to D2018/2001 legislation. We propose several improvements that aim to improve marketplace flexibility and data privacy. *1.)* The minimum quantity of renewable energy that GO can be issued for is 1kWh. This should enable prosumers to trade small amounts of electricity and bring flexibility into the marketplace ordering system. *2.)* To become a part of the marketplace, the EP has to be authorized by the National Regulatory Authority (NRA) to become a regulator as well as to issue and consume GOs. In this way, no external trusted party participates in the energy trade process, *c. f.,* the regulator in **Paper III**. Further, such an improvement preserves prosumer's and EP's data privacy and enables P2P energy trade settlement.

The architecture of the energy marketplace is depicted in Figure 4.10. Here, each EP acts as a separate BO that is an NRA-authorized regulator. Each BO operates a number of peer nodes, which are the primary guarantors of valid transaction execution. Further, since EPs are now regulators, they must execute the transaction ordering process. Thus, every EP

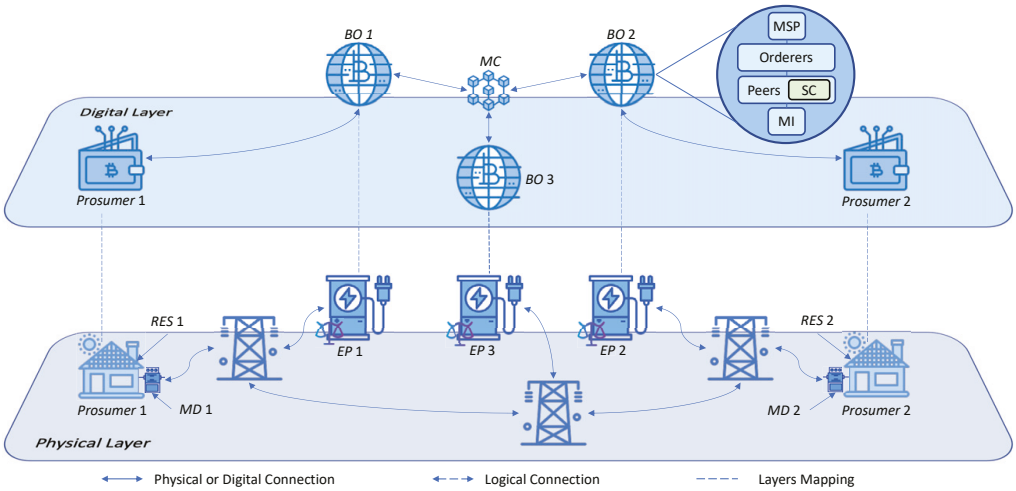


Figure 4.10: *P2P HF-based energy marketplace architecture.*

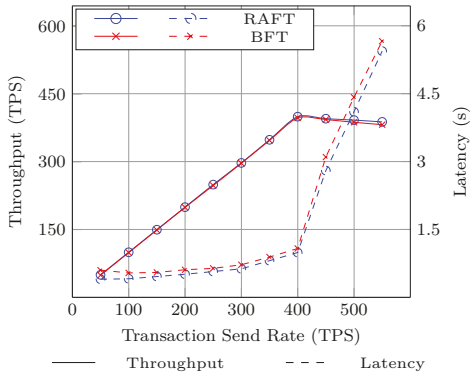
operating within a marketplace must run at least one orderer node to ensure the integrity and auditability of the ordering process. Further, prosumers do not act as BO, *i. e.*, have no peers, relying on the EPs to endorse and validate trade settlement transactions on their behalf. Also, prosumers' EP certifies their RES and metering devices.

Security Analysis: The new BFTS-based energy marketplace architecture enjoys the same execution guarantees and fulfills the requirements defined in the security analysis conducted in **Paper III**. Further, due to BFT consensus, no external trusted party participates in the energy trade process. This allows for avoiding marketplace partial centralization and preserves prosumers' and EPs' data privacy.

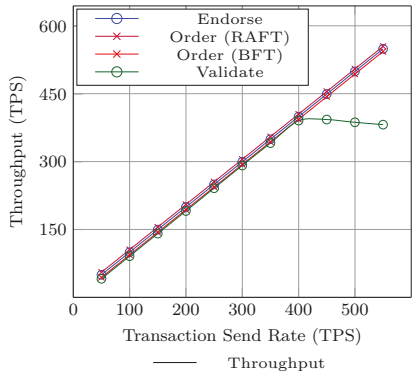
System Evaluation: Algorithm 14 was executed as an SC function to test maximum write TPS. We test the baseline HF configuration with BFTS, which includes the minimum necessary infrastructure setup to operate, *i. e.*, 4 VMs with four peers and four orderers. Further, we execute the same experiments with RAFT consensus. The throughput measurement results are shown in Figure 4.11a. The BFTS consensus demonstrates a similar maximum throughput as CFT RAFT, *i. e.*, approximately 400 TPS. Further, BFTS and RAFT consensuses demonstrate similar latency, *i. e.*, up to 1s. Any send rate higher than 400 TPS significantly increases latency for both BFTS and RAFT.

The results of the endorse-order-validate transaction life-cycle stages evaluation demonstrate that the *validate* stage is the main bottleneck and limits the resulting throughput to 400 TPS, *c. f.*, Figure 4.11b. The validate stage requires the highest amount of computations and executes each transaction's validation sequentially, resulting in poor process scalability.

The horizontal scalability was investigated with a varying number of orderer nodes and BOs, *i. e.*, one BO corresponds to one peer and one orderer node. The results of orderer scalability are shown in Figure 4.12a. The RAFT consensus demonstrates better scalability than BFTS, showing minimal-to-no throughput loss at 16 orderer nodes. Further, the decrease in throughput of BFTS consensus is a result of the three-phase commit process,

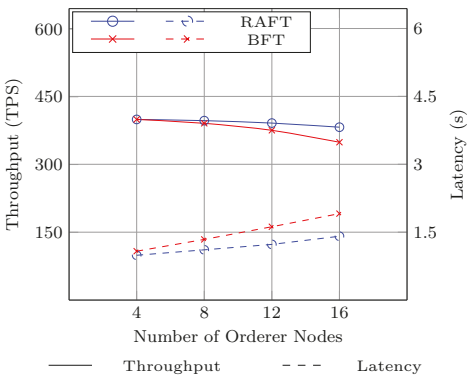


(a) *Transaction throughput and latency with varying send rate.*

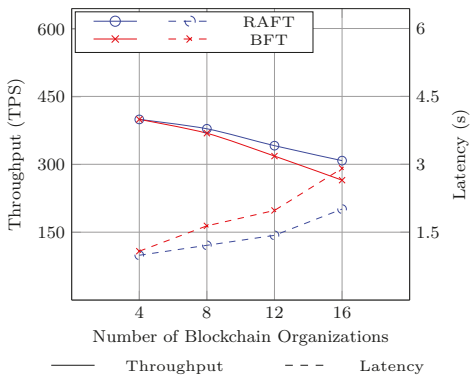


(b) *Throughput of all HF transaction lifecycle stages with varying send rate.*

Figure 4.11: *Hyperledger Fabric performance and scalability evaluation results.*



(a) *Throughput and Latency for varying number of Orderer Nodes.*



(b) *Throughput and Latency for a varying number of BOs.*

Figure 4.12: *Hyperledger Fabric scalability evaluation results.*

which requires more computations and information exchange between orderers to provide protection against malicious nodes. The results of BO scalability are shown in Figure 4.12b. Both consensus mechanisms demonstrate a significant throughput drop due to the increasing number of peer nodes for every BO. RAFT demonstrates approximately 25% throughput drop when scaling the BO number from 4 to 16, while BFTS shows 33% throughput loss under the same conditions, *i. e.* , a number of peer and order nodes.

Results and Conclusions: The results demonstrate that the new marketplace design confirms our hypothesis as BFT consensus enables assigning the regulator role to an EP. Further, the new design enables the P2P energy trade automation while providing better privacy-preserving capabilities than the HF-based marketplace design described in **Paper III**, *i. e.* , no external trusted party participates in the trade process. Next, the BFT consensus

protects the blockchain network from malicious nodes and demonstrates a similar maximum throughput as CFT RAFT, *i. e.,* approximately 400 TPS. Thus, the throughput of BFT consensus still fulfills the requirements of a medium-sized energy community and supports up to 360 000 prosumers. However, assigning the role of the regulator to the EP may affect the level of governmental control over the marketplace and energy trade within the national energy distribution system. Such a change requires strong regulatory controls over the EPs who become regulators, as there is always a risk of them colluding in order to manipulate the market. Thus, auditing combined with significant legal penalties should be established to efficiently deter market manipulation.

Finally, this study defines a new multilayered marketplace model to address scalability requirements, *c. f.,* Figure 10.7. In the multilayered model, different energy communities run their own blockchains and are interconnected through layer *interface*, *i. e.,* a marketplace actor who can communicate with underlying and overlying layers. Every layer corresponds to an expanding geographical and economic unit, *e. g.,* energy community, city, or region, with the aim to enable cross-country energy trade.

Chapter Five

Conclusions and Future Work

This thesis describes the work on the *advancement of the knowledge on the efficient design and evaluation of distributed marketplaces with an emphasis on trust and privacy.* The marketplace design is investigated as a complete system, taking into consideration application, security, performance, and regulatory characteristics. The main innovation of investigated designs is the combination of these characteristics to make the resulting system applicable in the real world. This thesis includes five research papers discussed as a continuous investigation where every paper builds on the results of prior studies. Furthermore, the included studies lead to conclusions on the applicability of the decentralized governance model in distributed digital marketplaces.

Marketplaces enable the rapid development and expansion of their respective domain, *e. g. ,* AI engineering or renewable energy trading. In particular, the centralized marketplace model enables privacy and trusted execution within a centrally governed, restricted environment controlled by the TTP. In contrast, blockchain technology enables decentralized system privacy and trust mechanisms deployment and enforcement. To this end, this thesis identifies and addresses two research objectives.

The first objective is *to design and evaluate the centralized digital marketplace with an emphasis on digital trust and data privacy.* This objective is addressed in the **Paper I**, where the application of the centralized marketplace model is investigated for the use case of collaborative AI engineering. The centralized marketplace model demonstrates that a single governing organization provides necessary guarantees in terms of data privacy and IPR preservation. Further, the developed SVP enables collaborative AI engineering where organizations rely on the computing infrastructure deployed and managed by the TTP. In this case, the TTP acts as the sole provider of digital trust and data privacy, bringing the organizations together for collaboration and accelerated development of efficient AI solutions' training and benchmarking. The SVP evaluation demonstrated that the designed security mechanisms provide a controlled execution environment where AI assets are protected from exfiltration.

However, centralization may limit the rapid expansion of the marketplace in its respective domain due to restricted process interoperability and automation. Furthermore, being an intermediary in a business settlement, the governor can dictate rules and trade transaction payment distribution. This may lead to a value distribution imbalance between the marketplace and sellers. Thus, our investigations shift towards the decentralized marketplace model where governance is distributed among multiple organizations. Decentralized governance enhances marketplace scalability and brings interoperability to the services of collaborating governor organizations. Further, driven by business interests, governing organizations create transparent market rules, ensuring accountability and compliance within a decentralized marketplace.

The second research objective is *to design and evaluate the decentralized digital marketplace based on blockchain technology with an emphasis on digital trust and data privacy.* This

objective is addressed in **Papers II**, **III**, **IV**, and **V**. The investigation started with a survey of blockchain-based marketplaces' state-of-the-art and identifying research gaps in **Paper II**. The survey demonstrates the increasing number of blockchain-based marketplace proposals and standardization activities. However, the survey also shows that blockchain technology is still in its infancy, requiring in-depth investigations of particular marketplace use cases and their services, *e. g.*, IAM, assurance, or trade settlement.

The use case of a decentralized blockchain-based P2P renewable energy marketplace was investigated in **Papers III**, **IV**, and **V**. The energy marketplace is working with the electricity distribution system, which is highly regulated by the government due to national security concerns. Thus, we define marketplace actors and requirements in accordance with Directive 2018/2001 of the European Parliament in order to comply with regulations. Eventually, the ultimate goal of RES promotion and widespread adoption is the decarbonization of the atmosphere. The energy marketplace, as a platform, should attract prosumers into installing RES due to the opportunity to become energy-independent and prosper from generated energy selling.

Paper III defines a decentralized P2P energy marketplace that uses private permissioned blockchain Hyperledger Fabric (HF) and its smart contracts (SCs) to automate electricity trade settlement execution. The HF runs a CFT consensus mechanism, which protects the blockchain network from failed nodes. The marketplace system evaluation demonstrates that blockchain builds digital trust between governing organizations while allowing private data processing. Further, the maximum system throughput allows the marketplace to support approximately 370 000 concurrent users, which confirms our hypothesis. However, compliance with D2018/2001 resulted in partial marketplace centralization around the governmental regulator.

Paper IV defines a decentralized P2P energy marketplace that uses private permissioned blockchain Hyperledger Besu (HB). The main aim of this study is to conduct the performance and scalability evaluation of HB's BFT consensus mechanism, which protects the decentralized marketplace from malicious nodes. The marketplace system evaluation demonstrates that blockchain builds digital trust between governing organizations while allowing private data processing. However, the performance evaluation results demonstrate that improved security of BFT consensus comes at the cost of decreased performance compared to CFT ones. The maximum system throughput allows the marketplace to support approximately 150 000 concurrent users, which falsifies our hypothesis. Further, compliance with D2018/2001 resulted in partial marketplace centralization around the governmental regulator.

In **Paper V**, we aimed to address the problem of partial centralization around the regulator actor. Thus, we propose a new model of the blockchain-based P2P energy marketplace with increased flexibility and scalability. The marketplace utilizes HF due to its superior throughput and private data handling capabilities in comparison to HB. Based on previous studies conducted in **Papers III** and **IV**, we propose an improvement towards HF security by utilizing a BFT consensus. Consequently, to improve marketplace flexibility and data privacy, we propose enhancements to D2018/2001 legislation and the regulator actor: *1.)* The minimum quantity of traded energy is 1kWh, bringing flexibility into the marketplace ordering system. *2.)* To become a part of the marketplace, the EP has to become a governmentally certified regulator. In this way, we avoid partial centralization around external regulators.

Further, such an improvement preserves prosumer and EP's data privacy by eliminating the need to share private data with external regulators. Finally, this study proposes a new multilayered marketplace model to address scalability requirements. In the multilayered model, different energy communities run their own blockchains, and every layer corresponds to an expanding geographical and economic unit: energy community, city, region, or country. The performance evaluation results demonstrate that the BFT consensus applied in HF's endorse-order-validate transaction life cycle shows similar trade settlement throughput as CFT, enables D2018/2001 improvements, and satisfies the requirements of a medium-sized energy community.

The investigation of distributed systems demonstrated that blockchain technology is only rational if the use case defines multiple non-trusting organizations controlling parts of the infrastructure, *i. e.*, decentralized governance model. In contrast, if the distributed system is governed centrally, conventional technologies such as a relational database management system can be applied, considering its current advantage in performance. However, private permissioned blockchain technologies, *e. g.*, Hyperledger Fabric, are rapidly developing and demonstrate considerable improvements in transaction throughput. Thus, they may meet the performance of conventional databases in the future. In such a case, blockchain can be used even within centrally governed distributed systems as a distributed data storage, providing such benefits as provenance, transparency, and accountability to all processed data. Further, it enables blockchain-based services interoperability, *e. g.*, enabling self-sovereign IAM systems and cross-system business settlement.

As a future research direction, we plan to investigate the application of the developed decentralized marketplace design principles in other research domains. The telecommunication service marketplaces (TSMs) are of particular interest as they involve virtual services trade between customers and communication service providers. Such a process involves complex steps that are challenging to automate and require designing specific security controls ensuring trust, privacy, and service level agreement (SLA) enforcement. Further, the telecommunication services domain incorporates numerous business entities, which pose specific performance and scalability system characteristics. Finally, the consensus mechanism is fundamental for the secure, compliant, and trusted collaboration between parties in decentralized systems. Thus, considering the conducted evaluations of various consensus mechanisms, investigation on how to improve their performance and scalability is of interest.

References

[1] R.-V. Tkachuk. “Towards Decentralized Orchestration of Next-Generation Cloud Infrastructures”. In: *Licentiate Thesis*. Blekinge Institute of Technology, 2021, pp. 1–166.

[2] B. Idem. *Coexistence of Centralized and Decentralized Markets*. 2021. arXiv: 2111.12767.

[3] M. F. Mubarak and M. Petraite. “Industry 4.0 technologies, digital trust and technological orientation: What matters in open innovation?” In: *Technological Forecasting and Social Change* 161 (2020), pp. 1–11. DOI: 10.1016/j.techfore.2020.120332.

[4] V. Koutsos, D. Papadopoulos, D. Chatzopoulos, S. Tarkoma, and P. Hui. “Agora: A Privacy-Aware Data Marketplace”. In: *IEEE Transactions on Dependable and Secure Computing* 19.6 (2022), pp. 3728–3740. DOI: 10.1109/TDSC.2021.3105099.

[5] O. Avellaneda, A. Bachmann, A. Barbir, J. Brenan, P. Dingle, K. H. Duffy, E. Maler, D. Reed, and M. Sporny. “Decentralized Identity: Where Did It Come From and Where Is It Going?” In: *IEEE Communications Standards Magazine* 3.4 (2019), pp. 10–13. DOI: 10.1109/MCOMSTD.2019.9031542.

[6] M. Jovanovic, D. Sjödin, and V. Parida. “Co-evolution of platform architecture, platform services, and platform governance: Expanding the platform value of industrial digital platforms”. In: *Technovation* 118 (2022), pp. 1–14. DOI: 10.1016/j.technovation.2020.102218.

[7] A. Zutshi, A. Grilo, and T. Nodehi. “The value proposition of blockchain technologies and its impact on Digital Platforms”. In: *Computers & Industrial Engineering* 155 (2021), pp. 1–17. DOI: 10.1016/j.cie.2021.107187.

[8] C. Janiesch, P. Zschech, and K. Heinrich. “Machine learning and deep learning”. In: *Electronic Markets* 31 (2021), pp. 685–695. DOI: 10.1007/s12525-021-00475-2.

[9] C. Zhang and Y. Lu. “Study on artificial intelligence: The state of the art and future prospects”. In: *Journal of Industrial Information Integration* 23 (2021), pp. 1–9. DOI: 10.1016/j.jii.2021.100224.

[10] M. Dabić, J. Maley, L.-P. Dana, I. Novak, M. M. Pellegrini, and A. Caputo. “Pathways of SME internationalization: a bibliometric and systematic review”. In: *Small Business Economics* 55 (2020), pp. 705–725. DOI: 10.1007/s11187-019-00181-6.

[11] T. Llewellynn, M. M. Fernández-Carrobles, O. Deniz, S. Fricker, A. Storkey, N. Pazos, G. Velikic, K. Leufgen, R. Dahyot, S. Koller, G. Goumas, P. Leitner, G. Dasika, L. Wang, and K. Tutschku. “BONSEYES: Platform for Open Development of Systems of Artificial Intelligence”. In: *Proceedings of ACM International Conference on Computing Frontiers*. Siena, Italy, 2017. DOI: 10.1145/3075564.3076259.

[12] J. Singh and J. D. Michels. “Blockchain as a Service (BaaS): Providers and Trust”. In: *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 2018, pp. 67–74. DOI: 10.1109/EuroSPW.2018.00015.

[13] R.-V. Tkachuk, D. Ilie, K. Tutschku, and R. Robert. "A Survey on Blockchain-Based Telecommunication Services Marketplaces". In: *IEEE Transactions on Network and Service Management* 19.1 (2022), pp. 228–255. DOI: 10.1109/TNSM.2021.3123680.

[14] A. Qazi, F. Hussain, N. A. Rahim, G. Hardaker, D. Alghazzawi, K. Shaban, and K. Haruna. "Towards Sustainable Energy: A Systematic Review of Renewable Energy Sources, Technologies, and Public Opinions". In: *IEEE Access* 7 (2019), pp. 63837–63851. DOI: 10.1109/ACCESS.2019.2906402.

[15] Y. Yang, S. Zhang, and Y. Xiao. "Optimal design of distributed energy resource systems coupled with energy distribution networks". In: *Energy* 85 (2015), pp. 433–448. DOI: 10.1016/j.energy.2015.03.101.

[16] B. Jasim and P. Taheri. "An Origami-Based Portable Solar Panel System". In: *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. 2018, pp. 199–203. DOI: 10.1109/IEMCON.2018.8614997.

[17] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. "Blockchain Based Decentralized Management of Demand Response Programs in Smart Energy Grids". In: *Sensors* 18.2 (2018), pp. 1–21. DOI: 10.3390/s18010162.

[18] Á. Hamburger. "Is guarantee of origin really an effective energy policy tool in Europe? A critical approach". In: *Society and Economy* 41 (2019), pp. 487–507. DOI: 10.1556/204.2019.41.4.6.

[19] EU Parliament. *Directives Directive (EU) 2018/2001 of the European Parliament*. 2022, pp. 82–209. URL: http://data.europa.eu/eli/dir/2018/2001/2022-06-07 (visited on 06/18/2023).

[20] B. Hertz-Shargel, D. Livingston, and A. C. of the United States. *Assessing Blockchain's future in transactive energy*. 2019. ISBN: 9781619775992. URL: https://www.atlanticcouncil.org/in-depth-research-reports/report/assessing-blockchains-future-in-transactive-energy/ (visited on 06/18/2023).

[21] T. Kollmann, S. Hensellek, K. de Cruppe, and A. Sirges. "Toward a renaissance of cooperatives fostered by Blockchain on electronic marketplaces: a theory-driven case study approach". In: *Electronic Markets* 30.2 (2020), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

[22] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. URL: https://bitcoin.org/bitcoin.pdf (visited on 06/18/2023).

[23] W. Zou, D. Lo, P. S. Kochhar, X.-B. D. Le, X. Xia, Y. Feng, Z. Chen, and B. Xu. "Smart Contract Development: Challenges and Opportunities". In: *IEEE Transactions on Software Engineering* 47.10 (2021), pp. 2084–2106. DOI: 10.1109/TSE.2019.2942301.

[24] Q. Zhou, H. Huang, Z. Zheng, and J. Bian. "Solutions to Scalability of Blockchain: A Survey". In: *IEEE Access* 8 (2020), pp. 16440–16455. DOI: 10.1109/ACCESS.2020.2967218.

[25] EU Parliament. *Regulation (EU) 2016/679 of the European Parliament (General Data Protection Regulation)*. 2016, pp. 1–99. URL: https://gdpr-info.eu/ (visited on 06/18/2023).

[26] A. Upadhyay, S. Mukhuty, V. Kumar, and Y. Kazancoglu. "Blockchain technology and the circular economy: Implications for sustainability and social responsibility". In: *Journal of Cleaner Production* 293 (2021), pp. 1–7. DOI: 10.1016/j.jclepro.2021.126130.

[27] A. Chakraborty, A. Mondai, and A. Srivastava. "Hardware-Assisted Intellectual Property Protection of Deep Learning Models". In: *2020 57th ACM/IEEE Design Automation Conference (DAC)*. 2020. DOI: 10.5555/3437539.3437711.

[28] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick. "Hyperledger Fabric". In: *Proceedings of the Thirteenth EuroSys Conference*. 2018, pp. 1–15. DOI: 10.1145/3190508.3190538.

[29] B. Podgorelec, V. Keršič, and M. Turkanović. "Analysis of Fault Tolerance in Permissioned Blockchain Networks". In: *2019 XXVII International Conference on Information, Communication and Automation Technologies (ICAT)*. 2019, pp. 1–6. DOI: 10.1109/ICAT47117.2019.8938836.

[30] T. Distler. "Byzantine Fault-Tolerant State-Machine Replication from a Systems Perspective". In: *ACM Compututing Surveys* 54.1 (2021). DOI: 10.1145/3436728.

[31] H. Foundation. *Hyperldger Besu Ethereum client*. 2022. URL: https://besu.hyperledger.org/en/stable/ (visited on 06/20/2023).

[32] D. Mourtzis, J. Angelopoulos, and N. Panopoulos. "A survey of digital B2B platforms and marketplaces for purchasing industrial product service systems: A conceptual framework". In: *8th CIRP Conference of Assembly Technology and Systems (CIRP)* 97 (2021), pp. 331–336. DOI: 10.1016/j.procir.2020.05.246.

[33] A. R. Sai, J. Buckley, B. Fitzgerald, and A. L. Gear. "Taxonomy of centralization in public blockchain systems: A systematic literature review". In: *Information Processing & Management* 58.4 (2021), pp. 1–35. DOI: 10.1016/j.ipm.2021.102584.

[34] L. Alawneh and A. Hamou-Lhadj. "Locating and categorizing inefficient communication patterns in HPC systems using inter-process communication traces". In: *Journal of Systems and Software* 194 (2022), pp. 1–21. DOI: 10.1016/j.jss.2022.111494.

[35] G. Fu, Y. Zhang, and G. Yu. "A Fair Comparison of Message Queuing Systems". In: *IEEE Access* 9 (2021), pp. 421–432. DOI: 10.1109/ACCESS.2020.3046503.

[36] J. Bachan, S. B. Baden, S. Hofmeyr, M. Jacquelin, A. Kamil, D. Bonachea, P. H. Hargrove, and H. Ahmed. "UPC++: A High-Performance Communication Framework for Asynchronous Computation". In: *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 2019, pp. 963–973. DOI: 10.1109/IPDPS.2019.00104.

[37] W. Tushar, T. K. Saha, C. Yuen, D. Smith, and H. V. Poor. "Peer-to-Peer Trading in Electricity Networks: An Overview". In: *IEEE Transactions on Smart Grid* 11.4 (2020), pp. 3185–3200. DOI: 10.1109/TSG.2020.2969657.

[38] W. Chen and I. Paik. "Toward Better Quality of Service Composition Based on a Global Social Service Network". In: *IEEE Transactions on Parallel and Distributed Systems* 26.5 (2015), pp. 1466–1476. DOI: 10.1109/TPDS.2014.2320748.

[39] K. Morris. *Infrastructure As Code: Managing Servers in the Cloud*. 2st. O'Reilly Media, Inc., 2020.

[40] N. Al-Zaben, M. M. Hassan Onik, J. Yang, N.-Y. Lee, and C.-S. Kim. "General Data Protection Regulation Complied Blockchain Architecture for Personally Identifiable Information Management". In: *2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*. 2018, pp. 77–82. DOI: 10.1109/iCCECOME.2018.8658586.

[41] G. Carl, G. Kesidis, R. Brooks, and S. Rai. "Denial-of-service attack-detection techniques". In: *IEEE Internet Computing* 10.1 (2006), pp. 82–89. DOI: 10.1109/MIC.2006.5.

[42] A. Patel, M. Taghavi, K. Bakhtiyari, and J. Celestino Júnior. "An intrusion detection and prevention system in cloud computing: A systematic review". In: *Journal of Network and Computer Applications* 36.1 (2013), pp. 25–41. DOI: 10.1016/j.jnca.2012.08.007.

[43] L. Alevizos, V. T. Ta, and M. H. Eiza. *Augmenting Zero Trust Architecture to Endpoints Using Blockchain: A Systematic Review*. 2021. arXiv: 2104.00460.

[44] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou. "A Survey of Distributed Consensus Protocols for Blockchain Networks". In: *IEEE Communications Surveys & Tutorials* 22.2 (2020), pp. 1432–1465. DOI: 10.1109/COMST.2020.2969706.

[45] R. M. Parizi, Amritraj, and A. Dehghantanha. "Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability and Security". In: *Lecture Notes in Computer Science*. Vol. 10974 LNCS. 2018, pp. 75–91. DOI: 10.1007/978-3-319-94478-4_6.

[46] CBAN. *Communications Business Automation Network Whitepaper Version 1.0*. URL: https://www.cban.net/resources (visited on 06/18/2023).

[47] P. Seltsikas and H. van der Heijden. "A Taxonomy of Government Approaches Towards Online Identity Management". In: *2010 43rd Hawaii International Conference on System Sciences*. 2010, pp. 1–8. DOI: 10.1109/HICSS.2010.38.

[48] A. G. Revar and M. D. Bhavsar. "Securing user authentication using single sign-on in Cloud Computing". In: *2011 Nirma University International Conference on Engineering*. 2011, pp. 1–4. DOI: 10.1109/NUiConE.2011.6153227.

[49] J. Bernal Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. Torres Moreno, and A. Skarmeta. "Privacy-Preserving Solutions for Blockchain: Review and Challenges". In: *IEEE Access* 7 (2019), pp. 164908–164940. DOI: 10.1109/ACCESS.2019.2950872.

[50] K. R. Rao, A. Nayak, I. G. Ray, Y. Rahulamathavan, and M. Rajarajan. "Role recommender-RBAC: Optimizing user-role assignments in RBAC". In: *Computer Communications* 166 (2021), pp. 140–153. DOI: 10.1016/j.comcom.2020.12.006.

[51] C. A. Alexander and L. Wang. "Cybersecurity, Information Assurance, and Big Data Based on Blockchain". In: *2019 SoutheastCon*. 2019, pp. 1–7. DOI: 10.1109/SoutheastCon42311.2019.9020582.

[52] A. Patil, A. Jha, M. M. Mulla, N. D.G., and S. Kengond. "Data Provenance Assurance for Cloud Storage Using Blockchain". In: *2020 International Conference on Advances in Computing, Communication & Materials (ICACCM)*. 2020, pp. 443–448. DOI: 10.1109/ICACCM50413.2020.9213032.

[53] A. Taha, A. Zakaria, D. Kim, and N. Suri. "Decentralized Runtime Monitoring Approach Relying on the Ethereum Blockchain Infrastructure". In: *2020 IEEE International Conference on Cloud Engineering (IC2E)*. 2020, pp. 134–143. DOI: 10.1109/IC2E48712.2020.00021.

[54] R. Angarita, A. Dejous, and P. Blake. "From centralized to decentralized blockchain-based product registration systems: the use case of lighting and appliances". In: *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2019, pp. 650–655. DOI: 10.1109/INFCOMW.2019.8845267.

[55] M. Liu, K. Wu, and J. J. Xu. "How Will Blockchain Technology Impact Auditing and Accounting: Permissionless versus Permissioned Blockchain". In: *Current Issues in Auditing* 13.2 (2019). DOI: 10.2308/ciia-52540.

[56] R. van Pelt, S. Jansen, D. Baars, and S. Overbeek. "Defining Blockchain Governance: A Framework for Analysis and Comparison". In: *Information Systems Management* (2021), pp. 21–41. DOI: 10.1080/10580530.2020.1720046.

[57] H. Cheng. "Research on the Distributed Photovoltaic Trading and Settlement Model Based on the Energy Blockchain". In: *2019 IEEE International Conference on Power Data Science (ICPDS)*. 2019, pp. 59–62. DOI: 10.1109/ICPDS47662.2019.9017201.

[58] C. Liu, Y. Xiao, V. Javangula, Q. Hu, S. Wang, and X. Cheng. "NormaChain: A Blockchain-Based Normalized Autonomous Transaction Settlement System for IoT-Based E-Commerce". In: *IEEE Internet of Things Journal* (2019), pp. 4680–4693. DOI: 10.1109/JIOT.2018.2877634.

[59] M. Nati et. al. *Federated CSPs Marketplace*. 2020. URL: https://www.tmforum.org/wp-content/uploads/2020/11/Federated_CSPs_Marketplace_Whitepaper_C20.0.34.pdf (visited on 06/18/2023).

[60] W. Park and K. K. Seo. "A study on cloud-based software marketing strategies using cloud marketplace". In: *Journal of Logistics, Informatics and Service Science* 7.2 (2020), pp. 1–13. DOI: 10.33168/JLISS.2020.0201.

[61] N. B. Somy, K. Kannan, V. Arya, S. Hans, A. Singh, P. Lohia, and S. Mehta. "Ownership Preserving AI Market Places Using Blockchain". In: *2019 IEEE International Conference on Blockchain (Blockchain)*. 2019, pp. 156–165. DOI: 10.1109/Blockchain.2019.00029.

[62] A. Kumar, B. Finley, T. Braud, S. Tarkoma, and P. Hui. "Sketching an AI Marketplace: Tech, Economic, and Regulatory Aspects". In: *IEEE Access* 9 (2021), pp. 13761–13774. DOI: 10.1109/ACCESS.2021.3050929.

[63] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi. "Personalized and Private Peer-to-Peer Machine Learning". In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. Vol. 84. 2018, pp. 473–481.

[64] H.-P. Cheng, P. Yu, H. Hu, F. Yan, S. Li, H. Li, and Y. Chen. *LEASGD: an Efficient and Privacy-Preserving Decentralized Algorithm for Distributed Learning*. 2018. arXiv: 1811.11124.

[65] J. Mello, J. Villar, R. J. Bessa, M. Lopes, J. Martins, and M. Pinto. "Power-to-Peer: a blockchain P2P post-delivery bilateral local energy market". In: *2020 17th International Conference on the European Energy Market (EEM)*. 2020, pp. 1–5. DOI: 10.1109/EEM49802.2020.9221901.

[66] R. Khalid, N. Javaid, A. Almogren, M. U. Javed, S. Javaid, and M. Zuair. "A Blockchain-Based Load Balancing in Decentralized Hybrid P2P Energy Trading Market in Smart Grid". In: *IEEE Access* 8 (2020), pp. 47047–47062. DOI: 10.1109/ACCESS.2020.2979051.

[67] F. Jamil, N. Iqbal, Imran, S. Ahmad, and D. Kim. "Peer-to-Peer Energy Trading Mechanism Based on Blockchain and Machine Learning for Sustainable Electrical Power Supply in Smart Grid". In: *IEEE Access* 9 (2021), pp. 39193–39217. DOI: 10.1109/ACCESS.2021.3060457.

[68] J. Abdella, Z. Tari, A. Anwar, A. Mahmood, and F. Han. "An Architecture and Performance Evaluation of Blockchain-Based Peer-to-Peer Energy Trading". In: *IEEE Transactions on Smart Grid* 12 (4 2021), pp. 3364–3378. DOI: 10.1109/TSG.2021.3056147.

[69] A. Esmat, M. de Vos, Y. Ghiassi-Farrokhfal, P. Palensky, and D. Epema. "A novel decentralized platform for peer-to-peer energy trading market with blockchain technology". In: *Applied Energy* 282 (2021), pp. 1–16. DOI: 10.1016/j.apenergy.2020.116123.

[70] Y. Wu, Y. Wu, H. Cimen, J. C. Vasquez, and J. M. Guerrero. "Towards collective energy Community: Potential roles of microgrid and blockchain to go beyond P2P energy trading". In: *Applied Energy* 314 (2022), pp. 1–16. DOI: 10.1016/j.apenergy.2022.119003.

[71] T. AlSkaif, J. L. Crespo-Vazquez, M. Sekuloski, G. van Leeuwen, and J. P. S. Catalao. "Blockchain-Based Fully Peer-to-Peer Energy Trading Strategies for Residential Energy Systems". In: *IEEE Transactions on Industrial Informatics* 18 (2022), pp. 231–241. DOI: 10.1109/TII.2021.3077008.

[72] S. Gawusu, X. Zhang, A. Ahmed, S. A. Jamatutu, E. D. Miensah, A. A. Amadu, and F. A. J. Osei. "Renewable energy sources from the perspective of blockchain integration: From theory to application". In: *Sustainable Energy Technologies and Assessments* 52 (2022), pp. 1–26. DOI: 10.1016/j.seta.2022.102108.

[73] T. Cioara, C. Pop, R. Zanc, I. Anghel, M. Antal, and I. Salomie. "Smart Grid Management Using Blockchain: Future Scenarios and Challenges". In: *2020 19th RoEduNet Conference: Networking in Education and Research (RoEduNet)*. 2020, pp. 1–5. DOI: 10.1109/RoEduNet51892.2020.9324874.

[74] U. Rai, G. Oluleye, and A. Hawkes. "An optimisation model to determine the capacity of a distributed energy resource to contract with a balancing services aggregator". In: *Applied Energy* 306 (2022), pp. 1–22. DOI: 10.1016/j.apenergy.2021.117984.

[75] J. G. Brookshear, D. Brylow, and S. Manasa. *Computer science: An overview*. Reading, MA, USA: Pearson (Addison-Wesley), 2015. ISBN: 978-1292061160.

[76] A. Fink. *The Survey Handbook*. Los Angeles, USA: SAGE Publications, Thousand Oaks, 2003. DOI: 10.4135/9781412986328.

[77] A. Carrera-Rivera, W. Ochoa, F. Larrinaga, and G. Lasa. "How-to conduct a systematic literature review: A quick guide for computer science research". In: *MethodsX* 9 (2022), pp. 1–12. DOI: 10.1016/j.mex.2022.101895.

[78] P. J. Denning. "ACM President's Letter: What is Experimental Computer Science?" In: *Communications of the ACM* 23.10 (1980), pp. 543–544. DOI: 10.1145/359015.359016.

[79] P. J. Denning. "ACM President's Letter: Performance Analysis: Experimental Computer Science as Its Best". In: *Communications of the ACM* 24.11 (1981), pp. 725–727. DOI: 10.1145/358790.358791.

[80] B. Boehm. "Spiral Development: Experience, Principles, and Refinements". In: *Spiral Development Workshop* (2000), pp. 1–49. DOI: 10.21236/ada382590.

[81] A. Tchernykh, U. Schwiegelsohn, E.-g. Talbi, and M. Babenko. "Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability". In: *Journal of Computational Science* 36 (2019), pp. 1–9. DOI: 10.1016/j.jocs.2016.11.011.

[82] T. Rangnau, R. V. Buijtenen, F. Fransen, and F. Turkmen. "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines". In: *IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC)*. 2020, pp. 145–154. DOI: 10.1109/EDOC49727.2020.00026.

[83] B. Blanchet. "Modeling and Verifying Security Protocols with the Applied Pi Calculus and ProVerif". In: *Foundations and Trends in Privacy and Security* 1–2 (2016), pp. 1–135. DOI: 10.1561/3300000004.

[84] J. Katz and Y. Lindell. *Introduction to Modern Cryptography*. 2nd. New York, NY, USA: CRC Press, 2015. ISBN: 978-1466570276.

[85] S. Jalali and C. Wohlin. "Systematic literature studies: Database searches vs. backward snowballing". In: *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. 2012, pp. 29–38. DOI: 10.1145/2372251.2372257.

[86] W. R. Shadish, T. D. Cook, and D. T. Campbell. *Experimental and quasi-experimental designs for generalized causal inference*. Boston, MA, US: Houghton, Mifflin and Company, 2002.

[87] M. D. Prado, J. Su, R. Saeed, L. Keller, N. Vallez, A. Anderson, D. Gregg, L. Benini, T. Llewellynn, N. Ouerhani, R. Dahyot, and N. Pazos. "Bonseyes AI Pipeline—Bringing AI to You: End-to-End Integration of Data, Algorithms, and Deployment Tools". In: *ACM Trans. Internet Things* (2020). DOI: 10.1145/3403572.

[88] Istio. *Security*. Online documentation. 2019. URL: https://istio.io/docs/concepts/security/ (visited on 06/18/2023).

[89] Microsoft. *Sidecar pattern*. Online documentation. 2020. URL: https://docs.microsoft.com/en-us/azure/architecture/patterns/sidecar (visited on 06/18/2023).

[90] A. Shostack. *Threat Modeling: Designing for Security*. Wiley, 2014.

[91] M. Héder. "From NASA to EU: the evolution of the TRL scale in Public Sector Innovation." In: *Innovation Journal* (2017).

[92] S.-F. Chang. "Application Marketplace as a Service - A Reference Architecture for Application Marketplace Service". In: *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. 2010, pp. 186–192. DOI: 10.1109/3PGCIC.2010.32.

[93] Y. Jiang, C.-S. Perng, A. Sailer, I. Silva-Lepe, Y. Zhou, and T. Li. "CSM: A Cloud Service Marketplace for Complex Service Acquisition". In: *ACM Transactions on Intelligent Systems and Technology* 8.1 (2016), pp. 1–25. DOI: 10.1145/2894759.

[94] D. Pudasaini and C. Ding. "Service Selection in a Cloud Marketplace: A Multi-Perspective Solution". In: *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*. 2017, pp. 576–583. DOI: 10.1109/CLOUD.2017.79.

[95] A. OI, M. Nakajima, Y. Soejima, and M. Tahara. "Reliable Design Method for Service Function Chaining". In: *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. 2019. DOI: 10.23919/APNOMS.2019.8892959.

[96] S. Popov. *The Tangle*. 2018. URL: https://assets.ctfassets.net/r1dr6vzfxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1_4_3.pdf (visited on 06/18/2023).

[97] CBAN. *Working Draft CBAN Reference Architecture*. Tech. rep. 2020.

[98] D. Woos, J. R. Wilcox, S. Anton, Z. Tatlock, M. D. Ernst, and T. Anderson. "Planning for change in a formal verification of the raft consensus protocol". In: *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs*. ACM, 2016, pp. 154–165. DOI: 10.1145/2854065.2854081.

[99] R.-V. Tkachuk, D. Ilie, R. Robert, V. Kebande, and K. Tutschku. "On the Performance of Consensus Mechanisms in Privacy-Enabled Decentralized Peer-to-Peer Renewable Energy Marketplace". In: *26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*. IEEE, 2023, pp. 179–186. DOI: 10.1109/ICIN56760.2023.10073510.

[100] G. Wood. *Ethereum: a secure decentralised generalised transaction ledger*. Ethereum Project Yellow Paper, 2014, pp. 1–32. URL: https://gavwood.com/paper.pdf (visited on 06/18/2023).

[101] A. Barger, Y. Manevich, H. Meir, and Y. Tock. “A Byzantine Fault-Tolerant Consensus Library for Hyperledger Fabric”. In: *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 2021, pp. 1–9. DOI: 10.1109/ICBC51069.2021.9461099.

Chapter Six

Towards a Secure Proxy-based Architecture for Collaborative AI Engineering

Abstract

In this paper, we investigate how to design a security architecture of a Platform-as-a-Service (PaaS) solution, denoted as Secure Virtual Premise (SVP), for collaborative and distributed AI engineering using AI artifacts and Machine Learning (ML) pipelines. Artifacts are re-usable software objects which are a) tradeable in marketplaces, b) implemented by containers, c) offer AI functions as microservices, and, d) can form service chains, denoted as AI pipelines. Collaborative engineering is facilitated by trading and (re-)using artifacts and, thus, accelerating the AI application design. The security architecture of the SVP is built around the security needs of collaborative AI engineering and uses a proxy concept for microservices. The proxy shields the AI artifact and pipelines from outside adversaries as well as from misbehaving users, thus building trust among the collaborating parties. We identify the security needs of collaborative AI engineering, derive the security challenges, outline the SVP's architecture, and describe its security capabilities and its implementation, which is currently in use with several AI developer communities. Furthermore, we evaluate the SVP's Technology Readiness Level (TRL) with regard to collaborative AI engineering and data security.

6.1 Introduction

Access to machine learning (ML) algorithms implemented in freely available and easy-to-use software development kits has lowered the bar for incorporating artificial intelligence (AI) into general-purpose applications. However, this simplicity is deceiving. Building robust and efficient AI features requires a deep understanding on how AI algorithms and data, *i. e.,* AI artifacts, interact. Moreover, ML requires large training data sets which typically exist in large enterprises that can afford to develop ML concepts. Small companies, however, might collect data or define models but must rely on collaborations with distributed stakeholders for developing AI solutions. Hence, trusted collaborations in AI engineering are needed for empowering ML beyond large companies [1].

The authors of [1, 2] developed a *collaborative form of AI engineering* within an H2020 project. This approach uses agile methods, *e. g.,* continuous integration, to accelerate collaborative AI development. Furthermore, the project has implemented an AI marketplace (MP), which is an open platform for trading AI artifacts. However, it requires a *Secure Virtual Premise (SVP)* which is a variate of a PaaS for distributed, secure and trusted AI engineering using artifacts and which is presented here. The SVP connects distributed

computation and storage resources (both physical and virtual) into a large virtual resource space for AI training and enforces a perimeter around this space. Hence, this SVP can be called a federation of distributed resources.

The MP supports the controlled exchange of AI artifacts, *i. e.,* of algorithms and data, among third parties. An application developer can obtain licensed access (*e. g.,* by paying fees) to these artifacts and use them locally, *i. e.,* in the local parts of the SVP. As a result, the developers can focus on application design while having access to AI artifacts and AI data, thus accelerating system development.

In this context, concerns arise that malicious users may try to bypass the constraints imposed by the license, or even share artifacts with unlicensed users. This would threaten not only the intellectual property rights (IPRs) of artifact owners but also may create significant data privacy issues.

The main contributions of this paper are the identification of the security challenges and requirements for collaborative AI engineering using the SVP, the definition of threat models, and the specification of a proxy-based PaaS security architecture that is based on threat analysis. *Proxies* are implemented close to AI artifacts, run *on-demand* (*i. e.,* dynamically when artifacts are instantiated), and shield the artifacts and pipelines from adversaries. To our knowledge, the SVP is the first PaaS that meets the needs of distributed and collaborative AI engineering and which enables users to have certain administration rights within distributed computational resources of SVP.

The paper uses the following *methodology* for defining the SVP's security architecture. First, it specifies the assets to be protected and identifies potential attack paths. Then, it uses a STRIDE approach [3] to define a threat profile against the assets. Finally, it translates the profile into requirements and specifies the architecture for the SVP.

The paper is structured as follows. Sec. 6.2 outlines the considered approach for *collaborative AI engineering*. Section 6.3 discusses the threats and security requirements of this concept. Sec. 6.4 introduces the SVP's secure design and discusses how it meets the requirements of the threat profile. Sec. 6.5 describes the implementation and operation of the SVP. Sec. 6.6 describes the evaluation of the current SVP implementation. Sec. 6.7 describes related work on securing digital marketplaces, AI engineering, and proxy-based security concepts. Finally, Sec. 6.8 sums up the security capabilities of the SVP and provides an outlook for future research.

6.2 Collaborative AI Engineering

The considered systematic engineering of data-driven AI solutions [2] is facilitated by the use of AI artifacts and ML pipelines, called *AI pipelines*. These pipelines are used to structure and eventually automate ML workflows. They consist of several modular steps to generate an AI application or to benchmark its accuracy. The solution of [2] assumes that each pipeline step can be provided by an *AI Tool* or *AI artifact*, which is implemented in a Docker container [4] and is similar to a microservice [5].

Fig. 6.1 shows the structure of an AI training pipeline comprising three AI tools: Data (Extraction) Tool, Training Tool, and Deployment Tool. The data is exchanged between artifacts through file objects, *e. g.,* volumes. Fig. 6.1 shows AI tools in containers, which are

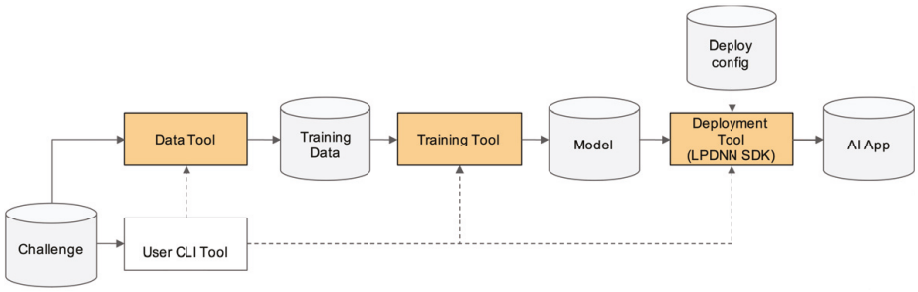


Figure 6.1: *AI Training Pipeline, after [2]*

represented by rectangles, and data objects as cylinders.

Another key element of AI engineering in [1] is the *Marketplace (MP)* for AI artifacts. The MP enables stakeholders to meet, offer, find, and exchange artifacts. Here, the users agree on terms and conditions for collaborations and artifact usage, *i. e. ,* on human- and machine-readable licenses.

While the MP is open, the Secure Virtual Premise (SVP) needs to be a closed and protected space where the AI computing tasks take place [2]. The SVP binds and federates eventually distributed compute and storage resources (due to the potential geographic distribution of stakeholders). A SVP is defined for a specific AI application design task and multiple SVPs for different applications may exist. Furthermore, a *SVP provider* may operate an SVP in a commercial way.

Only eligible users should access a specific SVP. Resources and users on the inside (*i. e. ,* on *premise*) are typically trustworthy and compliant, and everything on the outside is untrusted and potentially malicious.

6.3 Security Challenges and Requirements

From a security perspective, the AI artifacts are the assets at risk, and the SVP is the security scheme that protects them. A successful attack on this scheme will grant adversary access to the asset, such that the asset can be exfiltrated or configured to engage in potentially malicious behavior. To have a complete security definition, a threat model is also needed to outline the capabilities of the adversaries. This is done next.

6.3.1 Inside and Outside Adversaries

We choose a binary model consisting of adversaries either on the *inside* or the *outside* of the SVP. Inside adversaries are legitimate SVP users, formerly trustworthy, but now turned rogue, *e. g. ,* avoiding license fees. Outside adversaries have no legitimate access to the SVP and thus have less "power" to attack the security guarantee.

Protection against outside adversaries is typically enforced by access control (AC) mechanisms located at the boundary of the SVP. Thus, the security level is directly related to their robustness. Inside adversaries that have successfully passed the AC scheme can

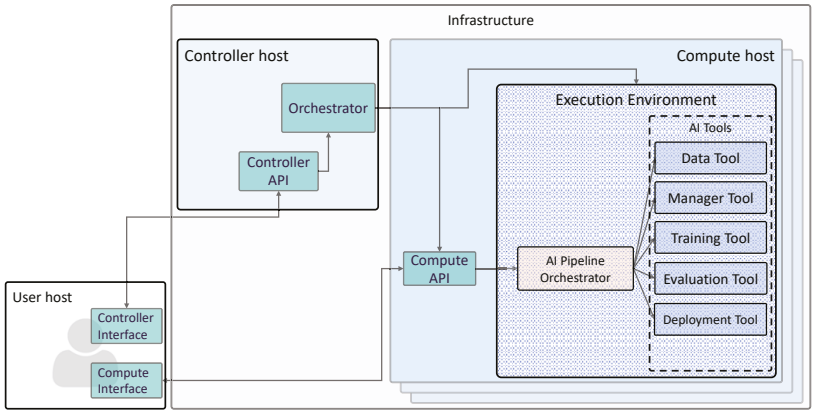


Figure 6.2: *SVP architecture without security mechanisms.*

access the SVP resources, *e. g.*, AI artifacts, that they are authorized for. Hence, they have the opportunity to tamper with resources, *e. g.*, attackers could instrument the execution environment (EE) to reverse engineer (RE) and bypass the license checking.

In general, handling inside adversaries is quite difficult. The authors of [6, 7] approached the problem by classifying adversaries into *regular users* and *malicious users*. The main difference between them is that regular users are not able to tamper with the software. By excluding malicious users from the threat model, authors were able to design a scheme based on public-key cryptography to ensure compliance from both outside and inside adversaries.

6.3.2 Tampering with the System Software

Software tampering (ST) is an umbrella term describing activities that aim at introducing unwanted modifications to binaries or to processes running on a host. ST provides possibilities for multiple attacks, including RE. Software-based defenses against ST rely primarily on code obfuscation [8]. Hardware-based defenses [9] are more robust to software attacks but are dependent on hardware from a specific CPU manufacturer and usually incompatible with each other.

The threat posed by ST through RE to the SVP exists because skilled *inside* adversaries (*i. e.*, legitimate users turned rogue) have direct access (physical or remote) to the host running the artifact. By removing direct access, the RE possibilities are severely curtailed. This requires a slight modification to the threat model defined in [6, 7]. In the modified model, the users and hosts are no longer trustworthy and no longer inside the SVP. They can still access resources from the SVP, but only through well-defined interfaces, as shown by the blue rectangles inside the User host box in Fig. 6.2. An interface is a stub proxying requests towards AI objects in the SVP and receiving responses from the SVP, much similar to the stub concept used by remote procedure calls (RPC). Thus, the user host interacts through interfaces, and the actual artifact is executed remotely on hosts belonging to the SVP, which acts as an infrastructure for the users.

We assume that SVP providers are not malicious, nor do they collude with adversaries.

We think this is a reasonable assumption because otherwise, the SVP providers would put their own business model at risk and lose their reputation.

6.3.3 Threat Model

Fig. 6.2 shows the functional SVP architecture without security mechanisms. An AI user communicates with the SVP and the AI components through two different Application Programming Interfaces (APIs). These are shown as blue rectangles inside the left box denoted User host. The user end of the APIs is connected to the server end on the Controller host (CtrlH) and Compute host (CpH). The Controller Interface enables the user to instruct the *orchestrator* to provision the EE inside the CpHs. The Compute Interface is used to download AI artifacts from an MP repository (not shown in the figure) to the EE on the CpH, to assemble the AI pipeline, and to manage the pipeline and its components during their life cycle. The AI artifacts inside the EE are the main assets at risk.

The concern is that these assets can be exfiltrated, configured to disobey the license terms, or modified to operate maliciously. An attacker operating the User host can follow two paths to reach the main asset.

The first attack path begins at the Compute interface and ends at the Compute API (CpAPI) component on the CpH. The Compute Interface can execute CpAPI calls that may result in potentially harmful actions toward elements inside of the EE. These types of API calls must be allowed only for privileged users. Thus, a potential threat is that attackers can escalate their privileges by spoofing the identity of a privileged user. Although the API does not provide means to exfiltrate AI artifacts, the attacker may be able to launch API calls that modify ownership and access rights to the pipeline and its components and to exfiltrate pipeline results. Even when the attackers can only impersonate non-privileged users, they can engage in damaging activities such as stopping, removing, or reconfiguring pipelines in order to tamper with the output of the pipeline, *e. g.* , degrading output quality, or to overload the infrastructure through Denial-of-Service (DoS) attacks.

Table 6.1: *Identified threats*

Threat	Element	Description
T01	CpAPI	Illegitimate user accesses CpAPI.
T02	CpAPI	Spoofing the identity of privileged users to execute undesirable or dangerous CpAPI calls towards the EE.
T03	User-CpAPI traffic	Interception and possibly tampering of data exchanged between the User host and CpAPI. May enable T-01 and T-02.
T04	CtrlAPI	Illegitimate user accesses CtrlAPI.
T05	CtrlAPI	Spoofing the identity of privileged users to the CtrlAPI. Enables access to EEs via the Orchestrator and exfiltration or damage to AI artifacts.
T06	User-CtrlAPI traffic	Interception and likely tampering of data exchanged between the User host and CtrlAPI. May enable T-01 and T-02.
T07	CtrlAPI	Malicious privileged user secretly exfiltrates artifacts and/or conducts operations with the intent of creating harm to the pipeline infrastructure.
T08	CpAPI	Malicious privileged user secretly conducts operations with the intent of creating harm to the pipeline and its output.

The second attack path starts at the Controller interface and continues, first through the Controller API (CtrlAPI) and then through the Orchestrator on the CtrlH. An attacker

who is able to infiltrate the system over this path obtains backdoor access to the EE and to the CpAPI. The Orchestrator can execute operations that interfere with AI artifacts and pipelines. Only privileged users must be able to execute these operations. However, the Orchestrator is not directly exposed to the User host, and it can only be reached through the CtrlAPI. Thus, an attacker must mount a successful privilege escalation attack against the CtrlAPI in order to control the EE through the Orchestrator. For example, an attacker from a User host would need to spoof the identity of users with privileged access. If successful, the adversary will obtain full access to the CpH OS and to its EE. Implicitly, the attacker will have access to the AI components inside and will be able to interfere with CpAPI calls. Tampering with CtrlAPI calls presents similar risks as described in the previous paragraph.

Table 6.2: *Threat profile*

Threat	S	T	R	I	D	E
T01	X			X		X
T02	X			X	X	X
T03		X		X	X	
T04	X			X		X
T05	X			X	X	X
T06		X		X	X	
T07		X	X		X	
T08		X	X		X	

Table 6.3: *Security requirements.*

Req.	Description
R01	User's identity and authentication must be based on X.509 digital certificates. Mitigates spoofing (T01, T02, T04, T05).
R02	User host to CtrlH / CpH communication must use HTTPS (SSL/ TLS) with certificate-based mutual authentication. Mitigates tampering (T03, T06) and information disclosure (T03, T06).
R03	CpAPI must use role-based access control (RBAC) to control which users can execute privileged API calls. Mitigates information disclosure (T01, T02), DoS (T02) and elevation of privilege (T01, T02).
R04	CtrlAPI must use RBAC to control which users can execute privileged API calls. Mitigates information disclosure (T04, T05), DoS (T05) and elevation of privilege (T04, T05).
R05	CpH must use secure logging that cannot be tampered with from the CpAPI to log all actions. Mitigates repudiation (T07). Secure logging does not mitigate tampering and DoS (T07) but provides information about unwanted activities, which allows the system administrator to take action against malicious users.
R06	CtrlH must use secure logging that cannot be tampered with from the CtrlAPI to log all API calls. Mitigates repudiation (T08). Secure logging does not mitigate tampering and DoS (T08) but provides information about unwanted activities, which allows the system administrator to take action against malicious users.

Even if attackers do not control the User host, they may be able to interfere with the data flows between the User host, CtrlHs, and CpHs. Unless the network traffic is protected, attacks on confidentiality, integrity, and authenticity become possible. For example, attackers would be able to extract credentials that enable them to impersonate legitimate users. DoS attacks from outside the SVP towards the hosts in Fig. 6.2 are also a potential threat. However, the SVP mechanisms described here do not aim to address this type of threat, as DoS attack mitigation typically requires the involvement of network operators [10].

Using the above-identified threats, *c.f.*, Table 6.1, we use the STRIDE approach [3] to produce the SVP's threat profile, *c.f.*, Table 6.2. Each of the letters in STRIDE denotes a

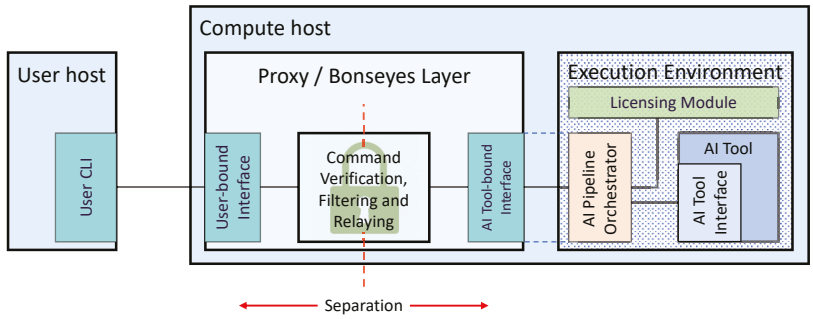


Figure 6.3: *Security and Separation by BL Reverse Proxy.*

specific threat category against a desirable security property: spoofing *vs.* authentication, tampering *vs.* integrity, repudiation *vs.* non-repudiation, information disclosure *vs.* confidentiality, DoS *vs.* availability, and elevation of privilege *vs.* authorization. As a result, we identified the requirements shown in Table 6.3.

6.4 Proposed Security Architecture

Next, we will present the SVP components used to address and implement the security requirements derived in Sec. 6.3.

6.4.1 A Proxy Concept for Artifact Security

Proxies provide security by applying the security concepts of *separation* and *verification* [11]. They implement services such as authentication, encryption of data communication, network isolation, or access control. [12] proposes proxies for addressing security, trust and privacy for collaboration in multicloud environments. However, they were not located close to the offered services. Lately, so-called *sidecar proxies* [13, 14] have been suggested. They are attached to parent services and provide them with supporting features.

The SVP architecture employs a reverse proxy which is denoted as *Proxy / Bonseyes Layer (BL)*. This proxy shields the artifact from misuse, is implemented next to the artifact like a sidecar proxy, and is started on-demand when the artifact is instantiated, *c. f. ,* Fig. 6.3.

The Proxy / BL is executed on the CpH and it interfaces to the User host by the *User-bound interface*. A user can connect to an artifact (here *AI tool*) only through this interface. Furthermore, the BL interfaces to the AI tool via the *AI Tool-bound interface*. The Proxy / BL relays only verified and filtered commands between these interfaces. These interfaces manage the control of the artifact and the pipeline. The AI Tool-bound interface has the additional advantage that the proxy can be adapted to arbitrary containerized tools, *i. e. ,* the available AI assets don't need to be adapted. A *trust boundary* between the user and AI artifact is implemented close to the artifact by permitting connections only to the Proxy / BL while having the Proxy / BL verify the eligibility of received commands. We believe that it is a compelling feature of this proxy concept that it provides clear separation without requiring any explicit support from the artifact. This reduces the efforts of developer

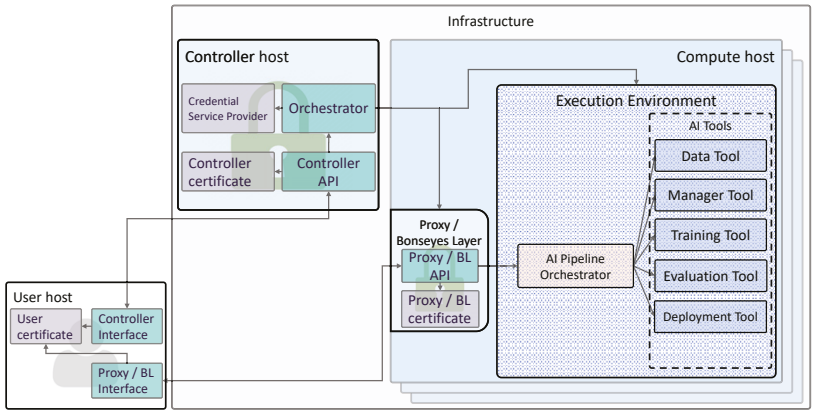


Figure 6.4: *Generic architecture of SVP.*

since they can focus on AI functions only and don't need to build security mechanisms into the artifact.

Fig. 6.3 depicts the *AI Pipeline Orchestrator*. It coordinates activities among the artifacts by exposing an interface to the AI tools. Also, it performs the syntax translation from a user command to the function calls of the AI tool. Unlike the Proxy / BL, the orchestrator usually does not perform any security checks, unless it has been delegated to do so.

6.4.2 Elements of the SVP Architecture

The requirements listed in Table 6.3 indicate the need to support the concept of a digital identity. Hence, the SVP architecture follows the NIST guidelines for digital identities [15] and relies here on three specific entities: Credential Service Provider (CSP), Verifier, and Relaying Party (RP). The CSP enables users (*applicants*) to enroll in digital identity services. If successful, the applicant becomes a *subscriber*, who will eventually attempt to claim access to a secured service. The Verifier will interact with the user to ascertain the *claimant* is a valid subscriber (*i. e.*, authentication is performed). The Verifier may also contact the CSP to obtain additional claimant attributes that are required by the RBAC in use. The result of the authentication is passed on to the RP, which can complete the RBAC and determine if the type of access required by the claimant is allowed (*i. e.*, authorization is performed).

The applicant who wishes to utilize the services of the SVP must undergo first a process called *identity proofing*, where the user's real-world identity is connected to a digital identity. The CSP contains a sub-function called Registration Authority (RA), which is the equivalent of a front desk where the physical applicant proves their real-world identity (*e. g.*, by showing a passport). Upon successful identification, the RA invokes another sub-function of the CSP, the Certification Authority (CA), to create a X.509 digital user certificate. The certificate is installed in the applicant's web browser (on User host in Fig. 6.4) and becomes the applicant's digital identify. Furthermore, it enables the Verifier and RP to conduct certificate-based authentication and authorization of users. **This addresses requirement R01**, *c. f.*, Table 6.3.

Table 6.4: *Location of digital identity elements*

Entity	Location
Applicant/Subscriber/Claimant	User host
CSP (with RA and CA)	CtrlH
Controller RP (incl. Verifier)	CtrlAPI on CtrlH
Proxy RP (incl. Verifier)	Proxy / BL API on CpH

For simplicity, we have kept the CA and RA inside the CSP, although in some scenarios they can be separate entities managed by different organizations. In Fig. 6.4, they are shown as the box denoted "Credential Service Provider" inside the CtrlH. Currently, the architecture contains two RPs: the Controller RP located within the Controller API and the Proxy RP located within the Proxy / BL API. Again, for simplicity, the functionality of the Verifier is merged in each of the RPs.

The CA is also used to produce digital server certificates for the elements listed in Table 6.4. These are installed in web servers running at the location shown in the same table and serve a dual-purpose. First, they enable authentication of the elements towards the User host. Secondly, they are fundamental in enabling encrypted communication over HTTPS. **This addresses requirement R02.**

The Controller RP interacts with the CSP to implement a simple RBAC scheme that differentiates between *regular* and *privileged* users. The CSP provides the subscriber attributes required by the Controller RP to enforce the scheme. **This addresses requirement R04.** All requests and replies that are processed by the CtrlAPI (with associated RP) are logged at the CtrlH. Logging features cannot be controlled from the CtrlAPI and thus logs can't be tampered with this way. **This addresses requirement R06**.

The Proxy / BL works as an interceptor between the User host and the EE on the CpH. Through the Proxy / BL, the User host can interact with the AI pipeline and various AI tools. Incoming Proxy API calls from the User host are processed by the Proxy / BL into low-level sets of API calls, each targeting specific components within the EE. This is required because although the Proxy BL exposes a uniform API towards the User host, the APIs of the components within the EE may vary substantially. Similar to the Controller RP, the RBAC scheme for the Proxy RP must also be able to differentiate between regular an privileged users. However, it must be able to support attributes that describe the type of low-level API calls that are allowed towards a component. In addition, all API calls (both high- and low-level) are recorded similarly as explained for the Controller RP. **This addresses requirements R03 and R05.**

6.5 SVP Implementation

The structure of the SVP implementation is shown in Fig. 6.5. It uses currently available technologies to implement the security mechanisms and requirements, cf. Table 6.3.

6.5.1 Controller Host (CtrlH)

At the CtrlH, the *CSP* is implemented with the *OpenSSL* as a bash script that manages certificates in an automated manner. *Controller certificate* is created initially and *User* and

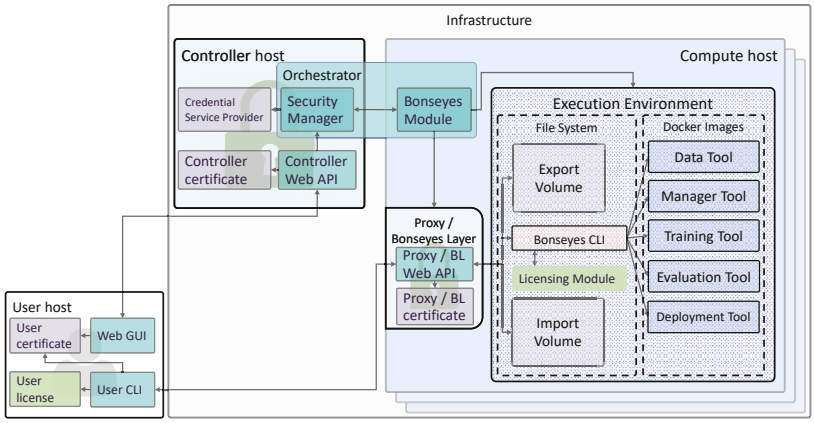


Figure 6.5: *Implemented architecture of the SVP.*

Proxy / BL certificates are created on demand.

The *CtrlAPI* is implemented as a *Python-Flask* web application that runs under *uWSGI* web server. uWSGI server is started with an X.509 CtrlH certificate generated by the CSP.

As part of the system design, the architecture of the *Orchestrator* can follow one of the following patterns: *a) Client-Server*, *b) Server-Only*, as was described in [16]. In a Server-Only architecture, the Orchestrator software is installed only on the CtrlH, whereas in a Client-Server architecture, there is also agent software installed on CpHs. We chose the *Client-Server* architecture because this allows the agent to individually check the integrity of downloaded AI artifacts. Orchestrator consists of two components – *Security Manager* on Server-side and *Bonseyes Module (BM)* on Client-side. They are implemented as Python-Flask web applications running under the uWSGI web server, with a high degree of integration between them. Namely, each BM belongs to a specific Security Manager and their communication is encrypted using HTTPS.

6.5.2 Compute Host (CpH)

The provisioning of the CpH is performed by the BM, which requests the *Proxy / BL certificate* from the Security Manager and downloads and starts all *Proxy / BL* and *EE* related components from a repository (not shown in Fig. 6.5).

The *Proxy / BL* is implemented in the form of a Docker container. It intercepts the communication between the User host and the EE and allows only a strictly defined set of *Proxy / BL Web API* calls to be executed. The Proxy / BL Web API is implemented as a Python-Flask web application that runs under the uWSGI web server. The server is started with X.509 Proxy / BL certificate generated by CSP. In order to communicate with the proxy's Web API, a user executes commands over *User Command Line Interface (User CLI)*. The User CLI sends both privileged and regular commands, which are filtered by the proxy, based on the user's role.

The EE resides inside of the CpH and contains all the needed data and tools to conduct AI pipeline execution. The *Import Volume* is used to store information within the EE, which

can later be used during AI pipeline execution. It is implemented as a folder in the CpH's file system which is attached as a volume to the Proxy / BL docker container. AI artifact users can upload data into Import Volume, but cannot execute or download it - this data is designed to be used only by *Bonseyes Command Line Interface (Bonseyes CLI)*. The Bonseyes CLI is the implementation of the AI Pipeline Orchestrator shown in Fig. 6.4. Its purpose is to control the respective AI Tools and provide them mediated access to data residing inside Import Volume. The Bonseyes CLI is implemented as a Python module that can be invoked by executing User CLI commands and is attached to the Proxy / BL docker container as a volume. AI Tools are implemented in Docker containers which are started by the Bonseyes CLI and receive input from it. After the AI Tool finished execution, the Bonseyes CLI gathers the output and places it in *Export Volume*, which is used to transfer information out from the EE, thus from the SVP. The Export Volume is implemented in the same manner as the Import Volume.

The *Licensing Module* is another essential security mechanism that resides within the EE. It is implemented with Python and ensures compliant usage of the artifact according to the *User License*. The user needs to upload the User license to the SVP and its validity is checked by Bonseyes CLI before the execution of any command from the user.

6.6 SVP Evaluation

A system-wide evaluation of the SVP's security level is difficult since it combines a variety of complex security mechanisms. This makes it prohibitive to rely on symbolical verification (*i. e.,* formal model) [17] or computational approaches (*e. g.,* reduction techniques) [18] to assess the security of the system. Hence, we decided to apply here a more differentiated approach that builds on practical use cases to verify its usability and security.

The SVP implementation was tested by several real-world AI engineering cases. These use cases permit judging the SVP's maturity and thus evaluating its *Technology Readiness Level (TRL)* [19]. The use cases were carried out by stakeholders and users of the collaborative AI engineering process of [1, 2] and in their labs and computing infrastructures. The stakeholders comprised two companies that were interested in AI application design, an SME (small- and medium-sized company) and a large automotive OEM manufacturer, as well as two international universities developing AI algorithms. The use-cases comprised the tasks of "AI training" and "AI model benchmarking" using AI pipelines, *c. f.,* Fig. 6.1. The various actors in the use cases were assigned one of the following roles: *legitimate SVP provider, legitimate SVP user, rogue SVP user*, and *outside adversary*.

Throughout these use cases, a legitimate SVP user and SVP provider were always able to execute the AI pipelines and to exchange the artifacts (Remark: this confirms the AI engineering capabilities of the SVP). However, a rogue SVP user and an outside adversary were experiencing the enforcement of security mechanisms. In detail, an outside adversary was given the task to penetrate into the EE and interfere with the process of pipeline execution. When an adversary attempted a connection, it was blocked by the X.509 digital certificate authentication. Moreover, since only network port 443 was open, there was no other point of interaction offered by the system. In parallel, the embedded logging system recorded every unsuccessful connection attempt for further analysis.

Another use case introduced a rogue SVP user on the inside of SVP's perimeter, having two malicious goals: 1) obtain the AI data and tools which are stored in the execution environment, and 2) execute an AI benchmarking tool with an expired license. By design, the Proxy / BL enforces for regular users a store-only mode for the data inside of the Import Volume. This prevents data exfiltration from the Import Volume. Only privileged users or programs can obtain direct access to data inside the Import Volume. In completing the first goal, the rogue user must authenticate himself using his digital certificate. Using the identity from the certificate, the RBAC mechanism determined that regular user privileges apply, and thus blocked read access to the assets.

In order to complete the second goal, the rogue user modified the license, which is encoded in plain-text JSON format, to extend the license expiration date. The modified license was uploaded as part of the pipeline construction process to the SVP along with the original license signature. This signature was computed over the sha256 hash of the license content, using a private key known only to the MP. During license verification, the SVP first verified the license signature using MP's public key. Since the license content was changed, the signature verification failed. Thus, the execution of the benchmarking tool was blocked.

The use cases confirmed that the AI and security requirements were appropriately addressed and that the security capabilities of the SVP provide sufficient trust for enabling the collaborative AI engineering process of [1, 2]. Moreover, users of this AI engineering process concluded in [20] that the SVP implementation provided by us achieves at least TRL 4 ("Technology validated in lab") and eventually also TRL 5 (TRL 4 + "... relevant environment (industrially relevant environment in the case of key enabling technologies)"). While the TRL shows the maturity of the approach, we suggest to carry out a more threat-focused verification in the future, *e. g.*, a full-fledged penetration test of a live SVP.

6.7 Related Work

Protecting collaborative and distributed software engineering environments involves a large number of technologies. Next, we outline important related works and outline the context for the research presented here.

Securing Docker containers has recently gained significant attention [21, 22]. Our approach complements the Docker-specific efforts by not relying on a specific containerization technique and adding another layer of protection.

Software marketplaces for Virtual Network Functions (VNFs) and AI are going back as early as 2014 [23] but rarely support edge clouds or multi-stakeholder collaborations. Trusted and secure collaborative software marketplace substrates and platforms using blockchains have been suggested lately [24, 25]. Ericsson's Nubo platform [24], however, doesn't support licenses and hasn't yet an option for distributed computation. The OceanProtocol [25] is the closest known larger-scale concept to the MP [1] and the SVP concepts. However, it also doesn't enforce licenses and requires the Docker containers to be adopted by a blockchain layer.

Related work on the network scope and dynamics of proxies were discussed in Section 6.4.1 in order to highlight the features of our mechanism. Recent proxy solutions for microservices and service chains are discussed in [26, 27]. Hereby, [26] is close to the proposed *Proxy*

/ BL concept. However, it is tailored to offline devices and the proposed architecture is based on the usage of *secure cryptoprocessors*, such as the trusted platform module (TPM). Thus, their proposal is not applicable to platforms lacking this type of devices. Distributed business processes employing proxies are detailed in [27]. However, their proposed proxy-based controller is designed for protecting document flows only. Other uses, such as chained services as AI pipelines are, not considered.

6.8 Summary and Outlook

In this work, we designed a PaaS solution for securing containerized AI artifacts and enabling trust in collaborative AI engineering. The design is based on a rigorous threat and security analysis process: *threat modeling* → *security requirements definition* → *security services development*. The idea of securing AI artifacts and pipelines is implemented by the use of the *Proxy / Bonseyes Layer (BL)*, which implements the security concepts of *separation* and *verification*. The BL is placed on-demand, close to each artifact, enforces licenses, and prevents direct access from users. The evaluation of the SVP has demonstrated that it has matured security and can be applied in real-world collaborative AI engineering using software marketplaces and AI pipeline concepts.

We think that our solution allows the SVP to be stretched into devices on edge clouds. In this scenario, edge devices act as CpHs that are managed from the SVP CtrlH. However, in order for the SVP to remain trustworthy, the edge operator must enjoy the same level of trust as the SVP provider.

As future work, we suggest investigating how to integrate and extend the MP and SVP concepts for the deployment of VNFs or Service Function Chains (SFC) [28] trustfully on edge devices of collaborating parties or service providers.

Acknowledgment

The authors would like to thank Lorenzo Keller, Samuel Fricker, and Yuliyan Maksimov for their support. Furthermore, this work has received partial funding from the European Unions Horizon 2020 research and innovation program under grant agreement No 732204 (Bonseyes). This work is supported by the Swiss State Secretariat for Education Research and Innovation (SERI) under contract number 16.0159. The opinions expressed, and arguments employed herein do not necessarily reflect the official views of these funding bodies.

References

[1] T. Llewellynn, M. M. Fernández-Carrobles, O. Deniz, S. Fricker, A. Storkey, N. Pazos, G. Velikic, K. Leufgen, R. Dahyot, S. Koller, G. Goumas, P. Leitner, G. Dasika, L. Wang, and K. Tutschku. "BONSEYES: Platform for Open Development of Systems of Artificial Intelligence". In: *Proceedings of ACM International Conference on Computing Frontiers*. Siena, Italy, 2017. DOI: 10.1145/3075564.3076259.

[2] M. D. Prado, J. Su, R. Saeed, L. Keller, N. Vallez, A. Anderson, D. Gregg, L. Benini, T. Llewellynn, N. Ouerhani, R. Dahyot, and N. Pazos. “Bonseyes AI Pipeline—Bringing AI to You: End-to-End Integration of Data, Algorithms, and Deployment Tools”. In: *ACM Trans. Internet Things* (2020). DOI: 10.1145/3403572.

[3] A. Shostack. *Threat Modeling: Designing for Security*. Wiley, 2014.

[4] D. Merkel. “Docker: lightweight linux containers for consistent development and deployment”. In: *Linux journal* (2014).

[5] J. Thönes. “Microservices”. In: *IEEE Software* (2015). DOI: 10.1109/MS.2015.11.

[6] V. A. Mehri, D. Ilie, and K. Tutschku. “Privacy and DRM Requirements for Collaborative Development of AI Applications”. In: *Proc. ARES*. Hamburg, Germany, 2018, pp. 1–8. DOI: 10.1145/3230833.3233268.

[7] V. A. Mehri, D. Ilie, and K. Tutschku. “Designing a Secure IoT System Architecture from a Virtual Premise for a Collaborative AI Lab”. In: *Proceedings of DISS*. San Diego, USA, 2019, pp. 1–7. DOI: 10.14722/diss.2019.23006.

[8] C. Collberg, C. Thomborson, and D. Low. *A Taxonomy of Obfuscating Transformations*. Tech. rep. 148. Auckland, New Zealand: Dept. of Computer Science, The University of Auckland, 1997.

[9] B. Parno, J. M. McCune, and A. Perrig. “Bootstrapping Trust in Commodity Computers”. In: *Proceedings of IEEE Symposium on Security and Privacy*. Oakland, CA, USA, 2010, pp. 1–10. DOI: 10.1109/SP.2010.32.

[10] T. Peng, C. Leckie, and K. Ramamohanarao. “Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems”. In: *ACM Computing Surveys* (2007), pp. 1–42. DOI: 10.1145/1216370.1216373.

[11] M. Schumacher, E. Fernandez-Buglioni, D. Hybertson, F. Buschmann, and P. Sommerlad. *Security Patterns: Integrating security and systems engineering*. John Wiley & Sons, 2013.

[12] M. Singhal, S. Chandrasekhar, T. Ge, R. Sandhu, R. Krishnan, G.-J. Ahn, and E. Bertino. “Collaboration in multicloud computing environments: Framework and security issues”. In: *Computer* 46.2 (2013), pp. 76–84. DOI: 10.1109/MC.2013.46.

[13] Istio. *Security*. Online documentation. 2019. URL: https://istio.io/docs/concepts/security/ (visited on 06/18/2023).

[14] Microsoft. *Sidecar pattern*. Online documentation. 2020. URL: https://docs.microsoft.com/en-us/azure/architecture/patterns/sidecar (visited on 06/18/2023).

[15] P. A. Grassi and J. L. Fenton. *Digital Identity Guidelines*. NIST Special Publication 800-63-3. June 2017.

[16] R.-V. Tkachuk, D. Ilie, and K. Tutschku. “Orchestrating Future Service Chains in the Next Generation of Clouds”. In: *Proceedings of SNCNW*. Luleå, Sweden, June 2019.

[17] B. Blanchet. “Modeling and Verifying Security Protocols with the Applied Pi Calculus and ProVerif”. In: *Foundations and Trends in Privacy and Security* 1–2 (2016), pp. 1–135. DOI: 10.1561/3300000004.

[18] J. Katz and Y. Lindell. *Introduction to Modern Cryptography*. 2nd. New York, NY, USA: CRC Press, 2015. ISBN: 978-1466570276.

[19] M. Héder. "From NASA to EU: the evolution of the TRL scale in Public Sector Innovation." In: *Innovation Journal* (2017).

[20] S. Fricker (Edt.) *Validation and Open Developer Community Report - Del. D2.5*. Tech. rep. H2020 Bonseyes – AI Marketplace, Jan. 2020.

[21] X. Gao, Z. Gu, M. Kayaalp, D. Pendarakis, and H. Wang. "ContainerLeaks: Emerging security threats of information leakages in container clouds". In: *2017 47th IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)*. 2017, pp. 237–248. DOI: 10.1109/DSN.2017.49.

[22] Docker Inc. *Content trust in Docker*. URL: https://docs.docker.com/engine/security/trust/content_trust/.

[23] G. Xilouris, E. Trouva, F. Lobillo, J. M. Soares, J. Carapinha, M. J. McGrath, G. Gardikis, P. Paglierani, E. Pallis, L. Zuccaro, Y. Rebahi, and A. Kourtis. "T-NOVA: A marketplace for virtualized network functions". In: *2014 European Conference on Networks and Communications (EuCNC)*. 2014, pp. 1–5. DOI: 10.1109/EuCNC.2014.6882687.

[24] J. Kempf, S. Nayak, R. Robert, J. Feng, K. R. Deshmukh, A. Shukla, A. O. Duque, N. C. Narendra, and J. Sjöberg. "The Nubo Virtual Services Marketplace". In: *CoRR* abs/1909.04934 (2019). arXiv: 1909.04934.

[25] Ocean Protocol Foundation. *Ocean Protocol: A Decentralized Substrate for AI Data and Services*. Tech. rep. 2019. URL: https://oceanprotocol.com/.

[26] M. Denis, C. Johansen, and A. Jøsang. "Offline Trusted Device and Proxy Architecture Based on a new TLS Switching Technique". In: *2017 International Workshop on Secure Internet of Things (SIoT)*. 2017, pp. 10–19. DOI: 10.1109/SIoT.2017.00007.

[27] N. Maroua, A. Adel, and Z. Belhassen. "A New Formal Proxy-Based Approach for Secure Distributed Business Process on the Cloud". In: *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*. 2018, pp. 973–980. DOI: 10.1109/AINA.2018.00142.

[28] C. Zhang et al. "L4-l7 service function chaining solution architecture". In: *Open Networking Foundation, ONF TS-027* (2015).

Chapter Seven

A Survey on Blockchain-based Telecommunication Services Marketplaces

Abstract

Digital marketplaces were created recently to accelerate the delivery of applications and services to customers. Their appealing feature is to activate and dynamize the demand, supply, and development of digital goods, applications, or services. By being an intermediary between producer and consumer, the primary business model for a marketplace is to charge the producer with a commission on the amount paid by the consumer. However, most of the time, the commission is dictated by the marketplace facilitator itself and creates an imbalance in value distribution, where producer and consumer sides suffer monetarily. In order to eliminate the need for a centralized entity between the producer and consumer, a blockchain-based decentralized digital marketplace concept was introduced. It provides marketplace actors with the tools to perform business transactions in a trusted manner and without the need for an intermediary. In this work, we provide a survey on Telecommunication Services Marketplaces (TSMs) which employ blockchain technology as the main trust-enabling entity in order to avoid any intermediaries. We provide an overview of scientific and industrial proposals on the blockchain-based online digital marketplaces at large, and TSMs in particular. We consider in this study the notion of *telecommunication services* as any service enabling the capability for information transfer and, increasingly, information processing provided to a group of users by a telecommunications system. We discuss the main standardization activities around the concepts of TSMs and provide particular use-cases for the TSM business transactions such as SLA settlement. Also, we provide insights into the main foundational services provided by the TSM, as well as a survey of the scientific and industrial proposals for such services. Finally, a prospect for future developments is given.

7.1 Introduction

Digital services and network computing constitute integral parts of today's and future telecommunication infrastructures [1, 2]. The services may range from operating high-performance hardware, dedicated computation for AI, Big Data services and extended to vertically integrated applications [3]. *Digital services marketplaces* were lately introduced as distribution platforms and permitted a booming economy on digital goods, *e. g.* , Apple's gleaming *App Store*[1].

[1] https://www.apple.com/app-store/

Marketplaces are appealing mechanisms to deliver digital goods, including services. The developers (*producers*) can take advantage of bundling effects of a marketplace, *e. g. ,* indexing, cataloging, or storing goods, deploying software, or advertising on the marketplace. The platforms allow the developers to *supply* their digital products through a trusted intermediary (*marketplace*) without having to take care of legal implications of business transactions, *e. g. ,* billing. Customers, in turn, can express their *demand* and may take advantage of the amount of supply and of the simplicity to locate goods and services on a marketplace [4]. Marketplaces in telecommunication systems may also be winning for today's network operators, denoted here as *CSPs (Communication Service Providers)*. The platforms can bring additional revenues and innovations beyond simply accelerating the connectivity. The marketplace may expose services and engage developers to implement applications using these services [5], *e. g. ,* games or AI model training.

The major business model of marketplaces is to charge the producer a *commission* on the amount paid by a customer. This commission is eventually used to maintain and operate the marketplace and its infrastructure. Digital service marketplaces have often a centralized architecture and act as trust, assurance, and governance providers for the market participants and their transactions. This centralization allows for efficient cataloging or easy billing. In addition, it permits an uncomplicated implementation of the required trust mechanisms. The centralization, however, opens up negative effects. It permits the operators of marketplaces to combine the security requirements of the participants with their pricing ambitions, which in turn is likely to create asymmetry in value generation [6]. Moreover, centralized architectures suffer disproportionately when unauthorized access is gained. In such an architecture, an attacker, if gained unauthorized access, can compromise a large number of identities and eventually all identities. Hence, a *distributed marketplace architecture* is preferred when these platforms should have less controlled business models avoiding monopolies or when they should be more robust to attacks.

An appealing way to eliminate a centralized entity between the producer and consumer is to provide the marketplace actors with the set of tools to perform the business transaction in a distributed and trusted manner. Such a task may be achieved by the blockchain technology [7], which is an implementation of *Distributed Ledger Technology (DLT)*. DLT provides system participants with distributed storage and brings benefits such as data provenance, accountability, and transparency to distributed systems. Moreover, DLT allows to reduce or completely eliminate the need for a trusted third-party [8], *e. g. ,* the marketplace, from the business transaction process, and bring balance to value distribution inside digital marketplaces. Although blockchain technology is still in its infancy, it has enabled a significant number of application scenarios in today's digital marketplaces, which we discuss in this work.

In addition, the process of business transaction execution, *i. e. ,* the business settlement, has gained importance as it enables to reach the final business agreement. Today's CSPs enjoy their independence and build their network infrastructures with the centralized operation and governance [9]. In order to execute inter-CSP business transactions, *e. g. ,* for allowing mobile customers to roam across different operators infrastructure and to pay for the usage, a *third-party* has to be involved, which acts as a *trust provider* towards non-trusting CSPs participating in business relations. Another use-case is the business transaction between

customer and CSP. In this case, the signing of a Service Level Agreement (SLA) [10] may take place where the CSP commits to provide a customer with a certain level of quality of service (QoS) [11] for infrastructure and telecommunication services. Having a third-party in the middle results currently in parts of the process being executed manually which can be complex, expensive, and time-consuming [9]. The application of DLTs in inter-CSP and customer-CSP business transactions may allow the automation of transaction processes. In this way, the need for any manual human efforts can be eliminated almost totally as the DLT acts as a *trust-enabling* entity which under agreed rules, defined in smart contracts, does not need a trusted third-party to take care of parts of the transaction. In the case of SLA signing, the conditions on agreed QoS can be recorded on the distributed ledger [12], as well as intermediate measurements of service quality. In this way, the DLT as a trusted distributed storage enables all parties to agree on the recorded data and to settle in the case of SLA violation.

In this work, we describe, analyze and discuss the concept of a distributed *Telecommunication Services Marketplace (TSM)* which employs blockchain technology as the main trust enabling entity and which integrates multiple services offered by different CSPs. We outline the capabilities of distributed TSMs that provide a common set of processes that CSPs can trust and rely on. In addition, we provide a survey on scientific and industrial proposals on the blockchain-based digital marketplaces at large and blockchain-based TSMs in particular. We discuss major standardization activities around the concepts of blockchain-based TSMs and provide use-cases for TSM business transaction functions. Furthermore, we provide insights into the main services provided by blockchain-based TSM, as well as a survey of the scientific and industrial proposals for such services. Finally, a prospect for future developments is given.

The remainder of the paper is structured as follows. In Section 7.2 the methodology of this survey is described along with the discussion of related survey collections on digital marketplaces at large and proposals on TSMs in particular. Section 7.3 describes the technologies and background of blockchains and digital marketplaces. Furthermore, in this section, we discuss the *a)* scientific and industrial proposals for blockchain-based marketplaces, *b)* the benefits of using blockchains in digital marketplaces, and *c)* we describe a generic structure of blockchain-based TSMs. Section 7.4 discusses the main services provided by blockchain-based TSMs and surveys on scientific and industrial proposals for them. Section 7.5 discussed the prospects for future work in an area of blockchain-based TSM. Finally, Section 7.6 draws conclusions on the blockchain-based digital marketplaces at large and TSMs in particular.

7.2 Related Literature Overview

The overview of the related literature is an integral part of the survey since it outlines prior contributions on the topic of interest. As a consequence, a correct information retrieval methodology is required to ensure complete surveying and inclusiveness. The related literature and its retrieval methodology are discussed next.

7.2.1 Information Retrieval Methodology

We present a comprehensive survey on the work which has been done in the area of blockchain-enabled TSMs. This should clarify the view of the TSM as a concept and give an overview of the main building blocks that TSM's architecture comprise.

The survey information was retrieved using *database search* in combination with *snowballing* method [13]. The sources of information are bibliographic databases *Scopus*[2], *Web of Science*[3], IEEE *Xplore*[4], *ACM Digital Library*[5] and *Google Scholar*[6]. To create the search strings a number of keywords were used: *telecommunication, marketplace, blockchain, service, identity management, assurance, governance, business settlement.* Also, multiple variations of these words were constructed such as plural forms and different word combinations.

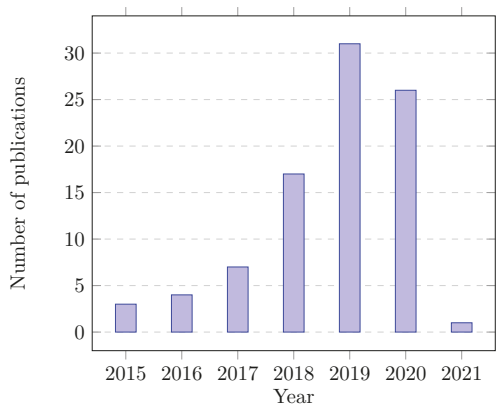


Figure 7.1: *Number of related publications per publication year.*

The search string that was used for TSM survey search in all bibliographic sources is: *blockchain AND telecommunication AND marketplace AND service AND survey*; (where certain parts of search string were excluded to increase the variety of search results). The main search criteria for related publications is the presence of discussion on blockchain technology and its applicability in the context of telecommunication services marketplaces. According to the search we conducted, there are no surveys at this point that target specifically TSMs based on blockchains. As a consequence, our search was extended to works that discuss the advantages and disadvantages of blockchain applicability in the context of digital marketplaces at large. In addition, the works which investigate digital marketplaces' core blockchain-based services were included as well. Fig. 7.1 presents a number of found related publications per publication year.

[2] https://www.scopus.com
[3] https://clarivate.com/webofsciencegroup/solutions/web-of-science
[4] https://ieeexplore.ieee.org
[5] https://dl.acm.org/
[6] https://scholar.google.com

7.2.2 Related Works

Despite no surveys found, a number of scientific proposals were discovered which discuss the idea of TSM. The work presented in [14] has shown a high degree of relevance for our own survey as its authors propose to use blockchain technology for the creation of a decentralized marketplace for telecommunication services. The proposed marketplace allows the entities collaborating within it to conduct business transactions without a need for a trusted third-party. Specifically, the authors describe the use-case of network infrastructure resource sharing, which nowadays involves trusted third-party and is a multi-step time-consuming process. Authors claim that blockchain technology can help to automate this process, enabling fast and efficient network resource sharing. Additionally, authors of [15] propose a blockchain-based system to manage SLAs between small cell providers (SCP) and mobile network operators (MNO). According to the authors, using blockchain smart contracts enhances the process of SCPs participation in the cellular market, as they can offer their capacity to MNOs in an automated and cost-efficient manner. These works have a high degree of relevance for this survey as they describe telecommunication network use-cases specifically and aim to enhance business settlement mechanisms between different CSPs.

7.2.3 Related Work

Despite no surveys that match the topics were found, a number of scientific proposals were found which discuss the idea of TSM. We highlight next the major contributions for decentralized marketplaces in the context of telecommunication services.

The work presented in [14] has shown a high degree of relevance for our own survey as its authors propose to use blockchain technology for the creation of a decentralized marketplace for telecommunication services. The proposed marketplace allows the entities collaborating within it to conduct business transactions without a need for a trusted third-party. Specifically, the authors describe the use-case of network infrastructure resource sharing, which nowadays involves trusted third-party and is a multi-step time-consuming process. Authors claim that blockchain technology can help to automate this process, enabling fast and efficient network resource sharing. Additionally, authors of [15] propose a blockchain-based system to manage SLAs between small cell providers (SCP) and mobile network operators (MNO). According to the authors, using blockchain smart contracts enhances the process of SCPs participation in the cellular market, as they can offer their capacity to MNOs in an automated and cost-efficient manner. These works have a high degree of relevance for this survey as they describe telecommunication network use-cases specifically and aim to enhance business settlement mechanisms between different CSPs.

7.2.4 Related Surveys

Considering that no surveys on specifically telecommunication marketplaces were found, we decided to incorporate surveys that explore the possibility of blockchain-based digital marketplaces at large.

The references provided next do not necessarily concentrate on a discussion of blockchain-based digital marketplaces. However, they provide some interesting insights on the concept

of the marketplace and the advantages and disadvantages of blockchain incorporation. In [16] authors discuss the blockchain application in a context of Smart Cities [17], where different aspects of citizens' life can be improved with the decentralized nature of blockchain. In terms of marketplaces, the authors discuss an application of blockchain in the context of Smart Grids and peer-to-peer energy trading. They assert that blockchain technology can enhance users' independence in an energy trading market, and allows to reduce the need for a trusted third-party presence in today's trading process. Authors of [6], through a case-study approach, provide a comprehensive description of digital marketplaces at large and provide an insight into the benefits of blockchain technology incorporation. According to them, the decentralized nature of blockchain technology can enable new forms of collaboration in digital marketplaces, as well as transform the existing process of business settlement. In [18] authors survey the research proposals on blockchain incorporation in the area of Internet of Things (IoT) [19], exploring the idea of blockchain-based IoT data marketplaces. In such marketplaces, blockchain technology acts as an enabler of data assurance, while IoT device's data is traded within a decentralized market in a trusted and secure manner. The author of [20] discusses the application of blockchain technology in the area of IoT data exchange in decentralized environments. In this work, the author explores the legal aspect of blockchain technology and its compliance with existing regulations in the area of information technology, and digital marketplaces in particular. In contrast to previous works, the author warns that the use of blockchain technology may harm the privacy of IoT device users, instead of enhancing it. In [21] authors conduct a comprehensive survey on the scientific and industrial proposals in network infrastructure resource sharing techniques. However, the authors provide very little insight into blockchain incorporation for the performance of resource sharing and the creation of market platforms.

Considering the information provided in the above surveys, the blockchain application in the area of digital marketplaces has gained traction and has been rather well defined. We aim to extend the application of blockchain technology to the TSM, by describing the needs of such a marketplace according to recent proposals and standardization activities. In addition, we describe a framework to enable CSPs to collaborate and conduct the business transaction execution.

7.3 Blockchains and Digital Marketplaces

It is important to provide an overview of technologies that are central to our survey. The discussion of the blockchain establishes a common understanding of this technology and helps to comprehend its features. In order to put blockchain into the context of telecommunication services, the applicability in the inter-communication service provider (inter-CSP) transactions is also discussed. Next, digital marketplaces are discussed at large, to establish a common understanding of this concept and the details behind it, with the survey of the proposals in blockchain-based digital marketplaces, to map the academic and industrial developments in this area. Finally, the definition of TSM and its core services is provided, to explain the concept and put it into the context of main application use-cases.

7.3.1 Blockchain Technology

Distributed Ledger Technology (DLT) has gained attention due to its decentralized nature and trust-enabling capabilities. DLT provides distributed data storage. It acts as a decentralized database where data is transmitted in a P2P network, thus, it does not have a central governing authority and all the security concerns that come with it, *c. f. ,* [22]. Information in the ledger is replicated on every node in the P2P network, which prevents data loss. In addition, due to the immutable nature of the ledger, it is extremely difficult to alter transaction history. DLT provides a lot of benefits, such as provenance, accountability, and transparency for all the data which is stored on a distributed ledger [16]. Blockchain is one possible implementation of DLT. It bundles the pieces of data into blocks, where each block contains a reference to the previous one, thus, forming a chain of data blocks. Another structure that is used to implement DLT is Direct Acyclic Graph (DAG) [23]. In DAG-based DLTs, the newly added transaction can reference multiple previously added ones. IOTA [24] is the representative of DAG-based distributed ledger implementations. Further in this work, we survey proposals that utilize both blockchain-based and DAG-based DLTs in the context of digital marketplaces. However, in this section we discuss blockchain technology exclusively. The reasoning is that majority of academic proposals use the blockchain implementation of DLT, and only a few use DAG-based DLT.

The architecture of the blockchain depends on two things: 1) whether the access for reading the information stored on blockchain is public or private, and 2) whether the right to write to the ledger and participate in consensus protocol execution is permissioned or permissionless. There are three main blockchain architectures [16, 25]:

7.3.1.1 Public permissionless blockchain

In this architecture, everyone is allowed to become a part of the network and participate in the consensus process. Every node carries a copy of the shared ledger. The transactions are visible for all blockchain nodes, but participants retain a certain degree of anonymity, which may be subject to privacy issues [26]. The Bitcoin [7] is the first and well-known blockchain technology implementation that utilizes public permissionless architecture. It is also the first cryptocurrency and was launched in 2009 after being introduced by Satoshi Nakamoto in 2008. It is mainly used as a decentralized financial system, where token exchanges emulate banking transactions. Next, Ethereum [27] is another representative of public permissionless architecture. It is believed to be an evolutionary step of Bitcoin since it aimed to solve some of Bitcoin issues such as flexibility of on-chain code execution.

7.3.1.2 Private permissioned blockchain

Here it is the governing node (or set of nodes) that decides whether a new participant can enter the blockchain network. Moreover, after the new node has gained access to read the ledger, the governing node decides whether it is allowed to participate in consensus. The decision on the ability to participate in consensus for the existing members can also be reviewed by the governing node during the operation of the blockchain network. The main idea of private permissioned blockchain architecture is to fully control the access to different aspects of blockchain network operation. The governing node can be also represented by a

regulatory authority which issues private blockchain participation licenses and helps to sign business agreements between participating stakeholders to carry out consensus process [28]. Hyperledger Fabric [29], which is developed by the Linux Foundation, is a representative of a private permissioned blockchain system. In Hyperledger Fabric, the nodes are divided into three types based on the task they are performing: endorsement, ordering, or validation. Endorsing nodes take a transaction proposal, execute it and return a transaction proposal response. Responses from multiple endorsers are then bundled together and then passed to the ordering nodes. These nodes take newly endorsed transactions and agree on the order in which these transactions are stored in the ledger. Finally, validation nodes receive the block that was newly added to the blockchain and check the validity of the transactions in that block. They check that each transaction has received all the endorsements it needed based on the configured policy and that it is not conflicting with a previous transaction. Invalid transactions are kept in the blockchain but do not modify its state.

7.3.1.3 Public permissioned blockchain

This blockchain architecture allows initially non-trusting organizations to establish a trust bridge over a public yet permissioned system. In public permissioned blockchain architecture, everyone is allowed to join the blockchain network, thus obtaining the right to read and verify the state of the ledger, as well as propose new transactions. However, only authorized nodes have the possibility to participate in the consensus process. This type of architecture also presents a possibility for only a specific group of nodes to write new blocks to the ledger. It creates an opportunity for the creation of consortium-governed ledgers, where a number of companies share blockchain's governance, maintenance, and orchestration. This type of architecture was made popular by the Sovrin Foundation [30] in their blockchain-based identity management system implementation which is discussed in Section 7.4.1.

7.3.2 Blockchain Infrastructure Model

In order to provide a rather familiar structuring of blockchain infrastructure, we provide a model derived from [16]. The entire blockchain infrastructure is divided into six layers which are shown in Fig. 7.2. The layering approach is used as a way to divide the infrastructure into a set of blocks with the underlying components on the inside presented as technologies and processes used in blockchain operation. Here, we discuss each infrastructure layer and its components.

7.3.2.1 Data Layer

The first layer in the blockchain infrastructure model is the data layer. It presents a fundamental set of technologies that lay at the core of blockchain. The *blockchain* has received its name due to the resemblance of a chain, where instead of metal rings the blocks of structured data are interconnected in a sequence. This data is structured chronologically and is immutable, *i. e.* , it is highly challenging to alter on-chain data. A blockchain with a detailed structure of a block [31] is shown in Fig. 7.3. The *block body* is used to store hashes of transactions that are verified and embedded in the block. These hashes are built as a

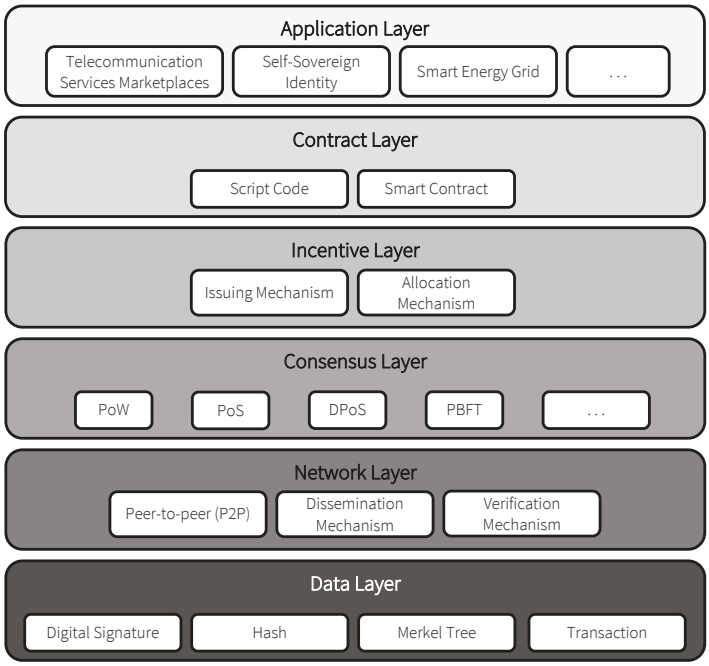


Figure 7.2: *Blockchain infrastructure model [16].*

Merkle Tree [32] and represented in the block header as a *Merkle Root Tree* component. The Merkle tree comprises a binary tree constructed of hashes of transactions that are stored in a block body and positioned at the beginning of the tree structure, *e. g.,* Tx1. When the hash of the transaction is computed, *e. g.,* H1, it is being added with the neighboring hash up until the moment when the top of the tree is reached. Since every computed hash is saved in a block, the Merkle tree can be later used for rapid and secure verification of the transactions included into the block [16]. The *block header* plays a key role in chain establishment since it contains the hash of the previous block in a sequence, which is called a *parent block*. The first block in a sequence is called *genesis block* and it does not have a parent block. The block header contains some additional metadata information such as the block owner's signature and timestamp of the block creation.

7.3.2.2 Network Layer

The network layer topology in permissionless blockchains is built and functions similarly to a P2P network [33]. The P2P network ensures no privileged participants partake in the life-cycle of blockchain events. The main events of the blockchain network are the dissemination, *i. e.,* forwarding, and verification of the transactions according to the network layer protocols. A distinct feature of the blockchain network layer is that it ensures that only verified transactions are transmitted in the distributed network and stored in the local node's ledger. First, the dissemination mechanism utilizes the distributed nature of the P2P network and broadcasts transactions to neighboring nodes. Second, the verification mechanism ensures

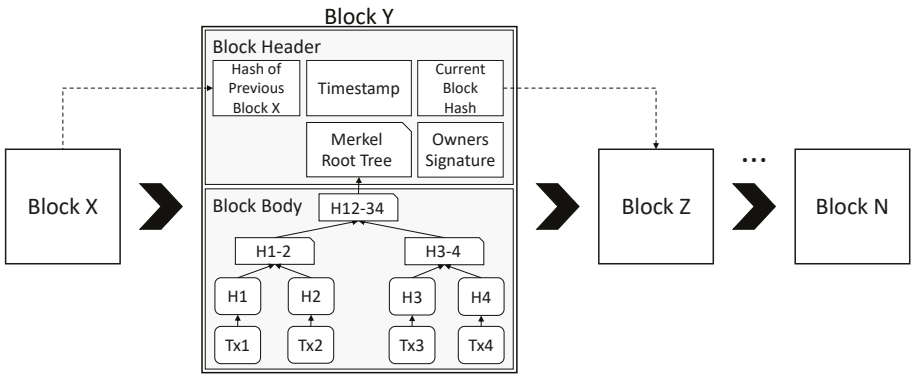


Figure 7.3: *Blockchain block structure [16].*

that only valid transactions continue to be forwarded by verifying transactions according to blockchain specifications. The verification itself is based on asymmetric cryptography where each node maintains a public and private key pair [34]. When a transaction is created, it is signed by the private key of the creator node, and then broadcasted to neighboring nodes. Meanwhile, neighbors use the public key of the creator node, to verify the transaction's signature [16]. If a transaction is valid, it is forwarded to other neighboring nodes. Otherwise, if the transaction is marked as invalid, forwarding is stopped, and the transaction is discarded.

In contrast, the blockchains with permissioned architecture are not necessarily a P2P network. Permissioned architecture frequently incorporates multiple interconnected blockchains and in some use-cases the peers from different organizations do not really communicate with each other. For instance, in Hyperledger Fabric the blocks distribution is reliant on the ordering node for providing blocks to the leader peer from each organization. The leader peer is then responsible for redistribution of the ordered blocks to the rest of the peers.

7.3.2.3 Consensus Layer

In centralized systems, the consensus is an inherent feature of the system, since all components are orchestrated by a centralized trust enabling entity. In contrast to centralized systems, where all the nodes are governed by the central silo which represents the root of trust, the blockchain network deliberately avoids centralized authorities, making the system decentralized. With this, a mechanism that allows establishing consensus between all nodes is needed to ensure secure and correct decentralized blockchain network operation.

At the present time, a number of consensus algorithms are used in blockchain systems. The Bitcoin blockchain uses Proof of Work (PoW) [7] consensus protocol where nodes in the blockchain network continuously execute hash calculations until the computed hash is less than a given target value. The first node to generate a correct hash obtains the ability to write the next block to the blockchain. The Proof of Stake (PoS) [35] consensus protocol was made popular by the Ethereum cryptocurrency and was developed as an alternative to PoW. In PoW, in order to generate a valid hash value, the entire network competes, thus, by design, consuming large amounts of electricity. PoS is designed to be energy efficient and gives the opportunity to add new blocks to the ledger to the nodes which hold the largest

amounts of cryptocurrency. Moreover, for each block, the actual node is selected with a certain degree of randomness. A Delegated Proof of Stake (DPoS) [36] consensus protocol was designed as an evolution of PoS. DPoS makes the blockchain network more democratic and gives every node an opportunity to decide what is being written to the blockchain. The downside of DPoS is that still the votes of the nodes which have the most cryptocurrency, weigh the most. A Practical Byzantine Fault Tolerance (PBFT) [37, 38] consensus protocol is designed to tolerate Byzantine faults in a system where the data is being replicated. For a deeper discussion on consensus mechanisms in blockchain systems, the reader is referred to a survey on consensus protocols [39].

7.3.2.4 Incentive Layer

The incentive layer combines the mechanisms to issue and allocate portions of cryptocurrency to nodes that participate in the data verification process. The cryptocurrency, *e. g. ,* Bitcoin or Ether, works as an incentivizing factor for blockchain network participants, as far as when awarded, it can be spent in the network, or exchanged to fiat currencies. In Bitcoin cryptocurrency, which is built as permissionless blockchain, the incentive *issuing mechanism* is called "mining". Comparison to the mining process comes from the fact that in order to get some precious metal or stone it has to be "mined" from the earth. The process of mining involves the nodes in the blockchain network spending their computational power to verify the next hash in a sequence of blockchain to take part in the PoW consensus. The more computational power the node has, the bigger incentive is allocated. In Bitcoin cryptocurrency, miners can unite into *mining pools*, where large computational "farms" are used to mine large amounts of cryptocurrency. According to *allocation mechanism*, economic incentives are provided to the node which generated a new block. When computational efforts are registered by the blockchain network, the generator node gets a portion of cryptocurrency allocated to its crypto-wallet. The incentive layer represents the attractiveness of the blockchain network, as far as the more rewarding incentive is for the miners, the more nodes are attracted to join the network and contribute to the general pool. Also, diversification of the miners allows for a more secure blockchain, thus, reducing the possibility for a 51% attack [40], where more than 50% of the miners are malicious and can perform consensus faster than honest miners. This allows malicious miners to control the blockchain network and to double-spend the cryptocurrency.

7.3.2.5 Contract Layer

The contract layer of the blockchain infrastructure model introduces the way to embed executable code into the transaction. The primary way to execute on-chain code was introduced in Bitcoin cryptocurrency as a *script* which is embedded into a block. A script is stored on the immutable ledger in an individual block and is based on a limited programming language that states the conditions to validate a transaction, and acts as a termination guarantee in case of transaction conditions are not met. A script has a limitation, namely, it does not have the possibility to execute more complex transaction scenarios. A *smart contract* is considered to be an evolution of a script. It was first introduced in Ethereum blockchain as the way to make cryptocurrency transactions more flexible and complex. It is

based on a programming language that is Turing complete and allows shifting from static transactions to the execution of code. A smart contract allows bringing a certain degree of programmability to the blockchain, thus, introducing the ability to describe cryptocurrency exchange scenarios of different complexity. In the context of business transactions, a smart contract is used to describe the conditions under which involved actors are collaborating. When all conditions of a business transaction are agreed upon, they are embedded into a smart contract and signed by collaborating actors. Next, the smart contract is verified by the blockchain network and written to the ledger [41]. Depending on the platform there might be an explicit signature by the involved parties. In some cases, the code of the smart contract is freely auditable, and just usage of the smart contract is equivalent to accepting the way it is written. When the conditions of the smart contract are met, it executes the code embedded inside of it, while acting as a guarantee of exactly what code is being executed as well as on what data it is operating. As a result of this, the smart contracts allow to automate complex business transaction scenarios, making them more error-resistant and time-efficient. Finally, as far as smart contracts can be executed by any member of the blockchain network to verify the validity of the data it is operating, this contributes to the transparency and trust establishment between smart contract actors. For a deeper discussion on smart contracts, the reader is referred to empirical studies of blockchain smart contracts at [41] and [42].

7.3.2.6 Application Layer

The top layer in the blockchain infrastructure model is the application layer. It aims to introduce different application scenarios for blockchain technology. The main application scenario discussed in this work is the Telecommunication Services Marketplaces, but there are multiple others such as Smart Energy Grid, IoT, Cloud Infrastructures, Self-Sovereign Identity, etc. [43]. The main aim of these applications is to enhance different aspects of business and social life such as enhancement of business settlement and augmentation of digital sovereignty. Nowadays, when the best application conditions of blockchain technology are yet to be found, we see that this topic is being researched by multiple academic communities as well as adopted by multiple companies in the industry sector. The sheer volume of academic works on blockchain technology generated in recent years gives an idea of the interest in the topic.

7.3.3 Reasoning for Blockchain Usage by CSPs

Existing operational frameworks of CSP are built on the premise that the entire telecommunication services chain belongs to one CSP. The *interoperability* of such operational frameworks is not always considered, as well as the system is centralized and governed by the owner company. Furthermore, existing operational frameworks are slow to adapt to the needs of next-generation internet, as *integration* of new technologies with the legacy systems is challenging and time-consuming. Transaction execution between two or more CSPs involves manual operation processes which can be complex, expensive, and time-consuming. As these processes involve human intervention, they are a subject of multiple issues: manual errors, long payment cycles, and exposure to fraud. In this way, *accountability* and *trust* of the operational framework are jeopardized which may lead to consistent revenue losses [9].

With the creation of a unified framework to operate telecommunication service chains, the development of next-generation network services will be accelerated. This will also ensure the interoperability and integration ability of new services with legacy systems. Furthermore, an automated approach to handle inter-CSP transaction execution will enable real-time and trusted settlement between two or more CSPs [9].

The DLT is highly applicable to inter-CSP business settlement transactions. It allows automation of inter-CSPs processes, thus eliminating the need for any manual human efforts. In this case, the DLT acts as a *trust-enabling* entity which under agreed rules, defined in smart contracts, does not need a trusted *third-party* to take care of parts of the transaction. While all inter-CSP transactions are recorded in the DLT's storage (every CSP can verify transactions or smart contract data at any time) the data stored on DLT is immutable and the storage itself is distributed. Private permissioned blockchain architecture has the highest applicability in the use-case of CSP business settlement transactions. The ledger where trusted parties authenticate and are authorized to verify the business agreement recorded in a smart contract at any time, enables trusted, secure, and automated business settlement for two or more CSPs [44].

Automation of inter-CSP business settlement processes benefits the business revenue growth, decreases the duration of transaction execution, and reduces the costs spent per business contract settlement [9, 44, 45].

A general risk from blockchain-based marketplaces for CSPs is that these platforms open the system for collaboration and may impact their functional integrity (incl. privacy violations) and business model. However, if trusted collaboration is enabled then the advantages of expanding a system by including additional stakeholders (developers) and new functionality leading to increased revenue, may outweigh the disadvantages of marketplace platforms, which are integrity checking and sharing revenues. Sharing revenue, however, may spark the discussion on how profits are taxed now in such a context. We believe that this discussion is important but would deviate too much the paper from its main objective to provide an understanding of the techniques to implement blockchain-based marketplaces.

7.3.4 Centralized Digital Marketplaces

Before we survey the blockchain-based digital marketplaces, we discuss centralized marketplaces that are being used nowadays. Today's marketplaces pose a number of challenges in terms of operations and fairness towards users. These challenges are discussed next.

Digital marketplaces are a common and widely accepted concept for the formation of business opportunities. They are open platforms where IT companies or individual developers can offer their products for purchase. The timeline of application and cloud marketplace development initially described in [46] is depicted in Fig. 7.4. In general, digital marketplaces are defined to meet the requirements of the concepts of *supply and demand* [4]. The popularization of smartphones, for example, created a demand for apps, which led the main mobile phones vendors to deploy their marketplaces as a way to supply applications to end-users.

The marketplace allows products to be supplied to customers with increased speed and stimulates the popularity and expansion of the software. Having fulfilled the supply and demand capabilities of the marketplace, the business relations inside of the marketplace

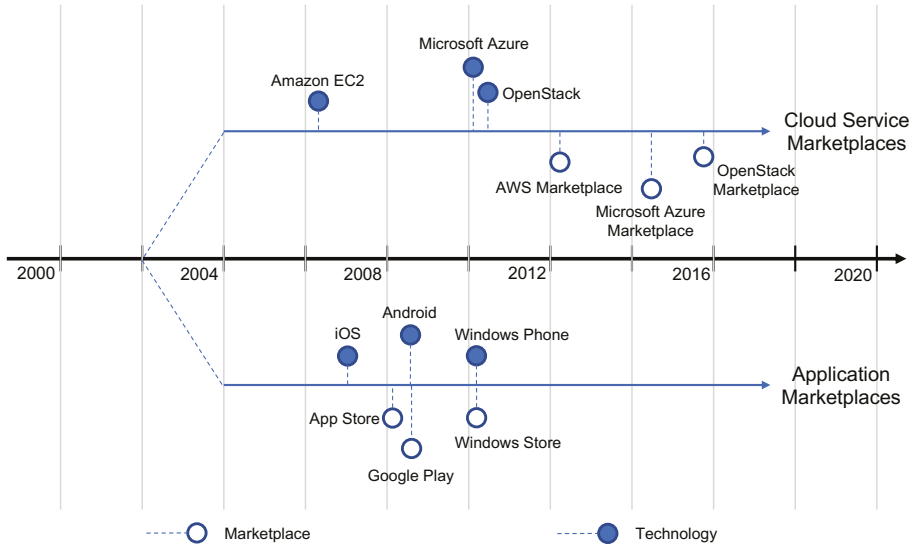


Figure 7.4: *Timeline of application and cloud marketplaces development [46].*

must be regulated. This is done through the *licensing approach* [47] when the relation of the consumer and supplier is defined in a document, *i. e.*, license, which is signed by involved parties. The licensing approach enables the marketplace to execute business operations, and makes the process transparent and legally correct. The supply concept also provides a *payment* [48] for a software product, which is charged by the marketplace on behalf of the supplier. It acts as a foundation of the *business settlement*, *i. e.*, business transaction execution, between the consumer and the supplier. In a marketplace with a centralized architecture, being the middleman in software distribution gives the ability to dictate the billing rules and payment distribution the majority of the time.

Digital marketplaces operate as a provider of specific foundation services to allow optimal operation for their customers. Starting with the *identity management* system [49], a marketplace acts as a provider of digital identity which authenticates customers in a system and authorizes the execution of an allowed set of actions. Moreover, acting as an *assurance provider* [50] towards the customers, the marketplace provides a certain degree of confidence in the services and platforms which are provided by it. As a main distributing entity, it has to provide a certain degree of *trust* [51] to make customers feel confident about payment transactions. For a system to be properly operated the *governance* over the marketplace has to be maintained by one or a number of trusted parties.

Another major part of the marketplace concept is how the software products are delivered to the customers, *i. e.*, executed, fulfilling their computational purpose. In today's marketplaces, there are two main types of software delivery: 1) *On-premise*, when the software is executed on customers hardware infrastructure, and 2) *In-cloud*, when cloud hardware resources are used. These two software delivery models will be discussed next.

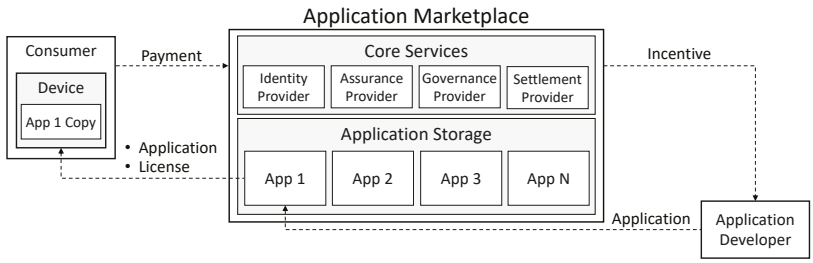


Figure 7.5: *Application marketplace with on-premise application delivery model.*

7.3.4.1 On-premise delivery model

The widely known *Application Marketplaces* (AM) [52] such as *Apple App Store* and *Google Play* have gained their popularity by providing numerous applications for their respective *iOS* and *Android* operating systems (OS) [53]. These OSs are installed on a wide variety of personal devices, which nowadays carry a substantial amount of computing power. The AMs define rather strict distribution rules for the applications they provide. The applications for some of the operating systems can officially be provided only through the respective marketplace. Moreover, the application itself is allowed to be executed only within a certain operating system. This limits the developers of the applications in terms of the number of different marketplaces where they can distribute their products. Also, developers are limited in the tools that they can use to develop their applications since every operating system acts as an execution environment for a certain runtime. On the other side, when the marketplace is bound to a specific OS, it also can act as an *assurance provider*, who guarantees the proper execution of the application. For example, applications on the Apple App Store go through a multi-step verification process before becoming available for the customers. Also, the AMs are built around a centralized server which acts as a *trust provider* between the customers and application developers. Being in the middle, the marketplace has the opportunity to dictate the rules on billing, *e. g. ,* taking a percentage of the profits as the main distributor.

The *on-premise* delivery model is intrinsic for AMs due to their *consumer-oriented* approach. The structure of this delivery model is shown in Fig. 7.5. In the on-premise model, the applications are delivered as software packages with the aim to perform a range of tasks. The specifics are in the behavior of software that is being distributed - it works as a stand-alone program, and not as a part of a Service Function Chain (SFC) [54], which can be defined as a sequence of software components that build the service chain and interact with each other to reach a common goal. Therefore, AMs are aimed to fulfill the demands of a single consumer, and the software distributed by it is not intentionally designed to be a part of SFC. Although the end-devices with mobile operating systems can become a part of SFC, it is not the aim of the products which are mainly distributed by the AM.

7.3.4.2 In-cloud delivery model

The major public cloud infrastructure providers, such as *Amazon Web Services* and *Microsoft Azure*, have recently introduced a new type of digital marketplaces: *Cloud Services Market-*

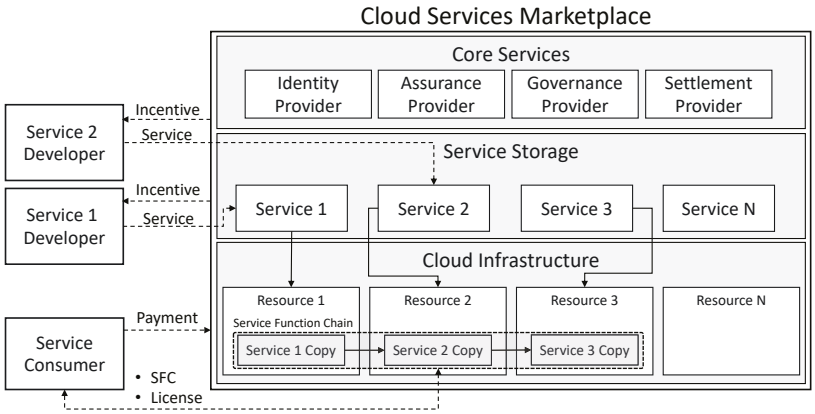


Figure 7.6: *Cloud service marketplace with in-cloud service delivery model.*

places (CSM) [55, 56]. The cloud services are not bounded to specific software requirements, on a contrary, they are designed to be software- and OS-agnostic. The only limitation is that the cloud service has to be executed within the hardware infrastructure of the cloud provider. Much like the AM, the CSM is also built with a centralized server representing the trust provider. Also, while all services are executed inside of the cloud infrastructure, the CSM provider acts as an assurance provider towards the customers. Unlike the AM, CSM is designed to distribute services that can become a part of the SFC. SFCs can be executed simultaneously - forming a grid of services, or in a sequence - forming a service pipeline.

The *in-cloud* delivery model is intrinsic for the CSMs due to their service-oriented approach. The structure of this delivery model is shown in Fig. 7.6. In the case of CSMs, the services play a key role in forming the SFC. Within SFC, it is important to fulfill minimum hardware requirements for some specific service, *e. g.*, Trusted Platform Module (TPM) or specific CPU architecture, which cloud infrastructure can provide. SFCs can be used in application development as an execution environment for application builds generation in version control systems. Also, SFCs can execute so-called *Artificial Intelligence (AI) pipelines* which are used in a collaborative form of AI engineering [57]. An important feature of the in-cloud delivery model is that it introduces the ability for multiple stakeholders to interconnect their services deployed within the same cloud infrastructure provider. This opens the opportunity for a new type of collaboration when a business settlement is performed between multiple stakeholders. However, these stakeholders still need to rely on a trusted third-party, *i. e.*, cloud provider, which acts as a trust provider towards the business settlement execution.

Today's cloud providers are limited in providing business settlement interfaces for inter-cloud multi-stakeholder operations due to a number of reasons. To begin with, cloud infrastructure providers do not use a unified API standard for the provisioning of different cloud service models such as IaaS, PaaS, and SaaS. Next, identity management systems are built with a centralized model which makes multi-cloud provisioning challenging, as every cloud infrastructure uses a separate identity. The interoperability of services provided in CSM is not considered outside of the boundaries of a specific cloud provider. In addition, the assurance provided by CSM is centralized within one cloud provider, which makes it

challenging for every CSM involved in SFC construction to guarantee reliability and stability of its execution. Lastly, the existing monitoring frameworks are mostly designed to work with a specific cloud provider which makes the multi-cloud system maintenance costly and time-consuming. The authors of [58] provide an insight on this issue and propose a way to enable a multi-cloud provisioning system.

7.3.5 Blockchain-based Digital Marketplaces

In recent years, to address the disadvantages of centralized marketplaces, the concept of a decentralized marketplace has been introduced. Coupled with the widespread application of blockchain technology, it resulted in a number of scientific proposals exploring blockchain-based digital marketplaces. These proposals are listed in chronological order in Table 7.1 along with their *application area* and are discussed in more detail in the remainder of this paper. Additionally, a number of industrial proposals on blockchain-based marketplaces are presented in Table 7.2.

7.3.5.1 Smart Energy Grid

The application of blockchain in the area of Smart Grid [59] and peer-to-peer (P2P) energy trading has gained traction in recent years. Blockchain technology reduces distributed energy prosumers' dependence on the energy supplier during trading process and enables P2P energy trading using smart contracts as a tool for trade settlement. In [60], authors investigate a possibility of blockchain-based marketplace creation for P2P energy trading to bring flexibility and transparency to all actors involved in the energy market. In the proposed model, a blockchain holds the amount of electricity produced and allows regulating the electricity prices based on the prosumer generation rate. Authors of [61] propose a decentralized blockchain-based platform for energy trading. With blockchain technology, authors are able to reduce the need for a trusted intermediary in a trading process. The main aims of the platform are to make the usage of the energy generated by households more efficient and reduce electricity bills.

Both aforementioned proposals can be summarized in two main reasons to apply blockchain in the context of P2P energy trading. First, it is the reduction of the need for a third party, *e. g.* , a marketplace, in an electricity trading process. Eventually, the third party cannot be eliminated entirely since the electricity prices are regulated by the government, which in turn places a restriction on the maximum price that energy can be sold for [62]. However, blockchain technology enables P2P trading between marketplace customers, where different trading concepts can be applied and enhanced, *e. g.* , auction bidding or fixed price selling. Second, the ledger as a data storage provides statistical information which helps marketplace maintainers to dynamically adjust electricity prices during the day. This enables making trading more efficient in relation to energy consumption and production rates and possibly allows to reduce electricity bills for households.

Table 7.1: *Academic proposals taxonomy on blockchain-based marketplaces*

Reference	Application Area	Platform	Architecture	Description
N. Afraz et al. [14]	Telecommunication Services	Hyperledger Fabric [29]	Private Permissioned	A network infrastructure resource sharing in a blockchain-based decentralized marketplace.
E. D. Pascale et al. [15]	Telecommunication Services	Ethereum [27]	Public Permissionless	A decentralized system to manage SLAs between SCP and MNO with blockchain smart contracts.
C. Pop et al. [60]	Smart Energy Grid	Ethereum	Public Permissionless	A decentralized marketplace for P2P energy trading with blockchain smart contracts addressing enerfy flexibility.
S. Saxena et al. [61]	Smart Energy Grid	Hyperledger Fabric	Private Permissioned	A decentralized blockchain-based platform for energy trading with increased efficiency.
G. S. Ramachandran et al. [63]	IoT	Ethereum	Public Permissionless	A test implementation and discussion of main challenges for decentralized IoT marketplaces advancement.
L. Mikkelsen et al. [64]	IoT	Ethereum	Public Permissionless	A concept of blockchain-based marketplace which distributes services and storage resources for IoT.
D.-D. Nguyen et al. [65]	IoT	Not Applicable[1]	Public Permissionless	A decentralized blockchain-based IoT data marketplace with the ability to consider data's location and supplier availability.
H. T. T. Truong et al. [66]	IoT	Hyperledger Fabric	Private Permissioned	A framework for a decentralized IoT data marketplace that stores access control policies and makes access controlling decisions.
S. Bajoudah [67]	IoT	Ethereum	Public Permissionless	A model for a blockchain-based decentralized IoT data trading marketplace that provides a trade-off between transaction costs and data loss risks.
K. R. Ozyilmaz et al. [68]	IoT, Sensors Data	Ethereum	Public Permissionless	A blockchain-based decentralized marketplace where for IoT data trading where developers of ML solutions can collaborate.
P. Tzianos et al. [69]	IoT, Sensors Data	IOTA [24], Blockchain agnostic[2]	DAG-based	A blockchain-based marketplace for IoT sensor data trading with the blockchain agnostic architecture.
S. Musso et al. [70]	IoT, Sensors Data	IOTA	DAG-based	A decentralized DLT-based marketplace designed to trade streaming data in a context of smart cities.
K. Nguyen et al. [71]	IoT, Sensors Data	Ethereum	Public Permissionless	A blockchain-based marketplace to search and trade IoT data based on the geographical location of the device.
A. Seitz et al. [72]	IoT, Applications	Ethereum	Public Permissionless	A blockchain-based marketplace with the ability to trace application installation on edge IoT devices.
D. Miehle et al. [73]	IoT, Supply Chain	Hyperledger Fabric	Private Permissioned	A blockchain-based marketplace model where machines perform full chains of tasks to supply manufacturers with needed details.

[1] No implementation available.
[2] IOTA DLT is used for data storage. Overall system is not designed to work with any specific DLT in mind and aims to be blockchain agnostic.

Reference	Application Area	Platform	Architecture	Description
V. P. Ranganthan et al. [74]	E-commerce	Ethereum	Public Permissionless	A decentralized e-commerce marketplace based where merchants fully control transaction process.
Z. Wang [75]	E-commerce	Ethereum	Public Permissionless	A blockchain-based marketplace for art trading that allows to trace the art assets owner and location history.
J. Martins [76]	E-commerce	Ethereum	Public Permissionless	A model of blockchain-based e-marketplace where suppliers compete with each other to fulfill the customer order.
N. Baranwal Somy et al. [77]	Cloud Services, Data Trading	Hyperledger Fabric	Private Permissioned	A blockchain-based decentralized marketplace where different actors can collaborate in AI engineering process.
J. Li et al. [78]	Cloud Services, Data Trading	Bitcoin[3][7]	Public Permissionless	A model for blockchain-based decentralized marketplace for online content trading with indexing of content names.
P. Banerjee et al. [79]	Cloud Services, Data Trading	Hyperledger Fabric	Private Permissioned	A decentralized blockchain-based marketplace for online content trading which provides a searching and trading mechanisms.
M. F. Franco et al. [80]	Cloud Services, VNF	Ethereum	Public Permissionless	A blockchain-based marketplace model for VNFs hosting where infrastructure providers compete to host VNF.
B. Nour et al. [81]	Cloud Services, VNF	Custom PoW-based blockchain	Public Permissionless	A blockchain-based brokering mechanism used to allocate and manage network slicing in 5G network.
E. Scheid et al. [82]	Cloud Services, VNF	Ethereum	Public Permissionless	A blockchain-based VNF package repository where package integrity is verified without the involvement of a third party.
M. Franco et al. [83]	Cloud Services, VNF	Ethereum	Public Permissionless	A blockchain-based catalog where vendors of distributed denial of service protection software can post and sell their products.
V. Arya et al. [84]	Cloud Services, API	Ethereum or Hyperledger fabric[4]	Multiple Architectures	A blockchain-based marketplace model for Artificial Intelligence APIs access trading.
M. Pincheira et al. [85]	Cloud Services	Ethereum	Public Permissionless	A decentralized blockchain-based marketplace for Fog/Edge computing resources trading.

[3] Proposed blockchain is based on the Bitcoin blockchain, with significant modifications.
[4] No implementation available, however authors plan to base future implementation either on Ethereum or Hyperledger Fabric.

7.3.5.2 Internet of Things (IoT)

In the area of IoT, blockchain technology enables new opportunities in the creation of decentralized data marketplaces. The IoT devices generate and exchange large amounts of data, which poses challenges in terms of privacy, data assurance, and scalability. However, the ability to trade the IoT data in a decentralized and democratic way creates an opportunity for IoT device owners to monetize their data and opens new business opportunities for IoT device manufacturers. In [63] authors explore the benefits of the blockchain technology incorporation into decentralized IoT data marketplaces. They present a test implementation of such a decentralized blockchain-based marketplace, where IoT data can be posted for further purchase. As result, the authors present such challenges for decentralized marketplace advancement as compliance with regulatory documents, *e. g.,* General Data Protection Regulation (GDPR) [86], identity management and trust establishment. Authors of [64] discuss the issue of central point of failure in centralized IoT marketplaces where devices rely on the availability of the marketplace to use remote services and storage. To address this issue authors propose the concept of the distributed blockchain-based marketplace. This enables the distribution of servers and storage resources, increasing the marketplace's availability and robustness. In addition, authors evaluate their test implementation in an experimental testbed which shows the transparency and operational ability of distributed marketplace, without the need for a trusted centralized entity. In [65], authors present a framework for decentralized blockchain-based IoT data marketplace. According to the authors, the novelty of their framework is the ability to consider such factors as data's location and supplier availability, which gives the buyer better data context. This opens the opportunities for the data collection tasks ordering, where suppliers fulfill the locational and contextual needs of data buyers. Authors of [66] propose a framework for a decentralized IoT data marketplace. According to the authors, the novelty of their proposal is that in addition to storing access control policy on the blockchain, it also makes access control decisions. This contributes to auditability of the marketplace and brings transparency to the marketplace participants. In addition, the authors also provide a possibility to settle the financial transactions on the blockchain by utilizing its on-chain currency. In [67] authors present a model for a blockchain-based decentralized IoT data trading marketplace. In their marketplace, authors enable the users to perform the full chain of trading operations, from the initial advertisement of the data to the final business settlement with payments delivered and legal contract signed. Authors claim, that usage of the blockchain smart-contracts enables initially non-trusted parties to conduct business settlements without third-party involvement providing a trade-off between transaction costs and data loss risks. Authors of [68] propose a blockchain-based decentralized marketplace for IoT data trading. In addition, the developers of Machine Learning (ML) solutions can collaborate within such a marketplace while using the IoT data for ML model training. According to the authors, usage of blockchain increases transparency and regulates access to the data traded within the marketplace. In [69] authors describe a DLT-based marketplace for IoT sensor data trading. The interesting part of the proposal is that authors use IOTA [24] distributed ledger as a data storage solution, while the overall marketplace is designed to be blockchain agnostic. Authors of [70] present a decentralized DLT-based marketplace designed to operate in the context of smart cities. According to the authors, their marketplace enables IoT devices data stream trading without the need

of a trusted intermediary. Authors claim that IOTA's distributed ledger is IoT-tailored with emphasis on system's scalability and trust. In [71] authors propose a blockchain-based marketplace for IoT data trading. This particular model of the marketplace allows to search and trade IoT data based on the geographical location of the device. According to the authors, with blockchain technology, they are able to mitigate such issues as accountability and correctness of geographical data and provide a platform that allows buyers to create more location targeted IoT data searches. Authors of [72] propose a blockchain-based marketplace with the ability to trace application installation on edge IoT devices. According to authors, the blockchain technology brings transparency and accountability into application trading and installation processes. In addition, authors also employ Augmented Reality (AR) [87], in order to enhance user experience during the application installation. In [73] authors propose a blockchain-based marketplace model where machines perform full chains of tasks to supply manufacturers with needed parts. While all trading decisions are recorded on the ledger, machines execute selection and ordering of the parts with a final trading settlement recorded in smart contracts.

All aforementioned IoT data marketplaces proposals share a number of common goals that they pursue when applying blockchain technology. First, data privacy has to be preserved according to legal regulations. For example, European GDPR poses rather strict regulations on the rights of users to rectify or remove the data from the storage of specific service, *e. g.,* marketplace. Thus, due to the immutability of the blockchain, any confidential data has to be stored off-chain with ledger storing only hashed references to real data locations. Second, due to the trust-enabling capabilities of the blockchain, it allows to eliminate the need for a trusted third-party and makes the data trading process more transparent and fair. The trade agreement conditions can be embedded into the smart contract and the contract itself can be executed automatically. It contributes to the value distribution balance since there is a possibility to reduce the price of smart contract execution in comparison to the involvement of third-party. Third, blockchain allows to securely store data access control policies which can be automatically enforced during and after trade settlement. Finally, the on-chain currency, *i. e.,* token or cryptocurrency, enables automated payment release to the seller according to contract conditions.

7.3.5.3 E-commerce

Blockchain technology has found an application in the area of e-commerce [88, 89] as well. It allows making marketplaces more democratic while acting as an instrument for distributed control over the users and merchants operating within the e-commerce platform. In [74] authors lay the foundation for e-commerce marketplaces based on Ethereum blockchain [27]. The common problems of today's marketplaces are in the lack of distributed instruments within the marketplace to make the trading process transparent and efficient. Today's marketplaces, being centralized systems, may block merchants at will, being the only entity to decide on such action, and fully control the process of the financial settlement between the customer and merchant while taking a portion of the payment to itself. According to authors, the blockchain technology helps to mitigate these issues by making the marketplace decentralized, thus, removing a middleman in the financial settlement process, providing an ability for merchants to fully control the transaction process and conduct an audit of the

data on an immutable ledger making the process transparent and secure. Authors of [75] present a blockchain-based marketplace for art trading. According to the authors, along with bringing such benefits of blockchain technology as transparency and data assurance of financial transactions, this platform also allows tracing the art assets owner and location history. Authors claim that their marketplace model is the first to address the task of art assets trading. In [76] authors approach the model of e-marketplace from the direction of the customer. They propose a blockchain-based e-commerce marketplace where customers make their orders and submit them to the platform. In turn, the suppliers make their bids on the order and compete with each other to fulfill it. The auction takes place on the blockchain which brings transparency and trust into the bidding process.

The e-commerce blockchain application contains a lot of similar goals which were described in the context of Smart Grids and IoT. However, the distinct feature of e-commerce blockchain-based marketplaces is the ability to provide fairness towards customers and merchants who operate within it. Since the data stored on a permissionless blockchain is open, all dispute resolution can be done in a transparent and fair way. Moreover, this distributes the merchants' or customers' blocking decision-making process, thus, removing authoritative control present in centralized e-commerce marketplaces. Also, the distributed blockchain storage permits indexation of products catalog, which makes search requests execution rapid and fair. Finally, the usage of blockchain in the bidding process makes it more transparent and tamper-proof.

7.3.5.4 Cloud Services

Blockchain technology is incorporated into Cloud Services [55] deployment as well. The blockchain allows to establish trust between different actors within the cloud infrastructure, thus, enabling trusted collaboration in multi-step computation tasks, *e. g. ,* ML pipelines execution training [57]. In addition, blockchain technology helps to bring transparency into the processes of cloud storage accounting and data assurance within a cloud infrastructure. In [77] authors present a blockchain-based decentralized marketplace where different actors can collaborate in the AI engineering process. The main asset that drives the AI applications development is the data on which the ML model is trained. With blockchain, authors make sure that data owners retain the ownership and privacy of the data while providing developers the means to access data for model training. The training algorithms are executed in the cloud infrastructure, where usage of permissioned blockchain allows to preserve the ownership and privacy of the data on the distributed computing resources. Authors of [78] present a model for a blockchain-based decentralized marketplace for online content. This model provides a tool-set to conduct data management and trading within the marketplace, without a need for a trusted intermediary. According to the authors, the novelty of the presented model is in a new content naming approach, which allows global indexing, thus, providing a mechanism for fast and transparent content search. In [79] authors present a decentralized blockchain-based marketplace for online content. The platform acts as an indexer, storing content listings and providing a mechanism to search and trade the content. In addition, the marketplace allows automatic payment to creators in case the content is bought. Blockchain technology allows performing the trading settlement without a trusted third party and enables transparent and fair content distribution within the marketplace. Authors of [80] present a blockchain-based marketplace model for Virtual Network Functions

(VNFs) [90] hosting. The owners of VNFs do not necessarily have sufficient resources to host their function. Thus, it creates a demand for the platform where these resources could be found. The proposed marketplace allows VNF owners to submit their order, indicating what are the requirements towards the resources that are needed to run VNF. Infrastructure owners in turn, compete to fulfill the order by placing their bids, indicating the cost of VNF hosting. According to the authors, blockchain technology allows making such a marketplace easy to audit and eliminates the need for a trusted third-party. Also, such a marketplace brings together VNF developers and infrastructure providers, thus promoting the development and usage of VNFs. In [81] authors propose a blockchain-based brokering mechanism that is used to allocate and manage network slicing in a 5G network. Authors introduce a *slice broker* as a new entity to help construct network slices from the resources supplied by different network providers. According to the authors, blockchain technology brings enhanced security and privacy features without a negative impact on the performance of the slice broker. Authors of [82] describe a Blockchain-based trUsted VNF packagE Repository (BUNKER). According to the authors, a blockchain-based BUNKER allows verifying VNF package integrity without the involvement of a trusted third party. Moreover, the rights and obligations of VNF package acquisition can be described in the Ethereum Smart Contract, which eliminates the need for a trusted third party and allows to automate the final settlement process. In addition, blockchain technology incorporation makes the VNF repository tamper-proof, and brings such benefits as transparency, data provenance, and accountability. In [83] authors describe a blockchain-based catalog *ProtectDDoS*, where vendors of distributed denial of service (DDoS) protection software can post and sell their products. Moreover, the users of ProtectDDoS can receive recommendations on the type of protection according to their requirements. Authors also use Ethereum smart contracts in order to maintain the integrity of the data about available DDoS protections. Finally, the authors implement their concept and demonstrate that confidentiality and integrity features are maintained for all parties collaborating within the ProtectDDoS system. In [84] authors propose a blockchain-based marketplace model for AI API access selling. The proposed marketplace, allows data owners to expose their cloud-hosted APIs in a secure way, allowing AI engineers to use ML models through the respective API. The novelty of this approach is that ML model exposure though API is distributed over several cloud infrastructures to secure the data from both cloud infrastructure providers and AI developers. Blockchain technology allows the distribution of API over multiple providers removing the need for a trusted centralized entity and making the system transparent and easy to audit. Authors of [85] present a model of a decentralized blockchain-based marketplace for Fog/Edge computing resources trading. Authors claim, that existing blockchain marketplaces while having decentralized components still partly rely on a number of centralized services. With their proposal, the authors show that the use solely of blockchain technology allows building a fully decentralized system while proving necessary functionality for marketplace operation.

Table 7.2: *Industrial proposals taxonomy on blockchain-based marketplaces*

Reference	**Application Area**	**Platform**	**Architecture**	**Description**
Ericsson [91]	Cloud Services	Hyperledger Fabric	Private Permissioned	A decentralized blockchain-based marketplace prototype and list of key requirements for decentralized digital marketplaces.
Wibson marketplace [92]	Cloud Services, Data Trading	Ethereum	Public Permissionless	A decentralized blockchain-based marketplace for fair, transparent and secure data trading.
Project XBR [93]	Cloud Services, Data Trading	Ethereum	Public Permissionless	An infrastructure for decentralized blockchain-based data marketplaces on-demand provisioning, where users can trade data assets.
IOTA marketplace [24]	IoT, Data Trading	Tangle	DAG-based	A decentralized blockchain-based marketplace based on a new type of distributed ledger based on DAG for IoT data trading.
Databroker DAO [94]	IoT, Data Trading	Ethereum	Public Permissionless	A decentralized blockchain-based marketplace designed to provide a transparent and secure way to buy and sell IoT sensors data.
Datum [95]	IoT, Data Trading	Ethereum	Public Permissionless	A blockchain-based decentralized network, that allows to store and trade data assets with enforcement of data usage rules.
Weeve [96]	IoT, Data Trading	IOTA, Ethereum, Hyperledger Fabric	Multiple Architectures	A blockchain-based decentralized platform to enable IoT data trading and establishment of Economy of Things.

In the context of cloud services, the application of blockchain technology has a number of common goals with the area of IoT such as data privacy and ownership preservation. However, within blockchain-based cloud services marketplaces, it is the infrastructure that is being the object of trade with the aim to host cloud-based services in it. Blockchain technology allows distributing the execution of the cloud-based service over several infrastructures, thus, protecting the data and services privacy and ownership. It can be also traced as a pattern, that the main goal of blockchain technology in all aforementioned application areas is the elimination of trusted third-party in the trading settlement process. In cloud services, blockchain smart contracts allow trading cloud infrastructure resources, as well as cloud-hosted data and services without an intermediary, allowing automation of settlement process and making it more transparent, time-efficient, and error-proof.

7.3.5.5 Industrial Proposals

There has been a number of industrial proposals and initiatives to apply blockchain technology to digital marketplaces. The majority of the proposals aim to implement a solution that will become a foundation for blockchain-based marketplaces mainly in areas of cloud services and IoT. In [91] author details the implementation of a prototype of decentralized marketplace using Hyperledger Fabric [29]. According to the author, smart contracts, while being the important technology for a marketplace implementation, are only a fraction of the functionality needed to build a functioning decentralized marketplace. The author claims that in private permissioned blockchain, in order for the participants to trust marketplace's operations each participant has to maintain at least one blockchain network node which hosts the ledger and smart contracts. Otherwise, there is no possibility to verify the validity of data operated by the smart contract, and trusted relationship on the blockchain can not be established. In addition, the author details requirements for decentralized marketplace implementation along with argumentation towards the design decisions made. Authors of [92] describe a decentralized marketplace called *Wibson*. The marketplace acts as a stage where the data is exchanged for tokens. The distinct feature of Wibson marketplace is that it uses an additional entity, called a notary, in the transaction settlement process. The notary is the data authenticity verification authority and acts as an intermediary in the transaction process. When the notary verifies the data, an encrypted copy of it is sent to the buyer. Further, after the funds are released to the seller, the data decryption key is sent back to the buyer. According to the authors, usage of blockchain technology and the introduction of the notary makes the data trading process fair, transparent, and secure. In [93] the author describes a decentralized marketplace infrastructure provider, called *XBR*. The main author's argument is that nowadays the data is collected and stored in a centralized manner, which limits the opportunities for potential buyers, *i. e.,* developers, to find and purchase the necessary data. Thus, the main aim of XRB is to develop the infrastructure which allows rapid, secure, and on-demand decentralized marketplace deployment. According to the author, deployed marketplaces are designed to perform trading operations with the help of blockchain smart contracts and provide a needed level of data privacy and security in a given data context. The author of [24] describes the main principles of new distributed ledger called *IOTA*. It uses a new data structure based on DAG called *Tangle*. In IOTA, transactions do not have fees, thus, eliminating the need for the mining process. The DAG itself is structured in a

manner different to the blockchain, thus, there no blocks or resulting blockchain. According to the author, IOTA solves such challenges of the public blockchain networks as scalability and privacy. Thus, as the main application area for the IOTA ledger is IoT, it enables high throughput of transactions as well as preserving data privacy within the IoT data marketplace. In [94] the authors introduce decentralized marketplace called *DataBroker DAO*. The main aim of the marketplace, according to the authors, is two-fold. Firstly, it aims to provide a transparent and secure way for IoT data owners to sell their data. Secondly, it aims to implement a decentralized marketplace where data consumers can easily find and buy required IoT data. According to the authors, such a marketplace enables new data usage scenarios and business opportunities such as smart city initiatives and governmental services enhancement. Authors of [95] describe a blockchain-based decentralized network called *Datum*. Datum allows secure storage of the data on the blockchain. For the purpose of data monetization, the DAT smart token is used as a currency in trading operations. The distinct feature of the Datum network is that it records data sharing rules established by the owner in a smart contract, and automatically enforces them during the trade process. Data sharing rules determine groups of entities, with whom data can be traded and access shared. In [96] authors present blockchain-based decentralized marketplace platform called *Weeve*. The main aim of the platform is to enable the deployment of transparent, secure, and scalable marketplaces for IoT data trading. The distinct feature of Weeve platform is that all IoT data is testified, *i. e.,* before being traded, data properties and validity are verified by the marketplace. According to the authors, with their platform they want to transform IoT data trading into Economy of Things, where data is traded transparently, fairly, and with reasonable pricing.

As can be seen from both academic and industrial proposals, there is a number of common goals that all aforementioned areas share in the context of blockchain-based marketplaces. However, each application area requires some specific service, *e. g.,* dynamic energy price regulation, or system characteristic, *e. g.,* GDPR compliance, which poses additional requirements to the blockchain system in terms of architecture, privacy-preserving capabilities, and trust.

7.3.6 Telecommunication Service Marketplaces

The applications and services in centralized AM and CSM typically do not span over several different governing entities. Thus, they do not introduce issues to build trust and automate business processes, since they are controlled in a centralized manner by one entity that acts as a trusted intermediary. However, in decentralized systems, where multiple non-trusting actors collaborate, a trust-enabling mechanism is needed for business settlement automation. The blockchain-based marketplace proposals discussed in Section 7.3.5 describe multiple application possibilities for blockchain technology, allowing us to reduce or completely eliminate the need for a trusted intermediary. In this work, we aim to extend the blockchain-based marketplace concept to *Telecommunication Services Marketplace (TSM)* that integrates multiple services offered by different CSPs. A structure of TSM is shown in Fig. 7.7. The idea behind the TSM is to make sure that there is a common set of processes that all actors who collaborate within the marketplace can rely on to establish their business relationship, without the need of a trusted third-party. In order to provide these processes, the following

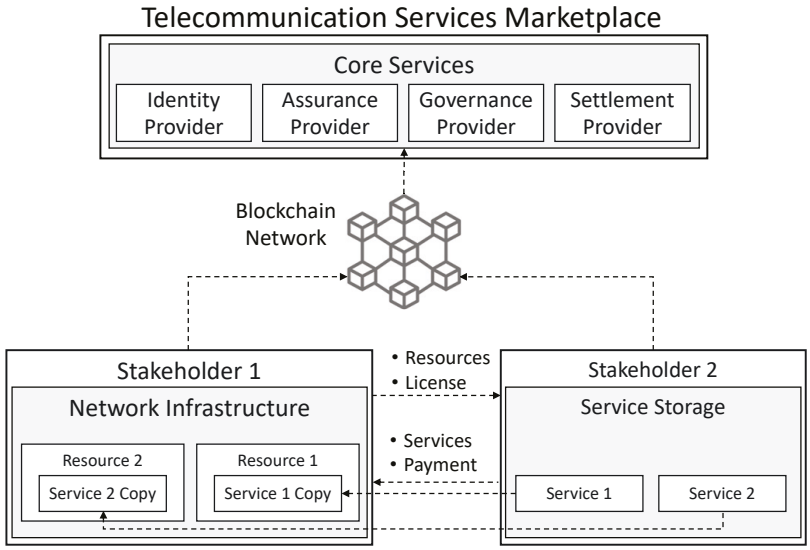


Figure 7.7: *Telecommunication Service Marketplace.*

core services of the TSM must be established: *Identity Management, Assurance, Governance* and *Business Settlement* [97]. The core services provide fundamental functionality for TSM operation and give users the ability to conduct business transactions in an automated manner. Due to telecommunication services being governed by different CSP it is essential to establish trust between CSPs within TSM to make automation of business transactions possible. In this case, automation and trust are achieved with the incorporation of the DLT into the core services of the TSM. With the distributed and immutable storage of DLT, every TSM participant can hold a synchronized copy of the ledger, and execute business transactions with blockchain smart contracts. With all core services established, a range of telecommunication services chains, *i.e., wholesale voice settlement, data on demand* or *mobile roaming*, can be decomposed into separate services, and provided within the TSM for CSPs. In turn, TSM is built to provide interoperability of core services with existing systems, *e. g. ,* blockchain-based self-sovereign identity management systems. In this way, we can maximize the inclusiveness of the TSM and conduct standardization activities to provide a unified way for companies to connect to TSM. For demonstration purposes we describe the next use-cases where such business settlement is involved.

7.3.6.1 SLA settlement use-case scenario

You are an innovative Virtual Reality (VR) [98] service developer. You have developed a new VR service that allows the users of your service to perform conference calls where they can see their conversation partners in full height and the figures of people can have premodeled face of a real person visible in VR [3, 99]. As a demonstration of the capabilities of your VR service, you have organized an event with 40 participants that takes place in two different physical locations. The participants will be brought together in a VR where they will have a possibility to break the barrier of distance by appearing near each other

in the digital world. The VR services, especially on such a large scale, transfer multiple gigabytes of data per second since different types of information are transmitted live: video, audio, location in a digital meeting, rendered face mimics, etc. For your event, you need an underlying network infrastructure that supports the communication requirements of the event. From a financial standpoint, you have no reason to buy all the network hardware, hence, you need to find an alternative. Such network infrastructure is owned by the local network provider, *i. e.*, CSP, which has the capacity to arrange the transmission of such high volumes of data per second, but none of the conventional billing plans covers the necessary network capacity. Ordinarily, you would need to go to the CSP and arrange the allocation of the network infrastructure beforehand. In such a case, the business settlement of the transaction would require the signing of an SLA involving a trusted third party, and possible additional costs and preparation time for the CSP if this is the first time that such a business transaction takes place. In contrast to manually executed, expensive and time-consuming [9] business settlement, *i. e.*, signing of SLA, described above, TSM allows you, as a VR developer, to find and rent the needed network infrastructure via the marketplace. The CSP is a supplier of infrastructure registered in the marketplace. The rental procedure of the required network infrastructure would be settled without involving a trusted third party, but via smart contract execution on the underlying blockchain P2P network. A ledger provides a distributed root of trust which is a key component of the automated business settlement. The governance of the system is distributed over the decentralized network of blockchain nodes who participate in the marketplace's maintenance. The assurance of QoS of the supplied services or the infrastructure is recorded in the smart contracts as an SLA and works as an obligation of the supplier towards the consumer [14].

7.3.6.2 Inter-CSP settlement use-case scenario

You are a small cell provider (SCP) that has entered a cellular market. As you just begin to establish yourself on the market, you have a limited network infrastructure as your customer base is still growing and revenue has not been generated yet. For your brand to grow, the coverage area and data plans have to be attractive for the customers to be competitive in the market. Since your investments are limited, the resources of your own network infrastructure may not be sufficient to meet the standards set by your competitors. Thus, you need to reach out to a large MNO, which has extended cellular network coverage, to rent additional network infrastructure on a long- or short-term basis. In this way, you can obtain the ability to grow as a cellular network brand by expanding over the areas of interest of your potential customers and generate new revenue by the increased customer base [15]. According to the current process of business settlement, you as an SCP need to contact every MNO available in your area and conduct negotiations on the terms and conditions of the agreement. The current process of such an inter-CSP transaction contains a portion of manually executed parts, which can be expensive and time-consuming. Also, the need for human intervention in the process may lead to manual errors and exposure to fraud. The TSM would enable you to review the network infrastructure options proposed by all the MNOs that are available in the area in a rich web and application interface, with the blockchain technology allowing you to automate the final settlement. The terms and conditions can be recorded in a smart contract providing legal context, with the blockchain bringing data assurance and trust into

the business settlement process without a need for third-party involvement.

The described use-cases are aimed to demonstrate the telecommunication industry blockchain applicability in situations where a trust-enabling technology is needed to enable process automation. With it, the creation of TSM would provide a platform for such processes to execute, in addition to the definition of the place where interested customers and CSPs can meet to enable new business opportunities.

7.3.7 Standardization Activities

Recently, a number of standardization activities have been conducted in the area of TSMs. These activities are aimed to present a set of well defined interfaces and processes which will help relevant parties in the industry to enable new business opportunities and collaboration models. Table 7.3 provides a summary of standardization activities included in this paper.

7.3.7.1 CBAN

Communication Business Automation Network (CBAN)[7] was launched by the ITW Global Leaders Forum ("GLF") and it is targeted to develop a platform which provides a set of core services to accelerate business settlement between different CSPs. The main premise to start this standardization activity is that nowadays business settlement process involves a mix of automated and manual activities. Manual activities are the results of involvement of a trusted third-party which acts as a trust anchor on behalf of participants of business settlement process. As far as manual activities involve human intervention, they are a subject of multiple issues: manual errors, long payment cycles, and exposure to fraud. The introduction of standardised framework for business settlement will make this process automated, real-time and trusted between two or more CSPs. The development of new telecommunication services will be also accelerated making them interoperable, while increasing integration ability of new services with legacy systems [9].

In order to achieve settlement automation, CBAN employs the DLT. By standardising DLT technologies which are used for core services CBAN makes the platform inclusive and interoperable. Also, due to DLT's distributed nature, it enables the possibility to avoid trusted third-party, thus, opening an opportunities to a full automation of business settlement process. Additionally, CBAN defines a TSM reference architecture [97], where minimum functionality and core services for the TSM are described.

CBAN initiative plays a number of roles in development of new unified approach to business settlement. First, CBAN governs the adoption of technological standards in order to guarantee interoperability between all participants of CBAN network. Second, CBAN governs the network of participants by maintaining participants registry. Lastly, CBAN coordinates all developments of new architectures and services within the CBAN network.

[7] https://www.cban.net/

Table 7.3: *Summary of standardization activities in the area of TSM*

Name	Area	Leader	Goals
Communication Business Automation Network [9]	Communication Service Provides	ITW Global Leaders Forum	An introduction of a standardized framework to make the process of business settlement automated, real-time, and trusted between two or more CSPs.
ETSI ISG PDL [100]	Permissioned Distributed Ledger	The European Telecommunications Standards Institute	A summary of standardization activities and research proposals on activities that have been done in the area of Permissioned Distributed Ledger.
Global System for Mobile Communications [45]	Mobile Operators	GSM Association	An analysis of the opportunities for the blockchain smart contracts to be used to record the agreements between operators, while using cryptocurrencies for business settlement.
TM Forum Catalyst [44]	Communication Service Provides	TeleManagement Forum	Construction of a federated CSPs marketplace where network infrastructure can be shared flexibly and securely in an automated way.

7.3.7.2 ETSI ISG PDL

The European Telecommunications Standards Institute (ETSI)[8] is an independent organization which performs standardization activities in the area of communications. In their recent document [100], exploring the global trend, ETSI have made a taxonomy of activities which have been done in the area of Permissioned Distributed Ledger (PDL). The document contains both standardization activities and research proposals. The main aim of the document is to identify applicable solutions, and provide enhancements and recommendations on the way forward.

7.3.7.3 GSMA

Global System for Mobile Communications (GSMA or GSM Association)[9] is an organization which represents mobile operators on a worldwide arena. In their report [45], they have analysed the opportunities which may be enabled by the blockchain technology in the area of business settlement for Mobile Operators. In this scenario, according to GSMA, smart contracts can be used to record the agreements between operators, while using cryptocurrencies for business settlement. The governance over a blockchain network is achieved by managing the network and smart contracts definition together, with the requirement that all parties agree on the contract revision. Also, since call data records (CDRs) are digital, they can be recorded on the ledger. Due to ledger's distributed nature, different operators connected to blockchain network can verify all CDRs in a trustworthy manner.

7.3.7.4 TM Forum

TeleManagement (TM) Forum is an association for CSPs in the telecommunication industry sector. With the recent initiative called *TM Forum Catalyst* their aim is to build a federated CSPs marketplace. The main premise of the initiative is that CSPs nowadays need a mechanism to share their network infrastructure flexibly and securely in an automated way

[8] https://www.etsi.org/
[9] https://www.gsma.com/

[44]. With the rise of 5G and an increasing number of IoT devices, such a mechanism can enable new ways for revenue generation and stimulate business growth. Thus, TM Forum employs DLT to define such a federated CSPs marketplace. DLT acts as a main trust-enabling technology, which enables transaction execution and value exchange between different actors within the marketplace. The business settlement agreement can be recorded in the smart contract and the settlement process itself can be automated and executed in real-time. In addition, DLT provides an audit infrastructure in the form of distributed immutable data storage providing a transparent way for all involved parties to verify any data operated by the smart contracts. Consequentially, the number of disputes can be reduced, due to the transparency and trustfulness of such a blockchain-based marketplace [101].

In [44], TM Forum specifies the high level architecture design as well as roles which are needed for minimum viable ecosystem establishment. Also, they define the value distribution mechanisms along with assurance and governance services description. Further, they describe APIs required for marketplace operation. Finally, they identify a number of challenges in federated CSPs marketplace implementation that TM Forum Catalyst initiative will explore in future.

7.4 Blockchain in Telecommunication Services Marketplaces

Having introduced the concept of the TSMs in Section 7.3.6, here we survey and elaborate the concepts and core services that comprise TSM. A basic set of necessary functionality for the TSM was outlined by CBAN [97]. It comprises the four core service *Identity Management, Assurance, Governance* and *Business Settlement*. We focus on this group of functions since it represents an agreed amount of functionality by the current telecommunication industry (operators and manufacturers). In addition, Table 7.4 provides a condensed view of the advantages of using TSM core services in the context of a decentralized blockchain-based marketplace as compared to their use in a centralized marketplace. The emphasis is on characteristics that lead to more democratic and robust services. The aim of the table is to provide an underlying basis such that the details of core services can be quickly understood. These core services are discussed in detail next.

7.4.1 Identity Management Service

The discussion on the identity management (IdM) service models is built in a way that shows the development and evolution of IdM systems derived from [26]. The evolution of IdM models in the history of computing systems and services development helps to realize the problems that each new IdM model solved or introduced. Also, it helps to avoid identified problems in newly designed IdM models. Finally, a number of scientific and industrial proposals on blockchain-based IdM models are discussed, *c.f.* Table 7.5.

IdM service combines all needed operations which are required to create and use a digital identity within a computing system. Digital identity is a requirement for any system where the target is to exchange data in a secure and accountable way. Traditionally, with the developments of network services architectures, the first IdM models were designed around the concept of centralized systems. In such a system, one centralized trust authority is set up to provide identity services for all users within a single service domain [102].

Table 7.4: *Advantages of decentralized blockchain-based marketplaces vs. Features of centralized marketplaces*

Core Service	**Features of Centralized Marketplaces**	**Advantages by Decentralized Blockchain-based Marketplace**
IAM	• A centralized authority manages a single identity database setup. • Centralized authority represents a single point of failure. • Restricts interoperability and reusability of digital identity.	• A decentralized network of nodes managing identity information, which protects from a single point of failure. • Enables reusability and interoperability of identity. • Enables users full control over the identity information.
Assurance	• Data assurance is provided by the centralized authority. • Inability to inspect the marketplace by external parties, *e. g.* , producers and consumers. • Possibility of violations by the marketplace, *e. g.* , unfair commission on consumer payments.	• Data assurance is provided by the immutability and transparency of the decentralized blockchain. • Blockchain's immutability ensures that the data the smart contract operates on is valid. • Transparency of operations for collaborating parties.
Governance	• Governance is performed solely by the marketplace operator. • All decisions are made within the centralized authority.	• Decentralization of governance within the system actors. • Increased automation, democratization, and time-efficiency of the governance activities.
Business Settlement	• Business settlement is executed and controlled by a central authority, which acts as a trusted third party. • May result in value distribution imbalance, as the marketplace may dictate billing rules and payments distribution.	• Smart contracts eliminate the need for a trusted third party. • Ability to verify the validity of the smart contract data. • Fair value distribution due to trusted, transparent, and automated business settlement.

Since there is only one entity that provides identity services, it is usually protected by perimeter-based defenses, *e. g.* , firewalls, IDS, and IPS. However, this implies that there is a central point of failure in the centralized IdM model with only one target to penetrate which frequently results in breaches and digital identity theft. In addition, the disadvantage of the centralized IdM model is that while identity can be used within a single domain, it cannot be reused in the context of another centralized IdM system. The further development of IdM to federated models enables the use of cross-domain identity. The IdMs which implement a federated model are called Single Sign-On (SSO) systems [103]. In the SSO model, the users are not bound to a single domain anymore, thus, the IdM system provides the ability to use the same identity to log-in to different services. The disadvantage of the federated model is that in a process of authentication, the user is still redirected to a home identity provider [26].

As an evolution of federated IdM, user-centric IdM systems allow users to use their identity across different domains without being bound to a home identity provider. In this case, a user takes control over personal identity data, and the home identity provider asks permission to release identity data to different IdM services for authentication. A Stork IdM initiative [104] applied this IdM model in the context of the European Union (EU) where users can authenticate in governmental services of different countries with the same identity. Although user-centric IdM is an advancement towards more flexible and usable identity management, it still has a number of disadvantages.

Table 7.5: *Proposals taxonomy on blockchain-based identity management services*

Reference	**Application Area**	**Platform**	**Architecture**	**Description**
T. Zhou et al. [105]	Self-Sovereign IdM	Ethereum	Public Permissionless	An IdM which uses smart contracts and is capable of merging different user identities under one unique identifier.
Y. Liu et al. [106]	Self-Sovereign IdM	Parity [107]	Public Permissioned	An IdM system architecture with guidelines for efficient usage of design patterns for data security and system scalability improvement.
H. Gulati et al. [108]	Self-Sovereign IdM	Not Applicable[1]	Not Applicable	An IdM scheme for a dynamic digital identity with the usage of blockchain technology.
Z. Cui et al. [109]	Self-Sovereign IdM	Hybrid blockchain model[2]	Multiple Architectures	A mutual authentication scheme for IoT nodes in Wireless Sensor Networks (WSN).
R. Soltani et al. [110]	Self-Sovereign IdM	Hyperledger Indy [111]	Private Permissioned	An IdM framework where authors aim to address the privacy requirements described in GDPR [86].
A. Othman et al. [112]	Self-Sovereign IdM	Not applicable[3]	Not Applicable	A novelty decentralized authentication method based on blockchain technology and SSI principles with the usage of DIDs.
R. Soltani et al. [113]	Self-Sovereign IdM	Hyperledger Indy	Private Permissioned	A key recovery solution that is based on SSI concepts and blockchain technology.
S. K. Gebresilassie et al. [114]	Self-Sovereign IdM	Tangle	DAG-based	A novel approach for IoT devices IdM based on DIDs, IOTA's Tangle DLT, and SSI principles.
M. P. Bhattacharya et al. [115]	Self-Sovereign IdM	Hyperledger Indy	Private Permissioned	An evaluation of blockchain-based IdM system's security against sensitive data leaks and man-in-the-middle attacks.
uPort [116]	Self-Sovereign IdM	Ethereum	Public Permissionless	An implementation of an IdM system that provides self-sovereign identity to users and organizations.
Sovrin [30]	Self-Sovereign IdM	Hyperledger Indy	Private Permissioned	An implementation of an IdM system which transitions the responsibility for identity information from traditional centralized identity to the identity holder.
Z. Zhao et al. [117]	User-centric IdM	Ethereum	Public Permissionless	A user-centric blockchain-based IdM model which allows users to control their identity information.

[1] No implementation available.
[2] The model assumes usage of hybrid public and private blockchains, but no implementation is available.
[3] No implementation available.

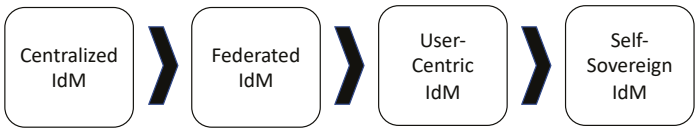


Figure 7.8: *Identity management models development.*

Despite user-centric identity being used across different domains, it is still stored on a server-side, *i. e.,* in centralized storage, and performs server-side authentication [26].

In contrast to previously described approaches, IdM systems based on a Self-Sovereign Identity (SSI) [118] are aimed to enable users' full control of their identity data. During online authentication, users can determine the amount of identity data released to authorizing party without a need for a centralized entity that stores identity data and which is placed in the middle between the user and the service. Users transitioning from the role of data subject to the data controller and manage their identity data directly determining ways in which data is being processed. The detailed path towards the Self-Sovereign Identity Management model was described by Christopher Allen in [119], where ten core principles of SSI are defined: Existence, Control, Access, Transparency, Persistence, Portability, Interoperability, Consent, Minimalization, and Protection. The progression of the IdM models from centralized to SSI is depicted in Fig. 7.8.

Nowadays, SSI is moving from a conceptual model to implementation in industrial solutions with the help of blockchain technology. In [120], the author describes the main advantages and disadvantages of blockchain incorporation in the implementation of the SSI. Additionally, the author provides a description of the SSI model where blockchain is used as a central technology. According to the author, the novelty of the paper is in bringing multiple opinions on the advantages and disadvantages of the SSI model from both academic and industrial representatives, where a road ahead is defined in the development of SSI. Authors of [105] propose an SSI framework based on the Ethereum blockchain, named EverSSDI. It uses smart contracts saved on the ledger, which merge different user identities under one unique identifier. With it, authors try to solve the identity information fragmentation problem by integrating into the proposed framework the Hierarchical Deterministic protocol [121]. This protocol allows the creation of cryptographical keys in a hierarchical structure, *i. e.,* to derive child keys from a parent key. According to the authors, the implementation solution allowed to demonstrate that the system allows the users to become single owners of their identity. In [106] authors identify and discuss critical components of the SSI system such as keys, identifiers, and credentials. Also, the authors provide an SSI-centred system view and guidelines for efficient usage of design patterns for data security and system scalability improvement. Finally, the authors present their platform's architecture and evaluate their proposal denoted as design pattern as a service (DPaaS). In [108] authors propose a scheme for a dynamic digital identity maintained with the help of blockchain technology. The proposed scheme makes use of biometric information in combination with other identity information that is being recorded on the blockchain with the consideration that it may evolve over time, *i. e.,* being dynamic. According to the authors, building a chain of dynamic identities recorded on the blockchain allows verifying previous identity operations up to the initial

identity, *i. e.,* the origin. In [109] authors propose a mutual authentication scheme for IoT nodes in Wireless Sensor Networks (WSN). Authors propose the usage of a hybrid blockchain network where cluster head nodes authenticate themselves on a global public blockchain and lower level IoT nodes are authenticated on a local private blockchain. According to the authors, the advantage of such a system is the control of local areas in the IoT network by enforcing private blockchain participation restrictions. In [110] authors employ a Hyperledger Indy [111] to their proposed identity management framework based on principles of SSI. With their proposal authors aim to address the privacy requirements described in the GDPR. In [112] authors employ the blockchain technology and SSI principles to propose a novel decentralized authentication method called *Horcrux*. According to the authors, the advantage of their protocol is in relying on the Decentralized identifiers (DIDs) [122, 123], which enable a decentralized identity implementation and remove the single point of compromise in the identity verification. In [113] authors provide a key recovery solution that is based on SSI concepts and blockchain technology. According to the authors, the solution can recover decentralized keys and uses Shamir's secret sharing scheme and Hyperledger Indy as a blockchain solution. In [114] authors employ DIDs, IOTA's Tangle DLT, and SSI principles to introduce a novel approach for IoT devices IdM. According to the authors, having analyzed the scalability and performance of blockchain-based DLTs, they came to the conclusion that it is not the best option to choose in the context of IoT IdM. With these considerations, authors employed IOTA's Tangle DLT which is DAG-based and fundamentally focuses on IoT, and, according to authors, shows better scalability and performance. In [115] authors discuss Hyperledger Indy based IdM system which employs SSI principles. In their study authors evaluate the IdM system's security against such security risks as sensitive data leaks and man-in-the-middle (MITM) [124] attack. Based on the evaluation, the authors propose to mitigate MITM attacks by encryption of DIDs before sending them to a verifier, to ensure the confidentiality of the data. Also, in terms of sensitive data leaks, the authors propose a sensitivity score model based on DID's attributes to evaluate the risks of sharing a particular portion of personal data.

From the industry side, a number of solutions have been presented in recent years. *uPort* [116] is an industrial open-source IdM system that targets both individual users and organizations to provide them with a blockchain-based self-sovereign identity. Identity itself is implemented as an Ethereum smart contract that serves as a digital identifier. An interesting characteristic of the uPort IdM systems is the ability to perform the identity operations both on- and off-chain. Authors of [125] provide an assessment of the advantages and limitations of uPort's efficiency and architecture. Also, the authors implement a Decentralized Application (DApp) [126, 127] based on the uPort IdM system to evaluate its operational capabilities, scalability, and efficiency. Lastly, the authors discuss the advantages and disadvantages of the uPort IdM based on the evaluation. uPort was compared to another emerging industrial IdM system called *Sovrin* [30]. The Sovrin was introduced by Sovrin Foundation as an IdM system that transitions the responsibility for identity information from traditional centralized identity to the identity holder and allows individuals to make decisions on how their personal data is processed and disclosed. According to the Sovrin Foundation, this transforms the interaction between identity holder and verifier, enhancing the user's control over the use of their personal information. For a more detailed read on blockchain-based IdM systems that

employ SSI principles, the reader is referred to [128].

Blockchain technology is not only applied to IdM systems that follow the SSI model. In [117] authors build a user-centric decentralized IdM which is based on smart contract technology. According to the authors, such IdM allows users to take full control over their identity and make decisions on which third-parties can obtain access to their identity information. In order to preserve identity anonymity authors support an attribute-based authentication scheme, additionally supporting an attribute reputation model, which, according to authors, preserves the user's identity trustworthiness in a decentralized system. For a more detailed read on blockchain-based identity systems, the reader is referred to [49].

7.4.1.1 Lessons learned

Blockchain technology opens opportunities for IdM systems enhancement. It allows the conceptual model of SSI to be implemented with help of DLT and blockchain smart contracts. Blockchain-based SSI systems enhance user sovereignty and allow making the identity management process decentralized and secure. Further, blockchain-based SSI allows combining different types of identities, *e. g. ,* biometrics, identity attributes, and DIDs, allowing users to not be bounded to a single authentication technique. Also, distributed ledger allows making the user identity dynamic, enabling it to evolve over time while still having access to older identities which preserves identity operations accountability. Moreover, these IdM systems allow to completely eliminate the use of centralized storages of identity information, thus, protecting from a single point of failure, breaches, and identity theft, associated with centralized approaches, and allowing user's full control over personal information.

In the context of TSMs, blockchain-based SSI enables the inclusiveness of the marketplace, giving the ability for willing CSPs to seamlessly connect to TSMs' infrastructure and conduct business settlement within it. Digital identity can be applicable over different blockchain networks which would allow seamless and fast integration of blockchain-based IdM systems of an interested CSP into a TSM. However, it is important that blockchain-based IdM systems, whether they follow SSI principles or not, are implemented according to a unified standard which would make IdM systems interoperable and easy to integrate.

7.4.2 Assurance Service

The *assurance service*, *i. e. ,* data assurance, in any computational system, especially where financial operations take place, is essential to establish a trusted relationship between multiple parties. Data provenance is one of the key factors in data assurance, which is defined as information about data's origin, modifications, and storage. In centralized systems, trusted storage of information is present, which acts as an assurance guarantee towards collaborating parties. All operations take place on a centralized server where the data is processed, verified, saved, and accounted for. Thus, the system has the ability to assure the data provenance, under the condition that collaborating parties trust a central authority. With the introduction of decentralized environments and cloud-native service development methodology [129], it has become possible to distribute the processing of the data over several microservices, each performing a certain type of data manipulation, *e. g. ,* encryption, anonymization, and storage. In the case of the entire system being managed by one operator, the assurance level still can

be maintained since one governing authority is used. However, when the data assurance has to be ensured among several initially non-trusted parties, a number of challenges emerge in ensuring the data provenance and validity in today's decentralized environments [130].

Blockchain technology with its distributed immutable storage may help to solve the issue of providing data assurance in decentralized systems. The immutability of the distributed ledger serves as an assurance that stored data has not been changed and has the original ancestry. The transaction validation and consensus protocols prevent anybody on the blockchain network from flooding the ledger with unverified information, which ensures data validity and provenance. Furthermore, on-premise data auditing is possible for all collaborating parties, due to the ledger's distributed nature, *i. e.* , each party holds a copy of the entire history of blockchain. With the introduction of private blockchain architectures, stakeholders can control who enters the blockchain network, thus, preventing outside adversaries from tampering with the blockchain. Finally, having data assurance established, opens new opportunities for digital assets management in a trusted decentralized blockchain-based environment [3].

Several research efforts have investigated the ability of blockchain technology to enhance data assurance in distributed systems. The question of data provenance becomes more and more important for customers, amplified by the use of supply chains that make it increasingly more challenging to know the exact source from where the product came from. In [131] authors present a blockchain-based framework that increases products' provenance knowledge. In their work, authors conclude that with the blockchain incorporation overall system data assurance is increased with a reduction of the risks of failing to obtain original products provenance. In [132] authors propose a framework that collects the information about the interaction of supply chain participants and shares it among them. According to the authors, their system helps to monitor the product provenance and bring transparency to supply chain participants and end customers.

The issue of data assurance is present as well in every field where any kind of personal data is flowing. In [133] authors discuss data assurance related to Big Data [134] and Healthcare [135]. Data assurance and security are discussed in general, and how the incorporation of blockchain technology can improve the system characteristics in both areas. In the discussion, the authors claim that Big Data users can benefit from the immutability of the blockchain as it puts the power to ensure data provenance into users' hands. Also, smart contracts can be the provenance guarantee of the data resources present on the blockchain. In Healthcare, the storage of the patients' medical data has to meet strict confidentiality requirements. In [133], authors discuss another proposal from [136], where storing of patients' data on the blockchain is described. According to the authors, this would benefit data assurance in comparison to the usage of cloud infrastructure for the same task. However, direct storage of personal data on the blockchain brings a number of challenges due to ledger immutability and GDPR restrictions. With these considerations, authors of [137] develop a framework for personal and confidential data sharing with respect to GDPR regulations. In the GDPR, the roles and responsibilities of the data controller and data processor are strictly defined and any violations may be subject to punishment. While blockchain immutability brings benefits to the data provenance, GDPR requires the system to give users the possibility to modify or remove personal data. This is why the authors propose to record the data access

actions of the data controller in the smart contracts on the blockchain, while the data itself is stored and processed off-chain. In this way, the data processing contracts are recorded on the blockchain along with the location of the data itself. This enables data access withdrawal from the controller in case of any violations and makes data removal possible, while the provenance of the contracts is guaranteed by the immutable ledger.

In the area of cloud infrastructures, data assurance is being enhanced with blockchain technology as well. In [138] authors propose a hybrid system whereby a cloud server along with the blockchain is used for the assurance of data provenance in drone communication. The direct communication channel is established between the drone and the cloud server, where for each record stored in the cloud a blockchain receipt is generated for communication data assurance and enhanced auditing. The authors of [139] propose to use a dedicated blockchain to store the provenance data of the objects stored in the cloud. The system called *ProvChain* collects and verifies the provenance data and securely stores it on the blockchain. According to the authors, usage of a dedicated blockchain makes the provenance data tamper-proof, reliable, and enhances data accountability. In [140] authors propose a similar system, where they employ cloud storage, InterPlanetary File System (IPFS) [141] and Ethereum blockchain. While the data is being saved in cloud storage, the data provenance assuring records are saved in the IPFS. While being saved in IPFS, a hash is being generated from every provenance record and saved on the Ethereum blockchain. With the implementation of the system, the authors also show the performance and scalability analysis results, along with the simulation of a provenance data modification attempt. The authors of [142] went a step further and proposed a system that automatically compensates the party whose rights were violated. In case of any SLA disagreement from the side of the cloud infrastructure provider, the Ethereum-based system automatically repays the agreed amount of compensation.

7.4.2.1 Lessons learned

Data assurance is of high importance in decentralized computing systems where non-trusting parties collaborate with each other. In particular, data provenance and validity are the main assurance characteristics that define the overall level of trust in the *assurance service* of the system. Current centralized systems provide data assurance as a central silo of trust, but still are vulnerable due to being a single point of failure. In decentralized systems, we can benefit from blockchain incorporation by exploiting its immutable storage as well as smart contracts programmability and flexibility. However, we need to take into consideration private data regulations if personal data is being processed in the system. For example, the immutability of the blockchain, while being beneficial for accounting, collides with GDPR's requirements for the user to have the right to remove or modify the personal data, stored in the system. As can be seen from the surveyed proposals, it is manageable with hybrid solutions, where blockchain stores only a hashed reference to real data.

From a TSM perspective, a blockchain-based assurance service allows formerly non-trusting parties to meet on the TSM's platform and conduct business settlement in a trusted manner. In this way, the data assurance, *i. e.*, provenance, and validity are ensured by the distributed immutable ledger. Moreover, new opportunities are opened for TSM participants in business operations accounting and computational assets management.

7.4.3 Governance Service

Without an established *governance service*, the system's stability and operational ability may be compromised. The governance service denotes an entity or a consortium, *i. e.*, number of entities, which performs an orchestration of the system, *i. e.*, makes decisions on how the system is operated and maintained. In addition, the governor of the system makes the decisions on system architectural and functional changes, performs standardization activities, and decides which existing industry standards should be adopted [9].

In centralized systems, governance is concentrated within one entity. All decisions are made within the central entity's boundaries, and any discussions with the system's users have a recommendatory character, as the final decision is up to central authority [143]. While the centralized governance allows the taking of fast decisions, as no multi-stakeholder discussions have to be held, it may be inefficient in the adoption of new technologies, be slow in the transformation of its internal systems to meet requirements of next-generation internet, and become a platform for the creation of a monopoly. With the shift towards decentralized systems, the governance becomes distributed over several entities with equal decision-making power. From a business perspective, every party presents its own interests which may collide with the interests of other governing bodies. Ultimately, it leads to the decision-making process becoming more time-consuming since all governing bodies have to come to a verbal and legal consensus, but contributes to the system's democratization and trust enhancement, and may be beneficial for the information technology industry's positive transformations. Blockchain technology incorporation has introduced new opportunities in decentralized systems governance development.

From a technical perspective, depending on the architecture of the blockchain, a pool of governing entities may be open or restricted. The Bitcoin cryptocurrency, which has public permissionless architecture, originally was designed to be an open system with unprecedented democracy where the blockchain is governed by all nodes participating in the transaction verification, *i. e.*, mining, process [7]. In practice, due to PoW consensus protocol, the governance of the system is performed by less than 10 nodes which maintain the majority of the hash rate produced in the network [144]. Considering such a consequence of PoW protocol operation, this started a number of initiatives that resulted in private blockchain architectures. The Hyperledger Fabric is designed as a permissioned system and enforces so-called private governance, which restricts the number of consensus participating nodes, thus, leaving the governance power within this group. The private blockchain governance is suitable for organizational deployment since it has better privacy-preserving characteristics and is better suited to address business needs [145].

From a legal perspective, the governance over blockchain-based decentralized systems depends on the architecture of the blockchain as well. In a permissioned blockchain, it is done by the consortium of organizations that operate and maintain the computational system, whereas in the permissionless blockchain it is all or a certain portion of the mining community. In [146] authors propose a theoretical framework to describe governance in the blockchain networks. The framework consists of six governance dimensions and three governance layers. They divide the whole governance process into the actions which happen on-chain and off-chain, thus, splitting technical and legal perspectives. The introduced governance dimensions are aimed to describe different aspects of blockchain operation and how they should be

dealt with on different governance layers. Authors of [147] examine the history of decisions made by governing entities in Bitcoin and Dash [148] cryptocurrencies from both legal and technical points of view. In Bitcoin, the implemented governance mechanism is called *Bitcoin Improvement Process (BIP)* [149]. It requires 95% of the nodes to agree on the change before it is applied. In order to address the rapidly growing Bitcoin network, the *Bitcoin Core* team proposed to increase the block size [150], to future-proof the cryptocurrency. This proposal resulted in a clash of interests between the miners and the Core team. Miners, being mostly interested in the amount of cryptocurrency they are generating, didn't want to reduce the amount of incentive they are receiving with increased block size. In conclusion, it took three years of negotiations to make the final decision and resolve pending issues. In contrast, Dash cryptocurrency uses *Decentralized Governance By Blockchain (DGBB)* [151] mechanism which defines a certain group of nodes called *Masternodes* which perform governance of the system. The proposal submitted by Evan Duffield to increase the Dash block size to 2MB [152] was accepted by the majority of the Masternodes in 24 hours. Authors of [153] propose a new governance model for the prevention of high-level governance issues. The model is based on a combination of PoW and PoS consensus protocols and targets to secure the governance of blockchain systems from the authoritative control and re-centralization of decision power. In addition, according to authors, their model enhances the environment of Decentralized Autonomous Organizations (DAO) [154] creation by transitioning part of governance power into the creators' hands. The DAO is the new type of organization, which has the governance rules specified on the blockchain network.

Having discussed the governance of the blockchain networks themselves, research proposals where blockchain technology is used to enhance decentralized systems governance are discussed next. Authors of [155] propose a blockchain-based technical solution that enables governance of decentralized micro-clouds. According to the authors, the main reasoning behind this proposal is the lack of the governance layer in the decentralized micro-cloud environments. With the incorporation of the blockchain system, authors were able to build a trusted and decentralized governance layer. In [156] authors propose a governance framework to enhance transparency and trust in software delivery where the software is developed by multiple distributed teams. The framework allows to control and enforce the Software Development Life Cycle (SDLC) [157] process compliance with decentralized governance of development steps in SDLC. Authors of [158] present a blockchain-based academic governance system with increased transparency and trust towards the process of verification of student records. According to the authors, for current academic systems, it takes up to 30 days to handle the verification of the student records, since a large portion of process execution is performed manually. Blockchain incorporation allowed automating some manually executed parts, resulting in less time needed for record verification.

In the area of Big Data, a number of proposals suggested the usage of blockchain to improve data governance in decentralized systems, thus, putting control in the hands of the data owners to decide how and by whom it is being processed. In the [159], authors, along with highlighting the limitations of centralized systems governance, discuss the benefits of blockchain technology usage to enhance the distribution of data governance in a decentralized system. According to authors, the blockchain technology enables the distribution of the governance towards the data owners and enables them to decide on the full life-cycle of

the data processing, starting from storage and ending with the precise entities who retrieve access to process it. For a further read on the blockchain application for data governance in decentralized systems, the reader is referred to [160].

In the area of Smart Cities governance, authors of [161] propose a new system called *blockchain-based employee assessment system (BEMPAS)*. The system is designed to address the issues of centralized governance such as lack of trust and accountability. Authors take the use-case of employee performance assessment and build a decentralized blockchain-based system which, according to authors, achieves transparency and trust of the governance between governmental workers in a Smart City context.

7.4.3.1 Lessons learned

Without an efficient and secure governance service, the blockchain network may be compromised and even go out of operation. Although the very idea of blockchain technology is to establish digital democracy and self-sovereignty, the example of the Bitcoin governance model showed that when too many unverified nodes are included in the decision-making process, it may become cumbersome and time-consuming. The idea of Masternodes introduction solves the issue of unverified nodes. However, it also introduces the possibility of authoritative control and re-centralization of decision power in case a certain amount of Masternodes become malicious. With the introduction of the permissioned blockchain architecture, private governance became possible leaving the decision-making power within a certain group of nodes. This type of governance is said to have good privacy-preserving characteristics and is better suited to address business needs.

Blockchain technology incorporation can enhance the governance service of decentralized systems. Blockchain technology allows the distribution of governance within the respective system and enables increased automation and time-efficiency of the processes. Also, in the case of multi-step processes such as SDLC, it allows distributing governance over to developing parties, thus, making the process transparent and trusted.

From the perspective of the TSM, blockchain-based governance service enables the distribution of decision-making power over a number of consortium nodes, as well as preserving the possibility for a close circle of trusted decision-makers with private blockchain architecture. However, according to CBAN [9], the consortium which governs the TSM should be inclusive but secure, providing TSM's participants with a sufficient level of assurance that the TSM is future-proof, robust, and trusted.

7.4.4 Business Settlement Service

The *business settlement* is a key service for the companies enabling their business opportunities. The settlement process comprises a number of steps which include the negotiation of the license, and legal license signing, *i. e.,* settlement. The process of business settlement is essential in revenue generation for the companies, and such characteristics as fault-tolerance, automation, and time consumption may be decisive in the implementation of the business opportunity. Nowadays, when systems are mostly being built with a centralized architecture, the business settlement does not pose any challenges as long as it is concealed within one system. However, at the moment when two or more non-trusted parties require

settlement execution, human intervention is required. The parts of the process which are manually executed may be subject to fraud and simple human errors. The outcome of the manually executed part is the assembling of a contract, *e. g.*, license, which reflects the right and obligations of involved parties. Another major part of the settlement process is the establishment of trust between initially non-trusted parties. Nowadays, in the context of two or more centralized systems, a third-party is often needed to be involved in the settlement process with two aims: 1) to act as a trust-enabling entity between non-trusted parties, and 2) to sign an assembled contract. The involvement of a third-party, while allowing to establish trust, makes the settlement process time-consuming and expensive, since reaching a third-party manually takes time and its services have to be paid.

Naturally, the idea of third-party elimination from the process sets the prospect towards enhanced business settlement and enables positive developments, such as automation and cost reduction. Therefore, the main obstacle is the trust enabling technology that substitutes third-party, can be automated, and does not require or reduces the incentive for the job done. The DLT technology appears to have a set of characteristics that contribute towards the solution of the problems present in a settlement process. The ledger helps to establish a root of trust by providing immutable storage of information in the form of a blockchain. When collaborating parties are a part of a blockchain network, they write to the ledger only the information which was verified by the consensus protocol, thus agreeing on the information's correctness, *i. e.*, provenance, and validity. Further, the settlement agreement can be recorded in executable smart contracts where participants' rights and obligations are specified. Finally, when all parts of the system in place, the entire business settlement execution process can be automated by orchestration technology.

In recent years, a number of research articles have been published in the area of blockchain-based business settlement. With blockchain, researchers aim to make the on-chain settlement process trusted and efficient, and automated. In [162] the author explores the idea of blockchain technology application in the energy market, *i. e.*, Smart Energy Grid. The author claims that blockchain technology can help with the accounting of metered energy flows. The author claims that the smart contract technology applied to the settlement process makes it more efficient, thus allowing to remove a trusted third-party. In addition, blockchain technology allows the ability to fully automate the business settlement process. Authors of [163] present a blockchain-based decentralized market solution for P2P energy trading. The main argument of the paper is that current centralized market solutions cannot provide enough scalability and flexibility for a rapidly growing number of distributed prosumers of electrical energy. Thus, the energy market has to be decentralized, to be able to handle consumer and prosumer needs. The authors base the market's energy trading settlement process on blockchain smart contracts, where transaction details are recorded. The blockchain also enhances the market's ability to analyze electricity consumption and production rates through the immutable ledger. The trading data analysis helps to regulate electricity trading prices, thus increasing the efficiency of prosumer-generated electricity. In [164] authors provide a detailed analysis of blockchain-based settlement mechanisms called Global Balancing Settlement (GBS) and Splitting Settlement (SS). These settlement mechanisms are implemented as smart contracts stored on the blockchain. The authors provide a reference to current centralized settlement efficiency and how blockchain-based

mechanisms perform in comparison to it. The distinct feature of this study is that the implemented system was applied in real-life conditions in a small residential community. The results are provided in a form of an electricity price increase for the seller, and an electricity price decrease for a buyer throughout the day. The results have shown, that although GBS gives a larger price efficiency boost for both sellers and buyers compared to SS, both blockchain-based settlement methods outperform traditional centralized settlement.

In the area of e-commerce, the authors of [165] propose a blockchain-based system with autonomous transaction settlement called NormaChain. Since all transaction settlement is handled via cryptocurrency transition from one user wallet to another, authors derive their own coin called NorMaCoin (NMC) to decrease fiat currency inclusion in a system, thus increasing scalability and efficiency with native blockchain currency. The settlement terms are recorded in a blockchain smart contract where all information on buyer, seller, product, and amounts of currency is available. According to the authors, they are able to demonstrate that the settlement approach can be fully automated, and at the same time it can be executed in a trusted, transparent and efficient manner.

There has also been a number of industry initiatives to enhance and standardize the business settlement process in TSM. While academic proposals mostly discuss new ways of settlement execution on a newly designed blockchain-based marketplace, industrial proposals aim to help current CSPs to transform their operational frameworks in a way they are interoperable with TSM's core services. Since TSM is by design a decentralized marketplace, the participants are becoming a part of the distributed infrastructure, which exposes their telecommunication services and infrastructure to be rented within a marketplace. The CBAN [9] is an initiative that aims to define a *tool-set* for the TSM by standardizing the technologies and development practices with which the core services of the marketplace are implemented. With this, according to CBAN, it will become possible to fully automate business settlement process routines that are executed in a trusted and transparent manner. Authors of [44] describe the new initiative called *TM Forum Catalyst*, which aims to create a federated CSPs Marketplace. The authors emphasize the obsolescence of current approaches towards CSPs' network infrastructure rental settlement process. The settlement process has to be transformed in a way that allows CSPs to implement a more agile and on-demand rental which enables new business opportunities for telecommunication market participants.

7.4.4.1 Lessons learned

The business settlement process is of high importance in the context of digital marketplaces. It is the main service-enabler of new business opportunities for companies. In general, business settlement starts with the assembling of a contract. The next step, signing of the contract, in today's business conditions often requires the presence of a trusted third-party, *i. e.* , intermediary, which acts as a trust enabling entity and concludes business settlement. Such settlement process flow involves a number of manual steps, which require human intervention and represent the bottleneck in an execution. The aim of eliminating the trusted third-party requirement is pursued by both academia and industry and appears to be solved with the employment of DLT as the main trust-enabling technology. DLT and the blockchain as an implementation of it appear to be able to enable trusted and transparent business settlement while providing acceptable data assurance and security.

In the context of TSMs, the settlement process that is used nowadays complicates the implementation of new business opportunities. Blockchain technology enables the process of telecommunication services and network infrastructure renting to be executed without the involvement of a trusted third-party, thus allowing full automation. In addition, full automation allows making the rental settlement process more agile and on-demand, opening a new set of opportunities towards the implementation of new business scenarios.

7.5 Future Research Directions

Today's blockchain-enabled services still face a number of challenges that need to be solved in order for blockchain to be applied in the context of digital marketplaces at large and in TSMs in particular. Although in this paper we provide an overview of the TSM's architecture and core services that constitute the foundation of such a marketplace, next, we describe the challenges that are out of the scope of this survey and should be addressed in the future. Future challenges are chosen with the consideration of TSM's core services and the possibility to make them more *democratic* while *robust*.

First, the assurance and governance services are enabled by the inherent characteristics of blockchain technology. Thus, the improvements and new developments in this technology itself may trigger improvements in these core services. The openness of blockchain technology makes the overall system more democratic while its distributed nature makes it more robust.

Further, the Blockchain-based IdM service may benefit from incorporating physical identity, making it more *robust* in terms of available options to authenticate oneself in a system. The business settlement service may benefit from an established way to transition financial assets to a blockchain and use them as a cryptocurrency, making them more *democratic*. Finally, the interoperability of blockchain-based services is the main aim of the majority of standardization activities. Interoperable blockchain-based services would maximize the inclusiveness of digital marketplaces, and open a new set of business opportunities. These future research directions are discussed next in detail.

7.5.1 Physical Identity Management on a Blockchain

As discussed in Section 7.4.1, blockchain-based IdM systems are mostly based on the SSI principles and are well described by both academia and industry. Blockchain and SSI principles make the IdM process more democratic and decentralized, thus, eliminating the need for a centralized identity server. Research efforts and interest in blockchain-based SSI model resulted in such industrial implementations as uPort and Sovrin. Both implementations show that SSI is possible to apply in digital systems and it helps to solve such issues of previous IdM models as a central point of failure and data breaches. However, these systems do not consider the connection between the physical and digital identity within the IdM system.

Physical identity is represented most of the time by a card that is issued to the identity holder and provided to an identification entity as identity proof [166]. On a national level, it is for example the passport of a person, which is used for identification in governmental and private institutions within a particular country. However, there should be a mechanism

of physical identity usage in the context of a digital system. At this point, to the best of our knowledge, the systems which enable the digital representation of physical identity are mainly designed as SSO [167], and they are built on a user-centric IdM model which uses centralized silos of identity information. We think that it is worth exploring the possibility of physical identity representation in the blockchain-based SSI model. In this case, the digitized physical identity is stored on the user premise which enables full control over it and makes the overall IdM system more democratic and robust, in contrast to SSO.

7.5.2 The Transition of Financial Assets to a Blockchain

In the physical world, it is the fiat currencies, *e. g.*, a national currency or precious metals, that allow us to pay for items and services that we want to acquire. These currencies can be digitized in a form of banking accounts, where one's assets are accessible through the respective card number and password, as well as some ownership verification mechanisms, *e. g.*, text message or SSO. While a banking system is convenient to use within a particular bank, the transition of assets between different banks may require paying a high fee and pose possible complications on the transaction process, *e. g.*, additional reference number of a receiver bank. Thus, the employment of blockchain technology in financial systems opens an opportunity to implement trustworthy and agile money transferring systems in an inter-bank and cross-country context [168, 169]. In addition, on-chain currencies can be used in the digital marketplace context to make payments during transactions as was described in Section 7.4.4. However, we face a number of challenges in the case of fiat currencies transitioning to blockchain-enabled systems. First, there should be an entity that regulates an exchange ratio of fiat currency into an on-chain token. The regulatory body should be present to make the price of on-chain tokens stable and protected from illegal financial operations. Second, all transactions have to meet legal requirements for such financial transitions to take place. The governments of countries tend to regulate all financial transactions as it is a vital part of a country's economy [170].

Nowadays the cryptocurrencies such as Bitcoin and Ethereum have shown that financial assets in a form of fiat currencies can be transitioned to blockchain in a form of on-chain tokens and used as a new form of digital currency. While this makes the transactions more democratic, in certain countries such on-chain financial operations are banned due to the government's inability to track transaction participants and control the overall payment process [170]. Thus, we think that it is worth exploring the mechanisms to enable legal and secure fiat currencies transitioning into on-chain tokens.

7.5.3 Interoperability of Blockchain-enabled Services

Blockchain technology brings a number of advantages to the services that employ it as was shown in Section 7.4. DLT acts as a trust enabling entity and provides an immutable distributed storage that prevents data loss and makes it highly challenging to alter transactions embedded into it. However, while there are a number of initiatives to build blockchain-based services and systems, the majority of times initial system design is not based on a common standard. While this is not an issue for a sole system's operation, it limits its interoperability and restricts the ability to interconnect different blockchain-based systems [171].

There is a number of standardization activities conducted on blockchain technology application in an area of TSM as described in Section 7.3.7. However, the core services which potentially can be used within TSM, even at this point are being developed with different blockchain architectures, *e. g. ,* uPort and Sovrin. Thus, we think that it is worth investigating the interoperability of blockchain-based services overall. In addition, we would like to explore the possibility to propose unified guidelines to build interoperable blockchain-based services.

7.6 Summary and Outlook

This work provides a survey on the academic and industrial proposals in the area of digital marketplaces at large and TSMs in particular. We discuss the current state of the digital marketplaces and the advantages and disadvantages of the centralized marketplace architecture. As our discussion showed, a centralized architecture poses a number of challenges in terms of transparency and democracy for the entities operating within the marketplace. In addition, it is necessary for the trusted intermediary to be present in a centralized system's business settlement process. The transition towards a *decentralized marketplace architecture allows addressing these challenges.* Proposals of such marketplaces and adoption of blockchain technology resulted in a number of scientific and industrial solutions for architectural components needed by blockchain-enabled digital marketplaces. These solutions demonstrated that blockchain technology reduces or eliminates the need for a trusted intermediary in business settlement process while providing data assurance, transparency, and making the overall marketplace system more democratic. In addition, blockchain technology provides a foundation for identity management, governance, and business settlement services implementation.

We describe the challenges that current CSP face while conducting the business settlement and provide the use-cases for the TSM where the business settlement can be automated in a decentralized marketplace while giving service developers and CSPs a "central" place, *e. g. ,* marketplace's user interface, to look for telecommunication services or network infrastructure renting. Also, we discuss the main standardization activities on the TSMs and provide the discussion on the set of core services that establish TSM's platform and give marketplace's collaborators the common set of processes that they can rely on to establish their business relationships and enable new business opportunities.

The surveyed literature shows evidence that blockchain can provide advantages in the four core services in TSMs: *identity management, assurance, governance,* and *business settlement.* As shown in Section 7.4, there are multiple proposals on how to implement these four core services. The common denominator for these proposals is that they are all based on blockchain technologies, but otherwise they each emphasize various aspects and characteristics for the service they define. Whereas each proposal has its own benefits and drawbacks, it is not yet clear what combination is the optimal one when architecting a TSM. More work is required in this direction, which also explains the existence of several ongoing standardization activities, described in Section 7.3.7. Similarly, there are multiple competing blockchain technologies that can support the implementation of these proposals. It is important to understand the tradeoffs between these technologies in terms of performance (e.g., transactions per time unit, energy consumption, consensus algorithm) as well as security,

privacy, and last but not least compliance with existing (national and international) laws and regulations. These are fundamental research issues that require additional studies. We also see a need for a more comprehensive quantitative evaluation of existing solutions in realistic environments.

Acknowledgment

The authors would like to acknowledge the funding project supporting the work presented in this paper: The work was partly sponsored by the Swedish Knowledge Foundation through the project *Symphony - Supply-and-Demand-based Service Exposure using Robust Distributed Concepts*. The project partners in Symphony are Ericsson AB (Stockholm, Sweden) and Affärsverken Energi AB (Karlskrona, Sweden).

References

[1] F. Fitzek, F. Granelli, and S. Patrick, eds. *Computing in Communication Networks*. Academic Press, 2020. ISBN: 9780128204887.

[2] A. Sefidcon, W. John, M. Opsenica, and B. Skubic. "The Network Compute Fabric". In: *Ericsson Technology Review* (2021).

[3] S. Yrjola, P. Ahokangas, M. Matinmikko-Blue, R. Jurva, V. Kant, P. Karppinen, M. Kinnula, H. Koumaras, M. Rantakokko, V. Ziegler, A. Thakur, and H.-J. Zepernick. *White Paper on Business of 6G*. 2020. arXiv: 2005.06400.

[4] A. S. Prasad, M. Arumaithurai, D. Koll, and X. Fu. "DMC: A Differential Marketplace for Cloud Resources". In: *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. May 2019, pp. 198–209.

[5] J. Friman, M. Ek, P. Chen, J. Manocha, and J. Soares. "Service Exposure - A Critical Capability in a 5G World". In: *Ericsson Technology Review* (2019).

[6] T. Kollmann, S. Hensellek, K. de Cruppe, and A. Sirges. "Toward a renaissance of cooperatives fostered by Blockchain on electronic marketplaces: a theory-driven case study approach". In: *Electronic Markets* 30.2 (2020), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

[7] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. URL: https://bitcoin.org/bitcoin.pdf (visited on 06/18/2023).

[8] J. Singh and J. D. Michels. "Blockchain as a Service (BaaS): Providers and Trust". In: *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 2018, pp. 67–74. DOI: 10.1109/EuroSPW.2018.00015.

[9] CBAN. *Communications Business Automation Network Whitepaper Version 1.0*. URL: https://www.cban.net/resources (visited on 06/18/2023).

[10] Y. Diao, L. Lam, L. Shwartz, and D. Northcutt. "Modeling the Impact of Service Level Agreements During Service Engagement". In: *IEEE Transactions on Network and Service Management* (Dec. 2014), pp. 431–440.

[11] W. Chen and I. Paik. “Toward Better Quality of Service Composition Based on a Global Social Service Network”. In: *IEEE Transactions on Parallel and Distributed Systems* 26.5 (2015), pp. 1466–1476. DOI: 10.1109/TPDS.2014.2320748.

[12] R. B. Uriarte, R. de Nicola, and K. Kritikos. “Towards Distributed SLA Management with Smart Contracts and Blockchain”. In: *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. Dec. 2018, pp. 266–271.

[13] S. Jalali and C. Wohlin. “Systematic literature studies: Database searches vs. backward snowballing”. In: *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. 2012, pp. 29–38. DOI: 10.1145/2372251.2372257.

[14] N. Afraz and M. Ruffini. “A distributed bilateral resource market mechanism for future telecommunications networks”. In: *2019 IEEE Globecom Workshops* (2019).

[15] E. D. Pascale, H. Ahmadi, L. Doyle, and I. Macaluso. “Toward Scalable User-Deployed Ultra-Dense Networks: Blockchain-Enabled Small Cells as a Service”. In: *IEEE Communications Magazine* (Aug. 2020), pp. 82–88.

[16] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and Y. Liu. “A Survey of Blockchain Technology Applied to Smart Cities: Research Issues and Challenges”. In: *IEEE Communications Surveys and Tutorials* 21.3 (2019), pp. 2794–2830. DOI: 10.1109/COMST.2019.2899617.

[17] V. Fernandez-Anez. “Stakeholders Approach to Smart Cities: A Survey on Smart City Definitions”. In: *Lecture Notes in Computer Science*. Springer Verlag, 2016, pp. 157–167.

[18] A. Panarello, N. Tapas, G. Merlino, F. Longo, and A. Puliafito. “Blockchain and IoT Integration: A Systematic Survey”. In: *Sensors* (Aug. 2018).

[19] S. Singh and N. Singh. “Internet of Things (IoT): Security challenges, business opportunities & reference architecture for E-commerce”. In: *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*. Oct. 2015, pp. 1577–1581.

[20] G. Ishmaev. “The Ethical Limits of Blockchain-Enabled Markets for Private IoT Data”. In: *Philosophy & Technology* (Sept. 2020), pp. 411–432.

[21] N. Slamnik-Krijestorac, H. Kremo, M. Ruffini, and J. M. Marquez-Barja. “Sharing Distributed and Heterogeneous Resources toward End-to-End 5G Networks: A Comprehensive Survey and a Taxonomy”. In: *IEEE Communications Surveys & Tutorials* (2020), pp. 1592–1628.

[22] T. Gabriel, A. Cornel-Cristian, M. Arhip-Calin, and A. Zamfirescu. “Cloud Storage. A comparison between centralized solutions versus decentralized cloud storage solutions using Blockchain technology”. In: *2019 54th International Universities Power Engineering Conference (UPEC)*. Sept. 2019.

[23] Y. Lewenberg, Y. Sompolinsky, and A. Zohar. “Inclusive Block Chain Protocols”. In: *Lecture Notes in Computer Science*. 2015, pp. 528–547.

[24] S. Popov. *The Tangle*. 2018. URL: https://assets.ctfassets.net/r1dr6vzfxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1_4_3.pdf (visited on 06/18/2023).

[25] T. Ali Syed, A. Alzahrani, S. Jan, M. S. Siddiqui, A. Nadeem, and T. Alghamdi. "A Comparative Analysis of Blockchain Architecture and its Applications: Problems and Recommendations". In: *IEEE Access* (2019), pp. 176838–176869.

[26] J. Bernal Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. Torres Moreno, and A. Skarmeta. "Privacy-Preserving Solutions for Blockchain: Review and Challenges". In: *IEEE Access* 7 (2019), pp. 164908–164940. DOI: 10.1109/ACCESS.2019.2950872.

[27] G. Wood. *Ethereum: a secure decentralised generalised transaction ledger*. Ethereum Project Yellow Paper, 2014, pp. 1–32. URL: https://gavwood.com/paper.pdf (visited on 06/18/2023).

[28] R. Yang, R. Wakefield, S. Lyu, S. Jayasuriya, F. Han, X. Yi, X. Yang, G. Amarasinghe, and S. Chen. "Public and private blockchain in construction business process and information integration". In: *Automation in Construction* (Oct. 2020).

[29] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick. "Hyperledger Fabric". In: *Proceedings of the Thirteenth EuroSys Conference*. 2018, pp. 1–15. DOI: 10.1145/3190508.3190538.

[30] A. Tobin and D. Reed. *The Inevitable Rise of Self-Sovereign Identity*. Sept. 2017. URL: https://sovrin.org/wp-content/uploads/2018/03/The-Inevitable-Rise-of-Self-Sovereign-Identity.pdf.

[31] F. Tschorsch and B. Scheuermann. "Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies". In: *IEEE Communications Surveys & Tutorials* (2016), pp. 2084–2123.

[32] R. C. Merkle. "Protocols for public key cryptosystems". In: *IEEE Symposium on Security and Privacy* (1980), pp. 122–134.

[33] T. Neudecker and H. Hartenstein. "Network Layer Aspects of Permissionless Blockchains". In: *IEEE Communications Surveys & Tutorials* (2019), pp. 838–857.

[34] O. Pal, B. Alam, V. Thakur, and S. Singh. "Key management for blockchain technology". In: *ICT Express* (Aug. 2019).

[35] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, and E. Dutkiewicz. "Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities". In: *IEEE Access* (2019), pp. 85727–85745.

[36] F. Yang, W. Zhou, Q. Wu, R. Long, N. N. Xiong, and M. Zhou. "Delegated Proof of Stake With Downgrade: A Secure and Efficient Blockchain Consensus Algorithm With Downgrade Mechanism". In: *IEEE Access* (2019), pp. 118541–118555.

[37] M. Castro and B. Liskov. "Practical Byzantine Fault Tolerance". In: *Proceedings of the Third Symposium on Operating Systems Design and Implementation*. New Orleans, Louisiana, USA: USENIX Association, 1999, pp. 173–186.

[38] M. Castro and B. Liskov. “Practical byzantine fault tolerance and proactive recovery”. In: *ACM Transactions on Computer Systems* (Nov. 2002), pp. 398–461.

[39] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou. “A Survey of Distributed Consensus Protocols for Blockchain Networks”. In: *IEEE Communications Surveys & Tutorials* 22.2 (2020), pp. 1432–1465. DOI: 10.1109/COMST.2020.2969706.

[40] C. Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda. “Analysis of Security in Blockchain: Case Study in 51%-Attack Detecting”. In: *2018 5th International Conference on Dependable Systems and Their Applications (DSA)*. Sept. 2018, pp. 15–24.

[41] M. Bartoletti and L. Pompianu. “An Empirical Analysis of Smart Contracts: Platforms, Applications, and Design Patterns”. In: *International Conference on Financial Cryptography and Data Security*. 2017, pp. 494–509.

[42] R. M. Parizi, Amritraj, and A. Dehghantanha. “Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability and Security”. In: *Lecture Notes in Computer Science*. Vol. 10974 LNCS. 2018, pp. 75–91. DOI: 10.1007/978-3-319-94478-4_6.

[43] M. Pournader, Y. Shi, S. Seuring, and S. L. Koh. “Blockchain applications in supply chains, transport and logistics: a systematic review of the literature”. In: *International Journal of Production Research* (Apr. 2020), pp. 2063–2081.

[44] M. Nati et. al. *Federated CSPs Marketplace*. 2020. URL: https://www.tmforum.org/wp-content/uploads/2020/11/Federated_CSPs_Marketplace_Whitepaper_C20.0.34.pdf (visited on 06/18/2023).

[45] GSMA. *Blockchain – Operator Opportunities Version 1.0*. URL: https://www.gsma.com/newsroom/resources/ig-03-blockchain-operator-opportunities-v1-0/ (visited on 06/18/2023).

[46] L. Bondan, M. F. Franco, L. Marcuzzo, G. Venancio, R. L. Santos, R. J. Pfitscher, E. J. Scheid, B. Stiller, F. De Turck, E. P. Duarte, A. E. Schaeffer-Filho, C. R. P. dos Santos, and L. Z. Granville. “FENDE: Marketplace-Based Distribution, Execution, and Life Cycle Management of VNFs”. In: *IEEE Communications Magazine* (Jan. 2019), pp. 13–19.

[47] Y. V. Maksimov, S. A. Fricker, and K. Tutschku. “Artifact Compatibility for Enabling Collaboration in the Artificial Intelligence Ecosystem”. In: *Lecture Notes in Business Information Processing*. June 2018, pp. 56–71.

[48] J. D. Harris and B. Waggoner. “Decentralized and Collaborative AI on Blockchain”. In: *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*. July 2019, pp. 368–375.

[49] Y. Liu, D. He, M. S. Obaidat, N. Kumar, M. K. Khan, and K. K. Raymond Choo. “Blockchain-based identity management systems: A review”. In: *Journal of Network and Computer Applications* (Apr. 2020).

[50] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka. “Security Services Using Blockchains: A State of the Art Survey”. In: *IEEE Communications Surveys & Tutorials* (2019), pp. 858–880.

[51] B. Shala, U. Trick, A. Lehmann, B. Ghita, and S. Shiaeles. “Blockchain and Trust for Secure, End-User-Based and Decentralized IoT Service Provision”. In: *IEEE Access* (2020).

[52] S.-F. Chang. “Application Marketplace as a Service - A Reference Architecture for Application Marketplace Service”. In: *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. 2010, pp. 186–192. DOI: 10.1109/3PGCIC.2010.32.

[53] D. Tilson, C. Sorensen, and K. Lyytinen. “Change and Control Paradoxes in Mobile Infrastructure Innovation: The Android and iOS Mobile Operating Systems Cases”. In: *2012 45th Hawaii International Conference on System Sciences*. Jan. 2012, pp. 1324–1333.

[54] A. OI, M. Nakajima, Y. Soejima, and M. Tahara. “Reliable Design Method for Service Function Chaining”. In: *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. 2019. DOI: 10.23919/APNOMS.2019.8892959.

[55] Y. Jiang, C.-S. Perng, A. Sailer, I. Silva-Lepe, Y. Zhou, and T. Li. “CSM: A Cloud Service Marketplace for Complex Service Acquisition”. In: *ACM Transactions on Intelligent Systems and Technology* 8.1 (2016), pp. 1–25. DOI: 10.1145/2894759.

[56] D. Pudasaini and C. Ding. “Service Selection in a Cloud Marketplace: A Multi-Perspective Solution”. In: *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*. 2017, pp. 576–583. DOI: 10.1109/CLOUD.2017.79.

[57] R.-V. Tkachuk, D. Ilie, and K. Tutschku. “Towards a Secure Proxy-based Architecture for Collaborative AI Engineering”. In: *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*. Naha, Japan: IEEE, 2020, pp. 373–379. DOI: 10.1109/CANDARW51189.2020.00077.

[58] B. Karim, Qing Tan, J. R. Villar, and E. de la Cal. “Resource brokerage ontology for vendor-independent Cloud Service management”. In: *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*. Apr. 2017, pp. 466–472.

[59] R. Bayindir, E. Hossain, and S. Vadi. “The path of the smart grid -the new and improved power grid”. In: *2016 International Smart Grid Workshop and Certificate Program (ISGWCP)*. May 2016, pp. 1–8.

[60] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. “Blockchain Based Decentralized Management of Demand Response Programs in Smart Energy Grids”. In: *Sensors* 18.2 (2018), pp. 1–21. DOI: 10.3390/s18010162.

[61] S. Saxena, H. Farag, A. Brookson, H. Turesson, and H. Kim. “Design and Field Implementation of Blockchain Based Renewable Energy Trading in Residential Communities”. In: *2019 2nd International Conference on Smart Grid and Renewable Energy (SGRE)*. IEEE, Nov. 2019, pp. 1–6. DOI: 10.1109/SGRE46976.2019.9020672.

[62] P. D. Suzzoni. “Are regulated prices against the market?” In: *European Review* (2009), pp. 1–31.

[63] G. S. Ramachandran, R. Radhakrishnan, and B. Krishnamachari. “Towards a Decentralized Data Marketplace for Smart Cities”. In: *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, Sept. 2018.

[64] L. Mikkelsen, K. Mortensen, H. Rasmussen, H.-P. Schwefel, and T. Madsen. “Realization and Evaluation of Marketplace Functionalities Using Ethereum Blockchain”. In: *2018 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC)*. IEEE, Dec. 2018, pp. 47–52.

[65] D.-D. Nguyen and M. I. Ali. “Enabling On-Demand Decentralized IoT Collectability Marketplace using Blockchain and Crowdsensing”. In: *2019 Global IoT Summit (GIoTS)*. June 2019.

[66] H. T. T. Truong, M. Almeida, G. Karame, and C. Soriente. “Towards Secure and Decentralized Sharing of IoT Data”. In: *2019 IEEE International Conference on Blockchain (Blockchain)*. July 2019, pp. 176–183.

[67] S. Bajoudah, C. Dong, and P. Missier. “Toward a Decentralized, Trust-Less Marketplace for Brokered IoT Data Trading Using Blockchain”. In: *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, July 2019, pp. 339–346.

[68] K. R. Ozyilmaz, M. Dogan, and A. Yurdakul. “IDMoB: IoT Data Marketplace on Blockchain”. In: *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. June 2018, pp. 11–19.

[69] P. Tzianos, G. Pipelidis, and N. Tsiamitros. “Hermes: An Open and Transparent Marketplace for IoT Sensor Data over Distributed Ledgers”. In: *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. May 2019, pp. 167–170.

[70] S. Musso, G. Perboli, M. Rosano, and A. Manfredi. “A Decentralized Marketplace for M2M Economy for Smart Cities”. In: *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. June 2019, pp. 27–30.

[71] K. Nguyen, G. Ghinita, M. Naveed, and C. Shahabi. “A Privacy-Preserving, Accountable and Spam-Resilient Geo-Marketplace”. In: *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. New York, NY, USA: ACM, Nov. 2019, pp. 299–308.

[72] A. Seitz, D. Henze, D. Miehle, B. Bruegge, J. Nickles, and M. Sauer. “Fog Computing as Enabler for Blockchain-Based IIoT App Marketplaces - A Case Study”. In: *2018 Fifth International Conference on Internet of Things: Systems, Management and Security*. Oct. 2018, pp. 182–188.

[73] D. Miehle, M. M. Meyer, A. Luckow, B. Bruegge, and M. Essig. “Toward a Decentralized Marketplace for Self-Maintaining Machines”. In: *2019 IEEE International Conference on Blockchain (Blockchain)*. July 2019, pp. 431–438.

[74] V. P. Ranganthan, R. Dantu, A. Paul, P. Mears, and K. Morozov. “A Decentralized Marketplace Application on the Ethereum Blockchain”. In: *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*. Oct. 2018, pp. 90–97.

[75] Z. Wang, L. Yang, Q. Wang, D. Liu, Z. Xu, and S. Liu. "ArtChain: Blockchain-Enabled Platform for Art Marketplace". In: *2019 IEEE International Conference on Blockchain (Blockchain)*. July 2019, pp. 447–454.

[76] J. Martins, M. Parente, M. Amorim-Lopes, L. Amaral, G. Figueira, P. Rocha, and P. Amorim. "Fostering Customer Bargaining and E-Procurement Through a Decentralised Marketplace on the Blockchain". In: *IEEE Transactions on Engineering Management* (2020), pp. 1–15.

[77] N. Baranwal Somy, K. Kannan, V. Arya, S. Hans, A. Singh, P. Lohia, and S. Mehta. "Ownership Preserving AI Market Places Using Blockchain". In: *2019 IEEE International Conference on Blockchain (Blockchain)*. 2019, pp. 156–165. DOI: 10.1109/Blockchain.2019.00029.

[78] J. Li, A. Grintsvayg, J. Kauffman, and C. Fleming. "LBRY: A Blockchain-Based Decentralized Digital Content Marketplace". In: *2020 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*. Aug. 2020, pp. 42–51.

[79] P. Banerjee, C. Govindarajan, P. Jayachandran, and S. Ruj. "Reliable, Fair and Decentralized Marketplace for Content Sharing Using Blockchain". In: *2020 IEEE International Conference on Blockchain (Blockchain)*. Nov. 2020, pp. 365–370.

[80] M. F. Franco, E. J. Scheid, L. Z. Granville, and B. Stiller. "Brain: blockchain-based reverse auction for infrastructure supply in virtual network functions-as-a-service". In: *2019 IFIP Networking Conference, IFIP Networking 2019*. Institute of Electrical and Electronics Engineers Inc., May 2019.

[81] B. Nour, A. Ksentini, N. Herbaut, P. A. Frangoudis, and H. Moungla. "A Blockchain-Based Network Slice Broker for 5G Services". In: *IEEE Networking Letters* 3 (2019), pp. 99–102.

[82] E. J. Scheid, M. Keller, M. F. Franco, and B. Stiller. "BUNKER: A Blockchain-based trUsted VNF pacKagE Repository". In: *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. Vol. 11819 LNCS. Springer International Publishing, 2019, pp. 188–196.

[83] M. Franco, E. Sula, B. Rodrigues, E. Scheid, and B. Stiller. "ProtectDDoS: A Platform for Trustworthy Offering and Recommendation of Protections". In: *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. Vol. 12441 LNCS. Springer International Publishing, 2020, pp. 28–40.

[84] V. Arya, S. Sen, and P. Kodeswaran. "Blockchain Enabled Trustless API Marketplace". In: *2020 International Conference on COMmunication Systems and NETworkS, COMSNETS 2020*. Institute of Electrical and Electronics Engineers Inc., Jan. 2020, pp. 731–735.

[85] M. Pincheira, M. Vecchio, and R. Giaffreda. "Rationale and Practical Assessment of a Fully Distributed Blockchain-based Marketplace of Fog/Edge Computing Resources". In: *2020 Seventh International Conference on Software Defined Systems (SDS)*. Apr. 2020, pp. 165–170.

[86] EU Parliament. *Regulation (EU) 2016/679 of the European Parliament (General Data Protection Regulation)*. 2016, pp. 1–99. URL: https://gdpr-info.eu/ (visited on 06/18/2023).

[87] M. Matteucci, D. Raponi, M. Mengoni, and M. Peruzzini. "Tangible Augmented Reality Model to Support Manual Assembly". In: *13th ASME/IEEE International Conference on Mechatronic and Embedded Systems and Applications*. American Society of Mechanical Engineers, Aug. 2017.

[88] Ying Zou, Qi Zhang, and Xulin Zhao. "Improving the Usability of E-Commerce Applications using Business Processes". In: *IEEE Transactions on Software Engineering* (Dec. 2007), pp. 837–855.

[89] Y. Huang, Y. Chai, Y. Liu, and J. Shen. "Architecture of next-generation e-commerce platform". In: *Tsinghua Science and Technology* (Feb. 2019), pp. 18–29.

[90] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, and T. Ahmed. "Scheduling Wireless Virtual Networks Functions". In: *IEEE Transactions on Network and Service Management* (June 2016), pp. 240–252.

[91] R. Robert. *A decentralized marketplace with Hyperledger Fabric*. 2020. URL: https://www.ericsson.com/en/blog/2020/5/a-decentralized-marketplace-with-hyperledger-fabric (visited on 06/18/2023).

[92] A. Futoransky, C. Sarraute, D. Fernandez, M. Travizano, and A. Waissbein. "Fair and Decentralized Exchange of Digital Goods". In: (Feb. 2020). arXiv: 2002.09689.

[93] XBR. *Open Data Markets Infrastructure*. 2019.

[94] M. Van Niekerk and R. Veer. *Global Market for Local Data*. 2018. URL: https://www.allcryptowhitepapers.com/wp-content/uploads/2018/11/Databroker-DAO.pdf (visited on 06/18/2023).

[95] R. Haenni. *Datum Network: The Decentralized Data Marketplace*. 2017. URL: https://datum.org/ (visited on 06/18/2023).

[96] M. Davidsen, S. Gajek, M. Kruse, and S. Thomsen. *Empowering the Economy of Things*. 2017. URL: https://weeve-network.vercel.app/whitepaper (visited on 06/18/2023).

[97] CBAN. *Working Draft CBAN Reference Architecture*. Tech. rep. 2020.

[98] P. Häfner, V. Häfner, and J. Ovtcharova. "Teaching Methodology for Virtual Reality Practical Course in Engineering Education". In: *Procedia Computer Science* (2013), pp. 251–260.

[99] U. C. Pendit, M. B. Mahzan, M. D. Fadzly Bin Mohd Basir, M. Bin Mahadzir, and S. N. binti Musa. "Virtual reality escape room: The last breakout". In: *2017 2nd International Conference on Information Technology (INCIT)*. Nov. 2017.

[100] ETSI. *ETSI GR PDL 001: Permissioned Distributed Ledger (PDL); Landscape of Standards and Technologies*. 2020.

[101] TM Forum. *Blockchain-based Telecom Infrastructure Marketplace*. 2019. URL: https://www.tmforum.org/blockchain-based-telecom-infrastructure-marketplace/.

[102] P. Seltsikas and H. van der Heijden. “A Taxonomy of Government Approaches Towards Online Identity Management”. In: *2010 43rd Hawaii International Conference on System Sciences*. 2010, pp. 1–8. DOI: 10.1109/HICSS.2010.38.

[103] A. G. Revar and M. D. Bhavsar. “Securing user authentication using single sign-on in Cloud Computing”. In: *2011 Nirma University International Conference on Engineering*. 2011, pp. 1–4. DOI: 10.1109/NUiConE.2011.6153227.

[104] C. Ribeiro, H. Leitold, S. Esposito, and D. Mitzam. “STORK: a real, heterogeneous, large-scale eID management system”. In: *International Journal of Information Security* (Oct. 2018), pp. 569–585.

[105] T. Zhou, X. Li, and H. Zhao. “EverSSDI: Blockchain-based framework for verification, authorisation and recovery of self-sovereign identity using smart contracts”. In: *International Journal of Computer Applications in Technology* (2019), pp. 281–295.

[106] Y. Liu, Q. Lu, H.-Y. Paik, X. Xu, S. Chen, and L. Zhu. “Design Pattern as a Service for Blockchain-Based Self-Sovereign Identity”. In: *IEEE Software* (Sept. 2020), pp. 30–36.

[107] G. Wood. *Polkadot: Vision for a Heterogeneous Multi-Chain Framework*. 2017. URL: https://polkadot.network/PolkaDotPaper.pdf.

[108] H. Gulati and C.-T. Huang. “Self-Sovereign Dynamic Digital Identities based on Blockchain Technology”. In: *2019 SoutheastCon*. IEEE, Apr. 2019.

[109] Z. Cui, F. Xue, S. Zhang, X. Cai, Y. Cao, W. Zhang, and J. Chen. “A Hybrid BlockChain-Based Identity Authentication Scheme for Multi-WSN”. In: *IEEE Transactions on Services Computing* (2020).

[110] R. Soltani, U. Trang Nguyen, and A. An. “A New Approach to Client Onboarding Using Self-Sovereign Identity and Distributed Ledger”. In: *2018 IEEE iThings and IEEE GreenCom and IEEE CPSCom and IEEE SmartData*. July 2018, pp. 1129–1136.

[111] D. Li, W. E. Wong, and J. Guo. “A Survey on Blockchain for Enterprise Using Hyperledger Fabric and Composer”. In: *2019 6th International Conference on Dependable Systems and Their Applications (DSA)*. IEEE, Jan. 2020, pp. 71–80.

[112] A. Othman and J. Callahan. “The Horcrux Protocol: A Method for Decentralized Biometric-based Self-sovereign Identity”. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. July 2018.

[113] R. Soltani, U. T. Nguyen, and A. An. “Practical Key Recovery Model for Self-Sovereign Identity Based Digital Wallets”. In: *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing (DASC)*. Aug. 2019, pp. 320–325.

[114] S. K. Gebresilassie, J. Rafferty, P. Morrow, L. Chen, M. Abu-Tair, and Z. Cui. “Distributed, Secure, Self-Sovereign Identity for IoT Devices”. In: *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*. June 2020, pp. 1–6.

[115] M. P. Bhattacharya, P. Zavarsky, and S. Butakov. “Enhancing the Security and Privacy of Self-Sovereign Identities on Hyperledger Indy Blockchain”. In: *2020 International Symposium on Networks, Computers and Communications (ISNCC)*. Oct. 2020.

[116] C. Lundkvist, R. Heck, J. Torstensson, Z. Mitton, and M. Sena. *Uport: a Platform for Self Sovereign Identity*. 2016. URL: http://blockchainlab.com/pdf/uPort_whitepaper_DRAFT20161020.pdf.

[117] Z. Zhao and Y. Liu. "A Blockchain based Identity Management System Considering Reputation". In: *2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE)*. IEEE, Sept. 2019, pp. 32–36.

[118] P. C. Bartolomeu, E. Vieira, S. M. Hosseini, and J. Ferreira. "Self-Sovereign Identity: Use-cases, Technologies, and Challenges for Industrial IoT". In: *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Sept. 2019, pp. 1173–1180.

[119] C. Allen. *The Path to Self-Sovereign Identity*. Tech. rep. CA, USA, 2016. URL: http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html.

[120] A. Abraham. "A position paper on blockchain enabled identity and the road ahead". In: *Blockchain Bundesverband* (Oct. 2017), pp. 1–39.

[121] P. Wuille. *BIP-0032 – Hierarchical Deterministic Wallets*. 2012. URL: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki.

[122] W3C. *Decentralized Identifiers (DIDs) v1.0*. 2021. URL: https://w3c.github.io/did-core/.

[123] R. Ansey, J. Kempf, O. Berzin, C. Xi, and I. Sheikh. "Gnomon: Decentralized Identifiers for Securing 5G Iot Device Registration and Software Update". In: *2019 IEEE Globecom Workshops (GC Wkshps)*. Dec. 2019.

[124] P. Ekparinya, V. Gramoli, and G. Jourjon. "Impact of Man-In-The-Middle Attacks on Ethereum". In: *2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS)*. Oct. 2018, pp. 11–20.

[125] N. Naik and P. Jenkins. "uPort Open-Source Identity Management System: An Assessment of Self-Sovereign Identity and User-Centric Data Platform Built on Blockchain". In: *2020 IEEE International Symposium on Systems Engineering (ISSE)*. Oct. 2020.

[126] F. Wessling, C. Ehmke, M. Hesenius, and V. Gruhn. "How much blockchain do you need?" In: *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*. New York, NY, USA: ACM, May 2018, pp. 44–47.

[127] R. A. Mishra, A. Kalla, N. A. Singh, and M. Liyanage. "Implementation and Analysis of Blockchain Based DApp for Secure Sharing of Students' Credentials". In: *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*. Jan. 2020.

[128] B. Houtan, A. S. Hafid, and D. Makrakis. "A Survey on Blockchain-Based Self-Sovereign Patient Identity in Healthcare". In: *IEEE Access* (2020), pp. 90478–90494.

[129] B. Dab, I. Fajjari, M. Rohon, C. Auboin, and A. Diquelou. “An Efficient Traffic Steering for Cloud-Native Service Function Chaining”. In: *2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*. Feb. 2020, pp. 71–78.

[130] B. Lee, A. Awad, and M. Awad. “Towards Secure Provenance in the Cloud: A Survey”. In: *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing, UCC 2015* (2015), pp. 577–582.

[131] M. Montecchi, K. Plangger, and M. Etter. “It’s real, trust me! Establishing supply chain provenance using blockchain”. In: *Business Horizons* (May 2019), pp. 283–293.

[132] M. Demir, O. Turetken, and A. Ferwom. “Blockchain and IoT for Delivery Assurance on Supply Chain (BIDAS)”. In: *2019 IEEE International Conference on Big Data (Big Data)*. Dec. 2019, pp. 5213–5222.

[133] C. A. Alexander and L. Wang. “Cybersecurity, Information Assurance, and Big Data Based on Blockchain”. In: *2019 SoutheastCon*. 2019, pp. 1–7. DOI: 10.1109/SoutheastCon42311.2019.9020582.

[134] S. Wibowo and T. Sandikapura. “Improving Data Security, Interoperability, and Veracity using Blockchain for One Data Governance, Case Study of Local Tax Big Data”. In: *2019 International Conference on ICT for Smart Society (ICISS)*. Nov. 2019.

[135] F. Alam Khan, M. Asif, A. Ahmad, M. Alharbi, and H. Aljuaid. “Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development”. In: *Sustainable Cities and Society* (Apr. 2020).

[136] C. Esposito, A. De Santis, G. Tortora, H. Chang, and K.-K. R. Choo. “Blockchain: A Panacea for Healthcare Cloud-Based Data Security and Privacy?” In: *IEEE Cloud Computing* (Jan. 2018), pp. 31–37.

[137] R. Neisse, G. Steri, and I. Nai-Fovino. “A Blockchain-based Approach for Data Accountability and Provenance Tracking”. In: *Proceedings of the 12th International Conference on Availability, Reliability and Security*. New York, NY, USA: ACM, Aug. 2017.

[138] X. Liang, J. Zhao, S. Shetty, and D. Li. “Towards data assurance and resilience in IoT using blockchain”. In: *2017 IEEE Military Communications Conference (MILCOM)*. Oct. 2017, pp. 261–266.

[139] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla. “ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability”. In: *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. May 2017, pp. 468–477.

[140] A. Patil, A. Jha, M. M. Mulla, N. D.G., and S. Kengond. “Data Provenance Assurance for Cloud Storage Using Blockchain”. In: *2020 International Conference on Advances in Computing, Communication & Materials (ICACCM)*. 2020, pp. 443–448. DOI: 10.1109/ICACCM50413.2020.9213032.

[141] J. Benet. *IPFS - Content Addressed, Versioned, P2P File System*. July 2014. arXiv: 1407.3561.

[142] A. Taha, A. Zakaria, D. Kim, and N. Suri. "Decentralized Runtime Monitoring Approach Relying on the Ethereum Blockchain Infrastructure". In: *2020 IEEE International Conference on Cloud Engineering (IC2E)*. 2020, pp. 134–143. DOI: 10.1109/IC2E48712.2020.00021.

[143] R. Angarita, A. Dejous, and P. Blake. "From centralized to decentralized blockchain-based product registration systems: the use case of lighting and appliances". In: *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 2019, pp. 650–655. DOI: 10.1109/INFCOMW.2019.8845267.

[144] *Bitcoin is not ruled by miners*. 2017. URL: https://en.bitcoin.it/wiki/Bitcoin_is_not_ruled_by_miners.

[145] M. Liu, K. Wu, and J. J. Xu. "How Will Blockchain Technology Impact Auditing and Accounting: Permissionless versus Permissioned Blockchain". In: *Current Issues in Auditing* 13.2 (2019). DOI: 10.2308/ciia-52540.

[146] R. van Pelt, S. Jansen, D. Baars, and S. Overbeek. "Defining Blockchain Governance: A Framework for Analysis and Comparison". In: *Information Systems Management* (2021), pp. 21–41. DOI: 10.1080/10580530.2020.1720046.

[147] S. DiRose and M. Mansouri. "Comparison and Analysis of Governance Mechanisms Employed by Blockchain-Based Distributed Autonomous Organizations". In: *2018 13th Annual Conference on System of Systems Engineering (SoSE)*. June 2018, pp. 195–202.

[148] E. Duffield and D. Diaz. *Dash: A Payments-Focused Cryptocurrency*. 2018. URL: https://github.com/dashpay/dash/wiki/Whitepaper.

[149] Sudhir Khatwani. *What is a BIP (Bitcoin Improvement Proposal)? Why do you need to know about it?* 2017. URL: https://coinsutra.com/bip-bitcoin-improvement-proposa/.

[150] G. Andresen. *Bitcoin Improvement Process 101*. 2015. URL: https://github.com/bitcoin/bips/blob/master/bip-0101.mediawiki.

[151] Dash. *Governance — Dash documentation*. 2016. URL: https://docs.dash.org/en/stable/governance/.

[152] E. Duffield. *Block Size Limitation Increase*. 2016. URL: https://www.dashcentral.org/p/2mb-blocksize.

[153] M. Baudlet, D. Fall, Y. Taenaka, and Y. Kadobayashi. "The Best of Both Worlds: A New Composite Framework Leveraging PoS and PoW for Blockchain Security and Governance". In: *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*. Sept. 2020, pp. 17–24.

[154] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang. "Decentralized Autonomous Organizations: Concept, Model, and Applications". In: *IEEE Transactions on Computational Social Systems* (Oct. 2019), pp. 870–878.

[155] F. Freitag. “On the Collaborative Governance of Decentralized Edge Microclouds with Blockchain-Based Distributed Ledgers”. In: *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. Dec. 2018, pp. 709–712.

[156] K. Singi, V. Kaulgud, R. J. C. Bose, and S. Podder. “CAG: Compliance Adherence and Governance in Software Delivery Using Blockchain”. In: *2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*. May 2019, pp. 32–39.

[157] R. Lekh and Pooja. “Exhaustive study of SDLC phases and their best praxctices to create CDP model for process improvement”. In: *2015 International Conference on Advances in Computer Engineering and Applications*. Mar. 2015, pp. 997–1003.

[158] M. Bhagwat, J. C. Shah, A. Bilimoria, P. Parkar, and D. Patel. “Blockchain to improve Academic Governance”. In: *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. July 2020.

[159] X. Liu, X. Sun, and G. Huang. “Decentralized Services Computing Paradigm for Blockchain-Based Data Governance: Programmability, Interoperability, and Intelligence”. In: *IEEE Transactions on Services Computing* (2019), pp. 343–355.

[160] H.-Y. Paik, X. Xu, H. M. N. D. Bandara, S. U. Lee, and S. K. Lo. “Analysis of Data Management in Blockchain-Based Systems: From Architecture to Governance”. In: *IEEE Access* (2019).

[161] E. B. Sifah, H. Xia, C. N. A. Cobblah, Q. Xia, J. Gao, and X. Du. “BEMPAS: A Decentralized Employee Performance Assessment System Based on Blockchain for Smart City Governance”. In: *IEEE Access* (2020).

[162] H. Cheng. “Research on the Distributed Photovoltaic Trading and Settlement Model Based on the Energy Blockchain”. In: *2019 IEEE International Conference on Power Data Science (ICPDS)*. 2019, pp. 59–62. DOI: 10.1109/ICPDS47662.2019.9017201.

[163] K. Nakayama, R. Moslemi, and R. Sharma. “Transactive Energy Management with Blockchain Smart Contracts for P2P Multi-Settlement Markets”. In: *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. Feb. 2019.

[164] S.-V. Oprea, A. Bara, and A. I. Andreescu. “Two Novel Blockchain-Based Market Settlement Mechanisms Embedded Into Smart Contracts for Securely Trading Renewable Energy”. In: *IEEE Access* (2020).

[165] C. Liu, Y. Xiao, V. Javangula, Q. Hu, S. Wang, and X. Cheng. “NormaChain: A Blockchain-Based Normalized Autonomous Transaction Settlement System for IoT-Based E-Commerce”. In: *IEEE Internet of Things Journal* (2019), pp. 4680–4693. DOI: 10.1109/JIOT.2018.2877634.

[166] S. Dangalla, C. Lakmal, C. Wickramarathna, C. Herath, G. Dias, and S. Fernando. “Measuring the Correlation of Personal Identity Documents in Structured Format”. In: *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*. June 2018, pp. 240–245.

[167] Z. Saquib, S. Dwivedi, and A. Dubey. “Electronic authentication for e-Government services - a survey”. In: *10th IET System Safety and Cyber-Security Conference 2015*. Institution of Engineering and Technology, 2015.

[168] X. Wang, X. Xu, L. Feagan, S. Huang, L. Jiao, and W. Zhao. “Inter-Bank Payment System on Enterprise Blockchain Platform”. In: *IEEE International Conference on Cloud Computing, CLOUD* (2018), pp. 614–621.

[169] M. Zouina and B. Outtai. “Towards a distributed token based payment system using blockchain technology”. In: *2019 International Conference on Advanced Communication Technologies and Networking (CommNet)*. Apr. 2019.

[170] The Law Library of Congress. *Regulation of Cryptocurrency Around the World*. June 2018. URL: `https://www.loc.gov/law/help/cryptocurrency/world-survey.php#compsum`.

[171] Monika and R. Bhatia. “Interoperability Solutions for Blockchain”. In: *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*. Oct. 2020, pp. 381–385.

Chapter Eight

Towards Efficient Privacy and Trust in Decentralized Blockchain-Based Peer-to-Peer Renewable Energy Marketplace

Abstract

Renewable energy sources are becoming increasingly important as a substitute for fossil energy production. However, distributed renewable energy production faces several challenges regarding trading and management, such as inflexible pricing models and inaccurate green consumption information. A decentralized peer-to-peer (P2P) electricity marketplace may address these challenges. It enables prosumers to market their self-produced electricity. However, such a marketplace needs to guarantee that the transactions follow market rules and government regulations, cannot be manipulated, and are consistent with the generated electricity. One of the ways to provide these guarantees is to leverage blockchain technology. This work describes a decentralized blockchain-based P2P energy marketplace addressing privacy, trust, and governance issues. It uses a private permissioned blockchain Hyperledger Fabric (HF) and its smart contracts to perform energy trading settlements. The suggested P2P marketplace includes a particular regulator actor acting as a governmental representative overseeing marketplace operations. In this way, the suggested P2P marketplace can address the governance issues needed in electricity marketplaces. Further, the proposed marketplace ensures actors' data privacy by employing HF's private data collections while preserving the integrity and auditability of all operations. We present an in-depth performance evaluation and provide insights into the security and privacy challenges emerging from such a marketplace. The results demonstrate that partial centralization by the applied regulator does not limit the P2P energy trade settlement execution. Blockchain technology allows for automated marketplace operations enabling better incentives for prosumer electricity production. Finally, the suggested marketplace preserves the user's privacy when P2P energy trade settlements are conducted.

8.1 Introduction

Recently, with the widespread adoption of renewable energy sources (RES), *e. g. ,* solar panels and on-premise batteries, the distributed electricity generation in electrical grid infrastructure has opened great opportunities for *prosumers*, *i. e. ,* producers / consumers, of electricity [1]. Electricity generation allows prosumers not only to consume electricity as a conventional node in an energy grid but also to produce and trade it through an *energy marketplace*

by becoming a *distributed energy resource* (DER) [2]. However, this change has led to the introduction of a number of challenges in energy marketplace management.

The first is the *inflexible pricing model* of today's energy marketplaces in which the prosumer is limited to selling electricity to a single buyer (typically the energy provider he/she belongs to), and also to a price set by the buyer or governmental body. This creates a limitation for energy market customers. This not only limits the volume of buyers that a seller can reach but also prevents buyers from purchasing energy from prosumers belonging to a different energy provider. Further, it results in a value distribution imbalance, where the prosumer side suffers monetarily [3].

The second is the *suboptimal electricity distribution*, *i. e.*, when the prosumer generates more electricity than the energy grid can consume. In such a case, the overproduced electricity can be wasted due to insufficient demand within the energy grid. In this case, it is an *energy provider*, *i. e.*, prosumer's supplier of electricity, that manages the supply and demand balance within the marketplace. Using traditional methods, electricity supply management becomes complicated with the increasing numbers of DERs. In order to conduct proper management and accounting of the electricity and money flows in and between its networks, the energy provider requires a common secure database, where generation data is aggregated and can be further analyzed to discover optimal energy distribution strategies. In addition, such a common database can be interoperable with the systems of other energy providers to enable cross-provider efficient energy distribution [4].

The third is the *inaccurate consumption information*, *i. e.*, when buyers do not receive reliable information on the sources of electricity they consume. Due to inaccurate national regulatory frameworks, consumers frequently end up using electricity from fossil fuel energy sources while guaranteed to be using energy generated by RES. Nowadays, the information about the energy produced by RES is contained in the *guarantee of origin* (GO). According to *Directive 2018/2001* (D2018/2001) *of the European Parliament*, GO is *"an electronic document which has the sole function of providing proof to a final customer that a given share or quantity of energy was produced from renewable sources"* [5]. Within the marketplace, the GO is issued by the governmental *regulator*, *i. e.*, body that can certify RES generated electricity. However, as was investigated in [6], due to the absence of secure and unified storage for GOs, consumers still frequently end up using the electricity from fossil fuel energy sources while having the GO.

These limitations can be alleviated by introducing an energy marketplace allowing the *peer-to-peer* (P2P) energy trading between prosumers. According to D2018/2001, a *"peer-to-peer trading of renewable energy means the sale of renewable energy between market participants by means of a contract with pre-determined conditions governing the automated execution and settlement of the transaction..."* [5]. With P2P energy trading, prosumers are able to trade energy directly with each other, controlling where, when, and for what price[1] they sell or buy the electricity. Further, the energy provider controls the electricity generation and consumption data, preserving the privacy of the money flows between the prosumers. Finally, the regulator can be part of the marketplace system and issue GO to a unified and secure database accessible to prosumers during trade operations. Having all parts of

[1] The price margins are regulated by the government. However, these margins leave room for negotiation for the prosumer to make a higher or lower profit on generated electricity.

trade operation within a marketplace, *i. e. ,* generated energy, GO, and agreed upon contract, automated execution and settlement of the trade transactions can be achieved through automation, and orchestration techniques [7]. Ultimately, such a P2P energy trade should promote the installation of DERs by energy grid end-users, resulting in the decarbonization of energy distribution systems and widespread adoption of RES [8].

In today's systems with the *centralized architecture*, the energy marketplace acts as a middleman in all the trading operations and needs to be operated by a *trusted third-party* (TTP) to guarantee that it respects pre-determined conditions of P2P transactions and faithfully follows the orders from the prosumers. In such a case, the prosumers are required to trust their energy provider with the proper operation of the marketplace, making the provider de-facto TTP. *However, scaling this approach to more than one energy provider would raise trust issues.* In a capital economy, it is very likely that energy providers are privately-owned corporations that compete for revenue. Therefore, they have an incentive to keep their operations confidential to gain competitive advantages. The lack of transparency may cast doubts regarding fair play and conformance to negotiated conditions. Therefore, this would require the introduction of an external TTP if no single energy provider can be trusted by all the other actors. Besides, any energy provider acting as a middleman would be in a prime position to disproportionately profit from running the marketplace, creating an asymmetry in value generation [9]. An alternative solution is the adoption of a *decentralized marketplace architecture* that distributes the control of the marketplace over multiple organizations, e.g., the energy providers. Instead of trusting a single TTP with the operation of the marketplace, all the organizations are collectively running it following protocols that guarantee its correct operation [10].

The motivation of this study is to design, implement, and conduct a performance evaluation of a decentralized energy marketplace system that considers all actors, assets, and services present in power grids nowadays. Such a marketplace would enable user authentication, data privacy, decentralized marketplace governance, and the ability to automate trade settlement business logic, making P2P transactions possible. In addition, as the energy distribution and trade price are primarily regulated by the government, to effectively address today's energy systems, the *regulator* actor is introduced. It acts as a governmental representative within the decentralized marketplace and assures that all RESs are certified for usage by prosumers. In addition, it participates in energy trade transactions to ensure correct mapping between produced and traded electricity. To the best of our knowledge, the regulator role was not studied in the context of any other decentralized energy marketplace.

Based on the discussed energy marketplace challenges and outlined motivations, the main contributions of this paper can be summarized as follows. This study proposes a decentralized blockchain-based P2P energy marketplace and utilizes *Hyperledger Fabric* (HF) [11] as the main trust-enabling and consensus-reaching platform. The proposed marketplace uses *smart contracts* (SC), *i. e. ,* chaincodes, to automate and execute the trade settlement process and to manage the issue and consumption of GOs. In addition, it incorporates the *regulator* actor, which acts as a governmental representative and oversees the marketplace operations. Further, it ensures marketplace data privacy while preserving the integrity and auditability of all operations. The following methodology was used to define the blockchain-based energy

marketplace. First, with advice from an operating energy provider, we define a set of regulatory and operational requirements which have to be met by the marketplace. Next, based on defined requirements, we describe the proposed marketplace architecture. Further, we produce the marketplace's threat model and define security requirements. Next, we detail the implementation and map security requirements to appropriate countermeasures provided by HF. Further, we present the marketplace's performance evaluation with the SC tailored for energy trading. Finally, we provide insights and observations regarding the mechanisms that lead to efficient trust and consensus, *e. g.,* by better handling concurrent transactions in the blockchain-based marketplace.

The remainder of the paper is structured as follows. Section 8.2 describes the actors and requirements for the proposed decentralized blockchain-based energy marketplace. Section 8.3 details marketplace's architecture, security analysis, and operation. Section 8.4 details marketplace's implementation. Section 8.5 details the performance evaluation process and results. Section 8.6 describes the results and observations from marketplace implementation and execution of blockchain-based trade settlement. Section 8.7 describes the related work on the blockchain-based energy marketplaces and the performance evaluation of blockchain-based implementations. Finally, Section 8.8 draws a summary of the blockchain-based energy marketplace and provides an outlook.

8.2 Marketplace Requirements

To operate correctly, a decentralized P2P energy marketplace has to precisely define collaborating actors, *i. e.,* entities within the automated system. In addition, for the trade settlement contracts to be defined and executed according to the predefined conditions, the marketplace and the electricity trading process need to offer specific functionalities and fulfill some constraints. These elements were captured in the form of functional and non-functional requirements. The actors and requirements were defined in collaboration with the authors' local energy provider, which has DERs as part of its grid infrastructure and has been working on defining an energy marketplace. Further, the requirements were formulated in compliance with regulations described in D2018/2001 of the European Parliament regarding the issuing, trading, and consumption of GOs. In addition, marketplace actors and requirements have been defined in recent research works. The closest to our marketplace is the proposal described in [12]. The authors of that study define actors and requirements for the P2P energy marketplace. However, their marketplace does not include a regulator role, GO usage, and data privacy requirements which are intrinsic to energy market systems. To the best of our knowledge, none of the other energy marketplaces are taking into consideration such requirements.

8.2.1 Marketplace Actors

Energy marketplace actors and their respective places in the grid infrastructure are depicted in Figure 8.1. The solid line denotes a *physical or digital* connection between two marketplace components, *i. e.,* they are connected either through the power grid or by the network. The dashed line denotes a *logical* connection, *i. e.,* interaction ability between components

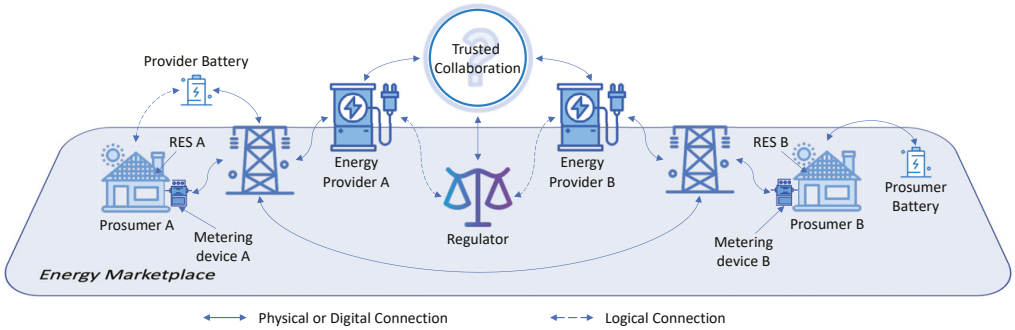


Figure 8.1: *Energy Marketplace Actors.* (The question mark under "Trusted Collaboration" signifies the need to define requirements which are followed by all actors of the marketplace).

through legal obligation or right. Further, each marketplace actor is discussed in detail.

8.2.1.1 Prosumer

The *prosumer* acts as a DER within the marketplace. It has a RES and metering device installed on-premise, *c. f.*, Figure 8.1. While generating the electricity, the prosumer is limited by the aggregator capacity [13], *i. e.*, battery cell. Thus, all generated energy that an aggregator cannot store has to be automatically sold. The energy aggregator may be installed on the prosumer's premise, *i. e.*, household, or be in close proximity, in which case it is owned[2] and operated by the energy provider [14]. While the prosumer is physically connected to the power grid, his/her main interest in becoming a part of the marketplace is to determine the conditions of energy trade settlement with another prosumer, even if they belong to a different energy provider. Further, the prosumers want to get GO for the electricity they produce within the marketplace's automated system. Finally, the prosumers do not want to disclose the electricity generation information to other marketplace participants. However, they are willing to disclose the orders they make, *e. g.*, buy or sell, which should only be visible to the other prosumer in the transaction and to the involved energy providers.

8.2.1.2 Energy Provider

In order to sell electricity, the producer is always connected to the local power grid, which is owned[3] by the *energy provider*. The provider's main interest is to collect data on electricity consumption fluctuations in order to optimize energy distribution and conduct proper accounting of the electricity and money flows in its network. In addition, energy providers do not want to share any information due to market competition. Thus, the local grid generation information from DERs, *i. e.*, prosumers, is private to the energy provider.

[2] There are cases when an energy aggregator is owned by the separate company and is rented for usage to an energy provider. This study assumes that only the energy provider or prosumer can be owners of the aggregator.

[3] The energy provider may not be the owner of the local energy grid. In this case, both the energy provider and grid owner need to become a part of the marketplace. This study assumes that the energy provider is the grid owner.

8.2.1.3 Regulator

In the energy marketplace, a GO is issued by the *regulator*. It is the governmental[4] entity with the legal power to issue a GO to a producer, *i. e.,* prosumer, as proof that the electricity was generated with the RES. During a trade transaction, the prosumer has to present a GO issued by the regulator to guarantee the buyer that the energy was generated with a RES. The GOs are issued for a certain amount of electricity, *e. g.,* 1MWh. In our marketplace design, to make the process flexible for prosumers, the regulator may issue the GO per smaller amounts of generated electricity, *e. g.,* 1kWh.

8.2.2 Functional Requirements (FR)

The *functional requirements* [15] define the functionality that has to be available to prosumers, energy providers, and regulators within the marketplace. These requirements outline a set of operations that comprise a core functionality that may be extended if necessary.

8.2.2.1 FR1 - Electricity Operations

Prosumers should have the ability to conduct *operations with the electricity*, *i. e.,* virtual kilowatt-hours (kWh). Within the marketplace, generated electricity acts as a *fungible token* (FT) [16]. FTs are tokens that are non-unique and represent the same value (in terms of the energy it represents) for every marketplace prosumer, *e. g.,* 1 kWh generated by Prosumer A is equal to 1 kWh generated by Prosumer B. As a marketplace user, the prosumer has to be able to *register generated electricity*, adding a number of kWh to his/her marketplace account. This happens automatically after the prosumer has connected to the marketplace system his/her RES, metering device, and battery if present, *c. f.,* Figure 8.1. Further, by utilizing the marketplace interface, the prosumer should be able to *trade the generated electricity*, *i. e.,* selling/buying electricity at a given price. The associated GO is consumed when the electricity is sold, making it impossible to further sell it to another prosumer. Finally, the prosumer should be able to *consume the bought electricity* for the needs of his/her household. The consumption of bought electricity refers to the balancing of the monthly bill.

8.2.2.2 FR2 - Ordering System Operations

Prosumers should be able to work with the electricity market through a *marketplace ordering system*. First, prosumers should be able to *create an offer to sell electricity* of a given quantity at a given price. Further, prosumers should be able to *buy a given electricity quantity from an offer initiated by another prosumer*. Finally, considering the limitations of an aggregator capacity installed at the prosumer's premise, it should be possible to *automatically sell the remaining quantity of overgenerated electricity to the energy provider at a predefined market price*.

[4] Within the EU, GOs are issued primarily by governmental entities due to regulatory measures. In this paper, the regulator is assumed to be a governmental body.

8.2.2.3 FR3 - GO Operations

The regulator should be able to *issue* the GO on the electricity generated by a specific RES. The GO acts as a *non-fungible token* (NFT) [17] within the marketplace. NFTs are unique objects, which guarantees that within the marketplace, no two GOs are issued for the same generated electricity. When the electricity is generated from a RES, the prosumer must be able to *get the GO*, which can be presented when the energy is sold. Further, the prosumer should be able to *transfer the ownership of the GO* to another entity within a marketplace. This operation is needed in case the GO ownership has to be changed due to legal necessity, *e. g. ,* energy provider becomes the owner of GO due to the prosumer's inability to pay the electricity bill. However, when the electricity is sold, the GO should be *consumed, signifying the depletion of a certain pool of previously generated renewable energy.*

8.2.2.4 FR4 - Aggregator Operations

The marketplace should allow to *register a battery cell*, *i. e. ,* aggregator. The battery cell can be owned either by the prosumer or the energy provider. When installed, the prosumer-owned battery and its capacity must be verified by the energy provider and registered within the marketplace.

8.2.3 Non-Functional Requirements (NR)

The *non-functional requirements* [18] define the global constraints which affect the marketplace system's reliability, usability, efficiency, and maintenance. NRs are divided into application and system level requirements for additional structuring and convenience.

Application level

The first group defines *application-level* NRs, which put additional constraints on FRs discussed above.

8.2.3.1 NR1 - Vitual kWh Data Correctness

The traded electricity within the marketplace represents the virtual kWh which on a digital level is a number in a database. Thus, *virtual kWh must only be issued following the actual generation of electricity and must not be duplicated within the marketplace.* Further, the marketplace system must ensure that *there is a perfect match between the virtual kWh consumed by a household and the actual kWh they consumed from the grid.*

8.2.3.2 NR2 - Order Data Correctness and Guarantees

The energy selling order system has to provide a number of guarantees to both seller and buyer sides of the trade operation. When an order is being executed, the marketplace has to *make sure that an appropriate amount of resources is available* for both sides of the trading operation, *i. e. ,* the electricity on the seller's side and currency on the buyer's side. In addition, the *trade operation has to be executed according to a set of conditions that were*

previously agreed upon by all participating actors, *e. g. ,* along with available resources, a GO must be present on the seller's side to settle the trade.

8.2.3.3 NR3 - GO Data Correctness

The GO acts as an NFT, which also requires a number of guarantees from the marketplace. A reliable checking mechanism must be enforced for the *GO to be only issued following the actual generation of electricity and with the proper source type*, *e. g. ,* hydro, wind, or solar [19]. Further, *when the GO is transferred, its uniqueness has to be preserved* and no duplicated GOs may occur within the marketplace. Finally, *when the GO is consumed, it has to be marked as such to make double consumption impossible.*

8.2.3.4 NR4 - Marketplace Scalability

Considering the scale of energy grid infrastructures, the number of prosumers may vary. Thus, the marketplace has to *scale to a reasonable transaction throughput*[5] based on the volume of orders and produced electricity. Finally, *all orders must come from real prosumers.* The marketplace has to ensure that no fake orders are being issued to take part in trading operations in an attempt to manipulate the market.

System level

The second group defines *system-level* NR, which put constraints on all operations that are executed within the marketplace platform.

8.2.3.5 NR5 - Data Privacy

Considering the sensitivity of the trade transactions, data privacy [20] has to be preserved within the marketplace. *All transactions from a prosumer, including generation, consumption, and purchase, should not be disclosed to other prosumers.* Further, *P2P energy trade details should be disclosed only to those prosumers and their respective energy providers who participate in the transaction.* In addition, *energy trade information has to be visible for the regulator* to ensure correct mapping between traded virtual kWh and GO consumption. Further, *prosumer information has to be visible for the regulator* to ensure correct mapping between virtual kWh and actual generated electricity. Finally, *transactions internal to an energy provider*, *i. e. ,* transaction between a prosumer and its energy provider or between two prosumers within an energy provider, *should not be disclosed to other electricity providers.*

8.2.3.6 NR6 - Marketplace Expandability

The marketplace may span different regions and countries, which requires the support of *a varying number of energy providers, regulators, and prosumers.* First, the marketplace solution must *allow transactions across prosumers belonging to multiple electricity providers.* Further, the marketplace that spans different countries must *support the possibility of multiple*

[5] The reasonable volume of orders and trade transactions depends on the area, the number of energy providers, and the number of potential prosumers wanting to become part of the marketplace.

regulators which issue the GO. Finally, the marketplace system must *allow for effective management of a reasonable number of prosumers*[6] *per energy provider*.

8.2.3.7 NR7 - Marketplace Operation

Aiming at widespread adoption, the marketplace is subject to integration and administration requirements. The need for simple operational processes is due to the system is supposed to be adopted by a large number of actors for whom maintaining such a system is not a core part of their job. Thus, the marketplace solution must *allow for easy onboarding and be straightforward to use by a prosumer*. Further, the marketplace should *have straightforward requirements for the equipment hosted by each prosumer*. Here, the main costs for the prosumer are concentrated in the installation and certification of RES and the metering device. The registration within a marketplace should not involve any additional charges and provides GUI through a web page or smartphone app. Next, the marketplace should *allow for straightforward onboarding and should be reasonably complex to manage for the energy providers*. It should facilitate adoption by energy providers who might have limited competencies and resources to maintain their participation in the marketplace. The marketplace should *minimize the administrative burden put on the regulator*. However, since the regulator is a governmental body, it should conduct the marketplace standardization activities and provide competent staff that conducts onboarding for energy providers. Finally, the marketplace *should allow straightforward management of the life-cycle of the application logic*, *e. g. ,* trade settlement process update among all energy providers. It makes it possible to cater to the market's evolution and the new requirements that might appear or the bugs that need to be fixed.

8.3 Blockchain-based Energy Marketplace

For the energy marketplace, correct operation means that the requirements previously defined are met. All the actors must have guarantees that the trade settlements are executed following rules that have been agreed upon beforehand, maintaining data provenance, and preventing tampering. Blockchain technology [21] can be used to provide the technical building block allowing for meeting these requirements. Blockchain provides marketplace participants with distributed storage, *i. e. ,* the ledger, and brings such benefits as provenance, accountability, and transparency to all data processed in a system. It also acts as a consensus reaching platform, allowing initially non-trusting energy providers and prosumers to establish a trusted relationship [22]. This removes the financial costs associated with having the middleman mediate the trade settlement. Also, it avoids the problem of having to find a TTP accepted by all participants.

[6] The reasonable number of prosumers varies depending on the population density within the marketplace operation region and the number of privately owned RES. Further, the main limiting factor regarding the prosumers amount is the throughput of the marketplace solution and its scalability.

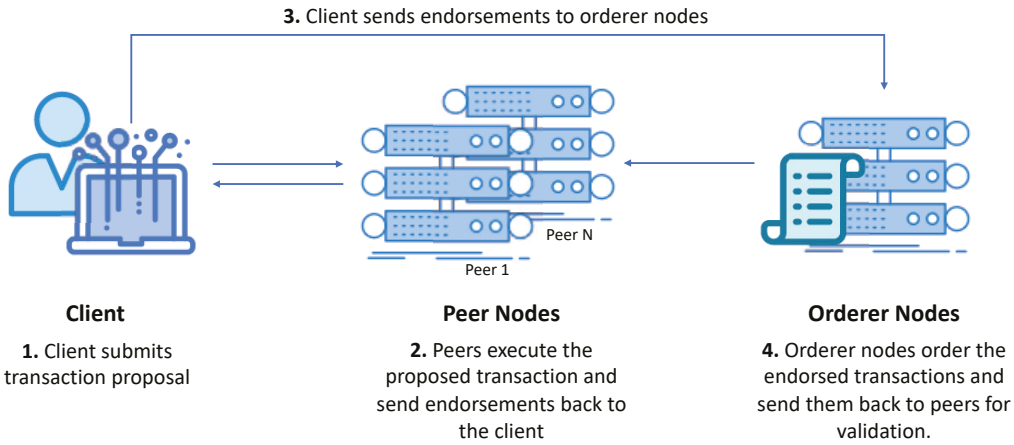


Figure 8.2: *Hyperledger Fabric transaction flow.* (Each number outlines the transaction execution stage).

8.3.1 Blockchain Platform

To effectively address marketplace requirements, first, the blockchain platform has to be chosen. There are several blockchain architectures that allow either *private* or *public* access to the ledger. Considering the NR5, which requires data privacy within the energy marketplace, in our marketplace, we utilize the private permissioned blockchain. A private blockchain has better privacy-preserving characteristics and is better suited to address business needs [23]. Here, it is the governing nodes, *i. e.*, blockchain organizations that participate in the network, that decide if a new participant can enter a private blockchain. Moreover, because it is permissioned after the new participant enters the network, governing nodes decide whether it is allowed to execute new transactions and add them to the ledger. Hyperledger Fabric (HF) [11], which is developed by the Linux Foundation, is representative of a private permissioned blockchain system. All transactions in HF are executed within a *blockchain channel*, which is an entity that establishes a connection between the ledger participants. In HF, there are two types of nodes: *peers* and *orderers*. Further, each node performs a certain type of task: *endorsement* (peers), *ordering* (orderers), or *validation* (peers). HF transaction flow is described in Figure 8.2. Endorsing peers take a transaction, execute it and return a proposal response. Responses from multiple endorsers are then bundled together and passed to the orderer nodes. These nodes take newly endorsed transactions, agree on the order in which these transactions are stored in the ledger, and generate a new block with the ordered transactions. Further, validating peers receive the block that was newly added to the blockchain and check the validity of the transactions in it. Valid transactions must receive endorsements from multiple organizations, according to the configured policy, and must not conflict with previous transactions, *e. g.*, two transactions modify the same data on the blockchain. If a transaction is invalid, it is kept on the blockchain but does not modify its state. Finally, the compiled block is then saved locally in the ledger of every blockchain network participant. All blocks that are saved in HF's ledger are *immutable*, *i. e.*, cannot be changed or removed.

HF peers and orderers are implemented using the *Go*[7] *programming language*. Further, *smart contracts* (SCs), *i. e.,* blockchain executable functions, can be implemented using Go and other programming languages such as Node.js[8] and Java[9]. Peer and SC processes are separated into different execution environments that communicate via *high-performance remote procedure call*[10] (gRPC) technology. The separation of peer and SC execution environments provides several advantages, such as deployment flexibility, modular development, and separation of concerns. Deployment flexibility allows instantiating SCs as a separate process, *e. g.,* can be executed on a different physical machine or docker[11] container. The only requirement is gRPC communication support between peers and SCs. Modular development refers to the decoupling of peer and SC development. This allows implementation and incremental improvements in SCs without affecting peers. Separation of concern refers to the division between the SC management and the functions that it implements, enabling a controlled SC deployment and update [24]. Further, HF defines a *shim*, which acts as an intermediary between the SC and HF environment. A shim allows making the SC accessible to the peer by enabling interactions via gRPC. Further, a shim provides an interface for the SC to access the ledger.

8.3.1.1 Hyperledger Fabric Execution Guarantees

It is assumed that organizations involved in the governance of decentralized system services do not necessarily trust each other, requiring an assurance that the data which is processed on a foreign part of the system is not being manipulated with malicious intent. Thus, in order to support decentralized services execution, a *trust*-enabling mechanism is required, which ensures compliant data processing. First, as services are executed in a decentralized environment, they require distributed storage which enables organizations to maintain an updated system state. A robust and secure *consensus mechanism* enables data synchronization and consistency in distributed storage. Further, decentralized services are defined as automated[12] functions, which all organizations have to approve, *i. e.,* confirm function code, before they are deployed in the infrastructure. The consensus mechanism is used to ensure data correctness and compliant data processing during automated services execution, building trust among organizations in a decentralized system.

The trust that all transactions in the marketplace are following the predefined rules is provided by the *consensus*, *tamper resistance*, and *trusted execution capabilities* of HF. All the rules, *e. g.,* trade settlement contract details, governing the marketplace functions are expressed as a *smart contract* (SC), that is audited by all the blockchain organizations, *i. e.,* energy providers and regulators, and is stored in the ledger [25]. As a result, there is a clear consensus regarding the rules, expressed as computer code, that the transactions need to follow. Every action that the participants can take in the marketplace is implemented solely through the execution of this SC. Hence, relying on the guarantees provided by HF

[7] https://go.dev/
[8] https://nodejs.org/en
[9] https://www.java.com/en/
[10] https://grpc.io/
[11] https://www.docker.com/
[12] Automation is referred to as an ability to define a concrete process block that can be executed as a transaction and result in a new system state, *i. e.,* modifying or creating data in distributed data storage.

that the execution of the SC can be trusted, every marketplace transaction can be trusted to follow the rules. After being executed, all the transactions are stored in the tamper-resistant decentralized ledger. At this stage, the consensus mechanism makes sure that all the participants agree on the list of transactions that are part of the ledger, as well as on their order, maintaining a complete history and providing secure accounting [26].

8.3.1.2 Private Data

A feature of some private blockchain platforms is the ability to store *private data*, *i. e.,* data that is disclosed only to a subset of the organizations in the blockchain. There are two approaches to store private data on HF's ledger: *separate channel* and *private data collections* (PDCs) [27]. The separate channel approach requires deployments of separate ledgers for each private dataset. This approach isolates ledgers from one another and reduces private data verification flexibility, *e. g.,* it cannot be verified if needed by the parties that didn't have access to it initially. This may create a limitation when certain private data has to be used in trade transactions or disclosed due to legal disputes. Instead, the PDC approach allows saving private data in the context of a channel. The PDC participant are able to access, store and modify the data. However, the other channel participants are only storing integrity protection data, i.e. a hash value, allowing to verify the data integrity in case it is disclosed.

8.3.1.3 Hyperledger Fabric Consensus Mechanism

In HF, the consensus mechanism is divided into two layers. The first layer, denotes the *endorse-order-validate* transaction life-cycle. This life-cycle provides the guarantees discussed above as well as trusted SC execution. The second layer of the consensus mechanism is concentrated on the transaction ordering process. Hyperledger Fabric currently only supports the *RAFT* protocol [28] to order transactions. RAFT is *crash fault tolerant* (CFT), *i. e.,* the consensus needs to be executed by one, or more, trusted organizations, a regulator in our case. However, a *byzantine fault tolerant* (BFT) alternative is currently being developed and will allow executing the consensus in a decentralized way, at the expense of performance. Unlike CFT, BFT can handle up to a certain level of failures caused by adversary nodes.

RAFT consensus is used to establish a definitive order of the transactions, between the peers participating in the ledger. It achieves consensus by decomposing the process of block committing to the ledger into several components: leader election, ledger replication, and safety. An elected *leader*, is an orderer node, which receives all endorsed transactions, agrees on their order with *follower* orderers, and sends data back to the peer for committing to the ledger. A leader is elected for an arbitrary *term* at the beginning of blockchain operation. It periodically sends a *heartbeat* request to its follower orderers, to renew its term and maintain leadership. A reelection process starts when one of the followers does not receive a heartbeat request from a leader for a certain period of time, *e. g.,* in the case of failure of the leader orderer node. The *ledger replication* component denotes the normal blockchain network operation when the leader orderer node has been elected. A leader accepts the requests from endorsing peers, groups them into blocks, and replicates them to the follower orderers, in order to keep the ledger synchronized. In RAFT, the majority of orderer nodes must agree

on the transactions order in the block, which is then sent back to the peers for validation and committing to their local ledgers. The *safety* component denotes the procedures of a RAFT consensus protocol that ensure ledger immutability, integrity, and consistency. Here, the leader orderer node has to contain all transactions that were processed during previous terms. This ensures the immutability and consistency of the ledger and allows to elect only the leaders with the most up-to-date ledger. Further, RAFT protocol ensures that at no point in time there are two elected leaders. Such a situation is possible when new orderers join the blockchain network. In this case, a separate consensus agreement is required from both old and new majorities of orderer nodes for the transactions generated during the transition to the new orderer number. After the transition, the new leader is elected, and the blockchain network continues its operation according to configured policy.

8.3.2 Marketplace Architecture

The architecture of the energy marketplace is depicted in Figure 8.3. It consists of two layers: *physical* and *digital*. The physical layer is the actual electrical grid where generated electricity is transported and distributed. The digital layer is the communication network between the energy providers and prosumers where trading of the virtual kWh takes place. Our work is concerned exclusively with the digital layer. With regards to the physical layer, our assumption is the regulator can correctly enforce the mapping of virtual kWh to real generated kWh by combining information from certified metering devices in the electrical grid with transaction information stored in the blockchain. Further, the regulator makes sure that only certified RES and metering devices are used to generate electricity traded in the marketplace.

8.3.2.1 Physical and Digital Layers Mapping

In order to correctly map both marketplace layers, each physical layer actor has to have representation in the digital layer, *c. f. ,* Figure 8.3. *Energy providers* and *regulator* act a separate *blockchain organization* (BO). The BO operates a number of *peer* nodes, which execute endorsement and validation operations of the transaction life-cycle. In addition, peer nodes are the main guarantors of valid transaction execution and require the most computational power. All peers are interconnected, forming the *marketplace channel* (MC), which is an entity that establishes a connection between collaborating BOs. Further, each energy provider BO has a *marketplace interface* (MI), which provides energy prosumers with the necessary functionality to easily join the marketplace and conduct energy trading settlements. In addition, all BOs have a dedicated *membership service provider* (MSP) which generates cryptographic identity information for the prosumers who join the marketplace, *i. e. ,* acts as a certificate authority (CA).

A *regulator*, besides being a BO, maintains the *orderer nodes* and connects them to the MC. The orderer nodes are responsible for the order of transactions in the block, which ultimately guarantees that the ledger contains correctly updated information in case conflicting transactions occur, *e. g. ,* when two transactions try to modify the same record within the ledger [29]. According to HF release documents, separation of endorsement and ordering phases of transaction life-cycle gives HF advantages in performance and scalability

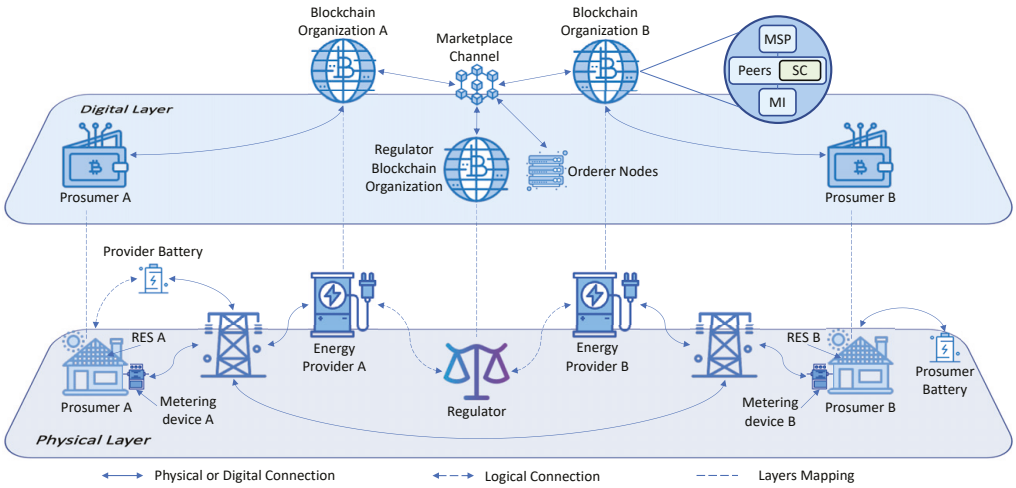


Figure 8.3: *P2P Blockchain-based Energy Marketplace Architecture.* (Physical layer, where actual electricity flows, is mapped to a digital blockchain-based layer, where the electricity trade operations are executed).

by eliminating the bottlenecks which may occur when these two phases are executed on the same node [30]. As orderer nodes are an integral part of the MC and execute-order-validate transaction life-cycle, the entity that operates them can filter, delay, or deny any transactions issued within the blockchain. In addition, the blockchain operation can be disrupted by the shut-down, *i. e.*, due to the DoS attack, of the orderer nodes. **We assume that the regulator, being the governmental body, is not interested in any malicious activity, instead assuring the validity of all transaction ordering processes within the MC.** Finally, the regulator assures the network security of the orderer nodes, providing high availability.

The *prosumers* act as marketplace customers and utilize MI to perform P2P energy trade settlement. As far as prosumers do not act as BO, *i. e.*, have no peers, they do not benefit directly from the ledger guarantees and need to trust the energy providers to endorse and validate trade settlement transactions on their behalf. The prosumers do not run peers practical reasons, since peer installation and operation can put an additional financial burden, *i. e.*, it may act as a discouraging factor when it comes to the decision of whether to become a user of the marketplace. The prosumers have their data records within the blockchain, as well as GOs issued on their generated electricity. They also have a *metering device* installed in their households, which monitors the RES generated electricity and sends the data to the energy provider. It is the responsibility of the energy provider and regulator to make sure that the data on the blockchain matches the actual generated electricity reported by the metering device, *i. e.*, addresses the **NR1**.

8.3.3 Marketplace Security Analysis

From a data privacy perspective, prosumer private data, GO, and trade transaction details are assets at risk. Here, the MI and decentralized blockchain network act as a security

scheme that protects these assets. In case of a successful attack on this security scheme, the adversary may configure the marketplace's functions to act maliciously and present fake electricity generation data and selling orders. To have a complete security analysis, an adversary model is required to outline the capabilities of the adversaries. Further, a threat profile is comprised, to identify final security requirements. This is done next.

8.3.3.1 Adversary Model

This study follows a binary approach used in [31] to construct the adversary model. It divides adversaries into those on the *inside* or the *outside* of the marketplace. Outside adversaries are all unauthorized entities that try to breach the marketplace access control (AC) system. Thus, the protection against outside adversaries is directly related to the robustness of the marketplace's AC system. Inside adversaries are entities with a cryptographic identity within a marketplace and can successfully authenticate. Thus, they can access the marketplace functions and try to tamper with the prosumer's personal data, trade order information, and GO generation. We assume that the *regulator*, as a part of a governmental institution that issues GO, always executes only legitimate actions and does not turn rogue. However, both prosumers and energy providers may act as inside adversaries. In a real-world environment, the inside adversaries may differ in their skills to breach the marketplace's security scheme. In this adversary model, inside adversaries are considered to be *regular users* who do not alter code logic in the security mechanisms, but rather exploit logical flows in the configuration and operation of the system.

Table 8.1: *Identified Security Threats*

Threat	Actor	Description
ST1	Prosumer	Registration of fake virtual kWh without actual electricity generation (breach of NR1).
ST2	Prosumer	Placement of selling orders with invalid false generation data (breach of NR2).
ST3	Prosumer	Illegitimate obtainment of valid GO by bypassing the regulator's checks (breach of NR3).
ST4	Prosumer	Accessing the private transaction data of other prosumers (breach of NR6).
ST5	Energy Provider	Accessing the private data of other energy providers (breach of NR6).
ST6	Energy Provider	Illegitimate production of GO (breach of NR3).
ST7	Outside Adversary	Intercept of traffic and likely tampering of the data exchanged between prosumer and MI (May enable ST1, ST2, ST3, ST4).

8.3.3.2 Threat Model

To comprise system security requirements, comprehensive threat modeling is executed. Here, both inside and outside adversary threats are discussed, and security requirements are detailed.

The inside adversaries consist of authorized prosumers and energy providers. The authorized *prosumers*, when turn rogue, want to manipulate their marketplace tokens, in order to register fake virtual kWh without actual electricity generation. Further, adversary

prosumer wants to place selling orders with modified data, *i. e.,* that are not backed by genuine RES generated electricity, to be able to sell fake virtual kWh to other marketplace actors, *i. e.,* prosumers or energy providers. In order to sell fake virtual kWh, the prosumer wants to obtain valid GO, bypassing the regulator's checks. Finally, adversary prosumer wants to access the private transaction data of other prosumers to gain knowledge of their trade history to obtain competitive advantages for future trade.

Next, the energy provider, being part of the marketplace governance system, when turns rogue, wants to access the private data of other energy providers with the aim to advance in energy market competition. Further, the energy provider wants to obtain the ability to produce fake GO with the aim to sell electricity that comes from a non-renewable source.

An outside adversary may try to intercept the traffic between the prosumer and MI to tamper with the transmitted data. Further, this may enable an outside adversary to operate on the inside, applying all threats that rogue prosumer poses to the system. Table 8.1 provides a condensed description on identified security threats.

Table 8.2: *Threat Profile*

Threat	S	T	R	I	D	E
ST1	X	X				X
ST2		X				X
ST3		X	X			
ST4	X			X		X
ST5	X			X		X
ST6		X	X			
ST7		X		X	X	

With the system security threats identified, *c. f.,* Table 8.1, this study employs STRIDE approach [32] to compile marketplace's *threat profile*, *c. f.,* Table 8.2. In STRIDE threat profiling approach, each letter of the word represent a specific threat category which has a desirable security property against it: *S*poofing *vs.* authentication, *T*ampering *vs.* integrity, *R*epudiation *vs.* non-repudiation, *I*nformation disclosure *vs.* confidentiality, *D*oS *vs.* availability, and *E*levation of privilege *vs.* authorization.

Table 8.3: *Security Requirements*

Req.	Description
SR1	It must be impossible to spoof any type of the transaction that is executed within the marketplace.
SR2	It must be impossible to tamper with any type of the transaction that is executed within the marketplace.
SR3	It must be impossible to repudiate any type of the transaction that is executed within the marketplace.
SR4	No private information disclosure must be possible of any information to an outsider, and to members of the ledger (only within the specific limits specified by NR5).
SR5	Within the smart contract, functions must be restricted based on the user role.
SR6	The traffic between the prosumer and MI has to be encrypted and authenticated, *i. e.,* protected from spoofing and tampering.

Based on the identified threat profile, a number of security requirements (SR) are identified in Table 8.3. Further, the identified security requirements are mapped to the appropriate countermeasures provided by the HF and MI.

8.3.4 Marketplace Regulations

The regulatory measures are defined by all BOs who collaborate within the blockchain-based marketplace. All regulations, including *configured policy (governance), identity management, data privacy*, and *trade settlement conditions* have to be defined in the respective places within the blockchain infrastructure [33] and legal documents which define penalties in case of non-compliance[13].

MC governance is defined in the configuration data structure stored in the ledger. This configuration defines the organizations that are part of the MC, *i. e.,* BOs and orderer nodes. Further, it also defines the endorsement policies, *e. g.,* endorsement policy configuration file (EPCF), which describe the organizations that have to endorse a transaction for it to be valid. Importantly, the configuration also defines the policies that govern how the configuration is modified. Thus, to become a part of the marketplace, the newly added BO has to meet the requirements on the number of peers (minimum one) and enforce the same configuration as all other BOs across the entire MC. Further, the modification of the MC configuration follows a process defined by the built-in SC and using the policies specified in the current version of the configuration. In the case of a non-agreed change in the configuration by any of the BOs, MC does not allow endorsement of transactions from the BO-violator.

The onboarding procedures have to be standardized by the regulator and made available for potential marketplace BOs, *i. e.,* energy providers. This should enable the energy providers to adopt the technologies and processes used by the marketplace. Further, the energy provider itself takes parts in the endorsement of the policies and BO maintenance, which should minimize the administrative burden put on the regulator. This addresses the **NR7**.

The *identity and access management* (IAM) [34] is performed by each individual BO through its respective MSP, *i. e.,* CA, and ensures that only authorized prosumers are able to view their private data and trade their electricity. All identities within HF are based on X.509 digital certificates [35] and consist of public and private key pairs. A client CA application running on the end-user equipment can generate the key pair locally. A digital certificate for the public key can be generated by submitting a certificate sign request (CSR) to CA. This way, the private key never leaves the user premises and cannot be used by CA to spoof the prosumer. In addition, such an approach contributes to prosumers' self-sovereignty, *i. e.,* they are the only owners of their private key. Further, since prosumers do not act as a BO, this opens a possibility for an energy provider to create fake prosumers on the digital layer of the marketplace and try to use it with malicious intent, *e. g.,* register electricity on a fake account and try to trade it with prosumers of another energy provider. Here, it is the regulator who ensures that only real prosumers participate in the marketplace by enforcing the correct mapping of real people and their RES to the marketplace accounts. This addresses the **NR4**.

In order to ensure *data privacy*, addressing **NR5**, the PDC approach was chosen to define all private collections and the actors that have access to them. This is done due to the possibility that some private data has to be disclosed in order to resolve the legal

[13] Such penalties are defined in the legal agreements signed by all collaborating parties when establishing a marketplace. However, this study does not investigate the legal aspect of the marketplace as it focuses on the operations executed on the digital layer.

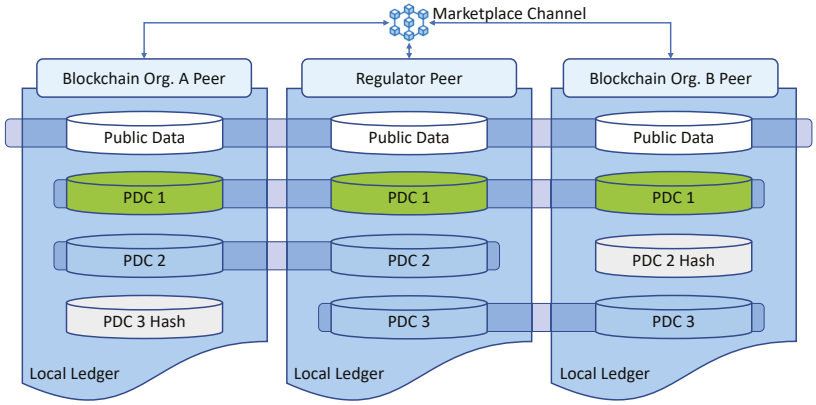


Figure 8.4: *Marketplace private data collections (PDCs).*

dispute between the BOs. With the PDC approach, even though the private data is not available for one or several BOs, they still store the data hash value and are able to verify its integrity on demand. All PDCs are shown in Figure 8.4 and described below. Every data collection is denoted as a cylinder with the respective color. All peers of the marketplace channel have access to *public data* collection, *i. e.*, cylinders with white background, which contains all GOs issued by the regulator and all orders issued by prosumers. This is done to make GOs available for all energy provider peers to conduct energy trade settlements and make orders visible for prosumers to fulfill. However, prosumers can see only their own GO or the ones presented with the electricity selling order. Further, when energy trade settlement is executed between two prosumers within the marketplace, they save it in *PDC 1*, *i. e.*, cylinders with a green background. *PDC 1* data is available only to the specific buyer and seller prosumers involved in the transaction, their respective energy providers, and the regulator. This requires creating PDCs between every two energy providers and the regulator within a marketplace, which may introduce an operational overhead. However, according to NR5, the trade settlement transaction data privacy has to be preserved. The prosumers utilize the GO as a guarantee that a sold or bought amount of electricity was, if fact, generated with RES. In addition, according to NR5, the rest of the energy providers cannot see trade settlement data and save only its hash value. Next, according to NR5, prosumers' data is private for each energy provider. However, in order to guarantee the correct mapping between the virtual kWh and real generated electricity, the regulator has to have access to prosumers' marketplace data, *i. e.*, issue the GO only for certified RESs and metering devices. The *PDC 2* and *PDC 3*, *i. e.*, cylinders with a blue background, contain the data about prosumers', batteries and RES installed within the energy provider's grid. Here, *Energy Provider B* saves only *PDC 2 Hash* value, since the data about prosumers and batteries of *Energy Provider A* is private. In the same manner *Energy Provider A* saves only *PDC 3 Hash* value. Finally, the regulator has access to both *PDC 2* and *PDC 3*. This PDC definition, together with prosumer data read restricted only to his/her own data, addresses the **SR4**.

Finally, the *trade settlement* process is defined within the SC. The SC management and

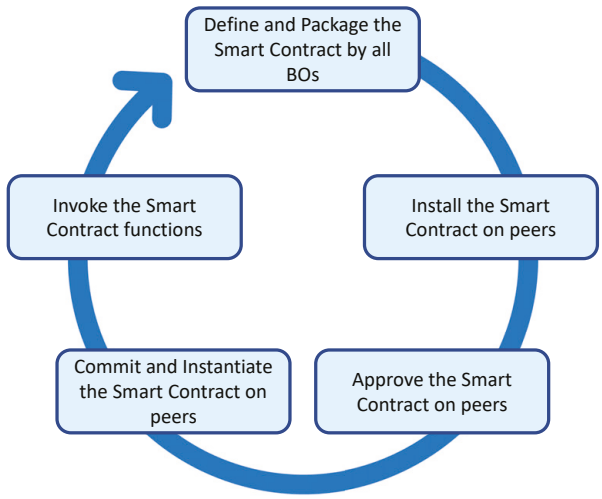


Figure 8.5: *Hyperledger Fabric smart contract deployment life-cycle.*

deployment life-cycle have a number of defined operations that have to be executed by all BOs to be able to invoke any of SC's functions, *c. f.* , Figure 8.5. First, all BOs have to define and package the SC on their premise, *i. e.* , on the machine which has access to the HF tools. Further, each BO has to install and approve the newly compiled SC on their peers. Next, the approved SC has to be committed and instantiated on peers, *i. e.* , SC docker container is started and available for execution. After these operations are performed, the prosumers can invoke the SC contract functions through MI on their respective BOs' peers. This addresses parts of the **NR7**.

The blockchain-based marketplace works as a decentralized system, where data privacy, identity management, and system governance are not located in a single central location. Thus, attacks aimed at spoofing the execution of transactions with tampered data require gaining access to all blockchain participants to perform a full endorse-order-validate chain of operations, which is unlikely. This addresses **SR1** and **SR2**. However, if the aim is to spoof the identity of peer nodes, an improperly implemented MI may become the most vulnerable point of the system, opening opportunities for adversaries to seek ways to execute transactions with tampered data.

8.3.4.1 Data Regulations Compliance

To have the right to operate within certain economic regions, *e. g.* , EU, USA, the marketplace must comply with the data protection and regulation documents. The General Data Protection of Regulation (GDPR) [36] gives a comprehensive definition of the system's obligations towards the customers' private data processing. In addition, it is enforced within the European Union, where D2018/2001 is established and used in this study to identify the requirements for GOs. According to Article 17 of GDPR *"the data subject shall have the right to obtain from the data controller the erasure of personal data concerning him or*

her without undue delay, and the controller shall have an obligation to erase personal data without undue delay...". This forces any system acting within the EU to erase the user data when requested. In HF, it is impossible since the ledger is immutable. Thus, the only way to make HF-based system GDPR compliant is to store no personally identifiable information (PII) [37] within the ledger. Consequently, it requires all PII that the system needs for the operation to be stored in a conventional database, *e. g.,* MySQL[14]. Further, the PII in the conventional database can be linked with the records in HF by storing identifiers of ledger data records. The protection of the conventional database is provided by the respective energy provider that stores prosumer PII. Thus, the security of such database is directly related to the AC mechanisms enforced by the energy provider.

8.3.5 Marketplace Execution Guarantees

In HF, the exact guarantees that are provided depend on the configuration of the ledger network. Regarding the SC execution, it is possible to specify which organizations need to verify and sign every execution, *i. e.,* the EPCF. Requiring more signatures provides stronger execution guarantees, but creates additional execution overhead. A balanced configuration needs to be found, depending not only on the technology but also on the relations between the blockchain participants. In practice, all the participants should be confident that it is unlikely that enough misbehaving parties can collude to hijack and fulfill the endorsing policy in a malicious way. If a misbehaving coalition of BOs were to be able to fulfill the endorsing policy, they would be able to execute transactions without following the SC. And while this would leave irrefutable proofs in the ledger, detecting them would be a tedious manual process. Going further, from the perspective of prosumers, which are not BOs, *i. e.,* they do not operate a peer, they need confidence that there is at least one trusted BO. Indeed, without direct access to the ledger, they need to rely on such a TTP to even just read the SC definition and endorsing policy. Further, if all the blockchain participants are colluding, nothing prevents them from altering the SC or even rewriting the entire ledger history. This is not a problem caused by our system design, but rather a problem shared by all types of private blockchains.

Next, guaranteeing that the virtual energy in the marketplace matches the energy in the distribution network cannot be handled by the blockchain alone. The blockchain itself is incapable of verifying the veracity of the data regarding the energy consumption or generation in the distribution network: this is the oracle problem [38]. What the blockchain can provide is auditability and data provenance, making sure that every virtual kWh recorded as produced or consumed in the blockchain can be traced back to the organization at the origin of the transaction. In case of misbehavior, this creates irrefutable evidence in the ledger allowing to identify the guilty party and handle the incident following the legal agreements established outside of the blockchain. Ultimately the energy provider is the entity with guaranteed real-time access to the correct information and, as a result, must be trusted. However, this could also be enhanced through technical means, *e. g.,* certified meters, as well as through regular independent validation of the data against the records in the ledger. Additionally, another guarantee that the blockchain can provide is that once the generated energy is

[14] https://www.mysql.com/

recorded in the ledger, the SC ensures that it is correctly processed, *e. g. ,* not duplicated or consumed twice.

Finally, the blockchain makes it impossible to manipulate the transactions submitted to the marketplace. In our implementation, this is not completely straightforward because the prosumers are not operating a peer or even interacting directly with the peer using the ledger protocols. Instead they rely on the MI, *c. f. ,* Figure 8.3. As a result, in order to completely prevent the MI from tampering with the received transactions, it is required that the prosumers sign themselves the transaction that will be recorded in the ledger, and with a key that was never disclosed to the MI. Furthermore, because the energy providers control the generation of identity information for the prosumers, they can technically create fake prosumers and subsequently fake trade orders, potentially manipulating the market price. Fraud suspicions could be handled by relying solely on the auditability provided by the blockchain to collect non-repudiable evidence of the misbehavior and prosecute the perpetrator outside of the blockchain. Alternatively, the process for issuance of prosumer identity information can be delegated to an external Credential Service Provider as proposed by NIST 800-63-3[15].

8.4 Marketplace Implementation

Both system design and HF's (version 2.1) technological characteristics are implemented to address all defined marketplace functional and non-functional requirements from Section 8.2. In addition, the security requirements defined in Table 8.3 are addressed within a marketplace SC and MI implementations. HF version 2.1 was the most recent version at the time of writing and based on the changelog, the findings here should still apply to the latest version 2.4.

8.4.1 Blockchain Data Structure

In HF, all assets, *i. e. ,* data records, are represented as *<key, value>* pairs. *Key* acts as a unique data identifier and must not repeat within a blockchain channel ledger. *Value* contains data associated with a specific key and contains all fields that the data record consists of. Further, *value* part may be represented in binary or JSON formats.

A new data structure was developed for the marketplace SC functions implementation. It is based on the concept of data separation to preserve data privacy and allow PDC correct operation. The improper SC function development may result in PDC data leakage [27]. Thus, an improperly defined blockchain data structure may affect SC function execution routines, where PDC leakage may occur.

The prosumer is represented within a marketplace with several logically connected data records that define entities associated with electricity generation and trade, *i. e. ,* prosumer, wallet, res, and aggregator (if the prosumer owns a battery) data records.

[15] https://pages.nist.gov/800-63-3/

Table 8.4: *Prosumer Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Prosumer's unique identifier
Electricity	*Double*	Amount of prosumer generated electricity *(kWh)*
WalletID	*String*	Prosumer's Wallet identifier
AggregatorID	*String*	Prosumer's Aggregator identifier
EnergyProviderID	*String*	Prosumer's Energy Provider identifier
RegulatorID	*String*	Prosumer's Regulator identifier

8.4.1.1 Prosumer Blockchain Data Record

Main prosumer record is described in Table 8.4. It is private for the prosumer, the respective energy provider, and the regulator, *c. f. ,* PDC2 or PDC3 in Figure 9.2. This record contains prosumer unique *ID*, which is generated by the marketplace SC when he/she joins a marketplace. The ID represents the *key* in *<key, value>* pair. It is also random, and non of the prosumer's personal information may be used for its generation (which also applies to any ID field within the marketplace). Further, the prosumer record contains generated *Electricity*, which is updated by the energy provider and regulator based on the data from the prosumer's metering device. In addition, the *WalletID* field connects the prosumer record to a particular wallet within the marketplace. Further, the prosumer record contains *AggregatorID*, which links generated electricity to a particular battery record within the marketplace. The prosumer record also contains the *EnergyProviderID*, which establishes the link with a specific energy provider and is used during automatic electricity selling transactions. Finally, the *RegulatorID* establishes a link between the prosumer and the regulator, which issues GOs on generated electricity.

The prosumer record intentionally does not contain any PII to comply with data regulation documents, *e. g. ,* GDPR. All PII needed for legal purposes can be saved in the conventional DB. Further, the record in conventional DB can contain an ID from the prosumer blockchain record, establishing a link between PII and non-PII saved in the ledger. If a prosumer leaves the marketplace, the PII record can be deleted from the conventional database, *i. e. ,* breaking the link between a real person and a blockchain record. Further, the records in the blockchain can also be marked as deleted. However, the history of deleted records will remain available for reading by the BOs with access to the respective PDC.

Table 8.5: *Wallet Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Wallet's unique identifier
Currency	*Double*	Amount of fiat currency, *e. g. ,* USD, EUR
Electricity	*Double*	Amount of prosumer bought electricity *(kWh)*

8.4.1.2 Wallet Blockchain Data Record

The wallet blockchain data record is described in Table 8.5. It contains the *Currency* field, *i. e. ,* the amount of fiat currency a prosumer has at his/her disposal. It is used during energy trade settlement execution. In addition, the wallet record contains *Electricity* field, which

shows the amount of bought electricity. The wallet record *Electricity* and the prosumer record *Electricity* fields are separated to ensure that the bought electricity is not resold twice. Every prosumer must have only one wallet record attached to it. The wallet record is private to the prosumer, the respective energy provider, and the regulator, *c. f.,* PDC2 or PDC3 in Figure 9.2. However, due to HF's PDC design, during transaction execution, the buyer and seller wallets are visible to BO peers who execute a transaction to conduct trade settlement in a trusted manner. This is a necessary measure for the selling prosumer to ensure that buyer has enough currency in his/her wallet.

Table 8.6: *RES Blockchain Data Record*

Field Name	Type	Description
ID	*String*	RES unique identifier
OwnerID	*String*	RES owner identifier
RegulatorID	*String*	RES certifier regulator identifier
IsCertified	*Boolean*	True if the regulator certified the RES

8.4.1.3 RES Blockchain Data Record

The RES blockchain data record is described in Table 8.6. It is private to the prosumer and the respective energy provider, *c. f.,* PDC2 or PDC3 in Figure 9.2. The RES record contains the *OwnerID* field, which links the RES to a prosumer record within the blockchain. Further, the RES record contains the *RegulatorID*, which links the RES to its certifier regulator. Finally, the *IsCertified* field is set *True* if RES is certified by the regulator. In the marketplace digital layer, the RES data record is represented by the combination of RES and a metering device in the physical layer. The regulator must certify the RES and metering device for them to be connected to the energy grid. Thus, after the certification process, the regulator and the energy provider produce and register a certificate document. As far as such certificate contains private data, due to regulations, *e. g.,* GDPR, it cannot be saved on the immutable ledger. Thus, such a certificate is saved in the conventional DB and contains the RES blockchain data record *ID*.

Table 8.7: *Aggregator (Battery) Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Aggregator unique identifier
OwnerID	*String*	Battery owner identifier
StoredElectricity	*Double*	Electricity currently stored by an Aggregator
MaximumCapacity	*Double*	Maximum amount of stored electricity

8.4.1.4 Aggregator (Battery) Blockchain Data Record

The aggregator (battery) blockchain data record is described in Table 8.7. It contains *OwnerID* field, which links it to the prosumer or energy provider who owns the aggregator. Further, the aggregator data record contains *StoredElectricity* and *MaximumCapacity* fields. The *StoredElectricity* field contains the number of prosumer generated kWh of electricity

stored in the aggregator. The *MaximumCapacity* field describes the maximum possible aggregator capacity. The difference between these two fields, *i. e. , MaximumCapacity - StoredElectricity*, reflects the aggregator's available capacity.

Table 8.8: *Energy Provider Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Energy Provider unique identifier
Description	*String*	Energy Provider description
EnergyPrice	*Double*	Energy Provider price for automatic selling, *(per kWh)*

8.4.1.5 Energy Provider Blockchain Data Record

The energy provider blockchain data record is described in Table 8.8. It contains a *Decription* of the energy provider, *e. g. ,* the data about the peers. This field may contain any descriptive information apart from PII. Further, the *EnergyPrice* field defined the price per kWh of automatically sold electricity. The energy provider may update this field based on the electricity demand during the day. However, the *EnergyPrice* field is verified by the regulator to prevent the energy provider's price undercutting.

Table 8.9: *Regulator Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Regulator unique identifier
Description	*String*	Regulator description
PriceLimit	*Double*	Upper price limit for electricity trade, *(per kWh)*

8.4.1.6 Regulator Blockchain Data Record

The regulator blockchain data record is described in Table 8.9. Like the energy provider, this data record contains *Description* field, where data about the country or regulated geographical area is saved. This field also must not contain any PII. Further, it contains *PriceLimit* field, which describes the upper price limit for the electricity selling per kWh. This is the regulatory measure to prevent electricity overpricing by the prosumers-sellers.

Table 8.10: *GO Blockchain Data Record*

Field Name	Type	Description
ID	*String*	GO unique identifier
OwnerID	*String*	GO owner ID
RegulatorID	*String*	Issuer of GO
RESID	*String*	Certified RES generator of electricity
ElectricityAmount	*Double*	Amount of GO certified electricity *(kWh)*
IsConsumed	*Boolean*	Set *True* when electricity is sold

8.4.1.7 GO Blockchain Data Record

GO is a significant asset for an energy marketplace. The energy trade is impossible without the generated and valid GO. Its blockchain data record is described in Table 8.10. The *OwnerID* field contains an identifier of the prosumer who is the owner of GO. Further, the *RegulatorID* field contains an identifier of the regulator, who is the issuer of GO and guarantor of its validity. Next, the *RESID* field contains an identifier of the RES used to generate the electricity. Such a RES must be certified by the regulator. Further, *ElectricityAmount* field contains the amount of electricity that is certified by the regulator for further trading. Finally, the *isComsumed* field is set *True* when the energy GO was issued for is sold.

Table 8.11: *Order Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Order unique identifier
Type	*String*	Order Type *(Sell or Buy)*
Price	*Double*	Electricity Price *(For the entire amount sold)*
ElectricityAmount	*Double*	Amount of electricity *(kWh)*
GOID	*String*	GO unique identifier
SellerWalletID	*String*	Seller wallet identifier
BuyerWalletID	*String*	Buyer wallet identifier

8.4.1.8 Order Blockchain Data Record

The order data record is central in the electricity trade transactions. Its blockchain data record is described in Table 8.11. The *Type* field shows what kind of order it is, *i. e.,* sell or buy. Further, the *Price* field contains the entire sum of currency that has to be paid for this order. In addition, the *ElectricityAmount* field contains the number of kWh traded and delivered to the buyer's wallet when the order is fulfilled. Next, the *GOID* links the GO data record for the electricity selling order. In the case of buying order, this field is left empty to be filled by the seller. Finally, the *SellerWalletID* and *BuyerWalletID* fields contain identifiers of prosumer wallets. Depending on the type of the order, when it is created, one of the wallet identifiers is left empty, *i. e.,* *SellerWalletID* is empty for a buy order, and *BuyerWalletID* is empty for a sell order. When the order is fulfilled, it is private for prosumers and energy providers participating in trade settlement. In addition, to verify the correctness of the transaction and consume the GO, the order is also visible to the regulator, *c. f.,* PDC1 in Figure 9.2.

8.4.2 Marketplace Smart Contract (SC)

The marketplace SC is the main component allowing the P2P energy trade process automation and orchestration. Each MC participant BO has to agree on the SC definition before it is installed in the channel, *i. e.,* stored in the ledger. Thus, the SC functions must be implemented securely, providing needed data privacy. The marketplace SC is written in GO programming language[16]. In the GO language, the SC code is complex and may be

[16] https://go.dev/

Algorithm 1 Enroll New Prosumer

```
1: function EnrollNewProsumer(prosumerId string, energyProviderId string, regulatorId string, aggregatorId string, prosumerPDC string)
2:     walletId ← [random]                                   ▷ Generate random wallet ID.
3:     wallet ← NewRecord()
4:     wallet.Currency ← 0
5:     wallet.Electricity ← 0
6:     prosumer ← NewRecord()
7:     prosumer.Electricity ← 0
8:     prosumer.WalletID ← walletID
9:     prosumer.AggregatorID ← aggregatorId
10:    prosumer.EnergyProviderID ← energyProviderId
11:    prosumer.RegulatorID ← regulatorId
       ▷ Saving prosumer records in the energy provider and regulator PDCs, c. f. , PDC2 and PDC3 in Figure 9.2.
12:    PutPrivateData(prosumerPDC, walletId, wallet)
13:    PutPrivateData(prosumerPDC, prosumerId, prosumer)
```

hard to read and understand. Thus, this study defines pseudocodes of main SC functions for increased readability and understanding. To address **SR5**, the execution of the SC function is restricted based on the user role, *i. e. ,* prosumer or BO. Further, certain functions within the marketplace are restricted to particular BOs, *e. g. ,* only the regulator BO can issue GOs. The marketplace SC functions and the implemented AC are discussed next.

8.4.2.1 Contract Level Base Functions

The HF contract SDK contains a number of base functions that are used to read and write data assets within the ledger. First, it is the *GetState(key)* function, which reads an asset with a particular id from public data ledger, *c. f. ,* Figure 9.2. Further, the *PutState(key, value)* creates or updates the record on the public data ledger. Next, to read private data saved in the PDC, the *GetPrivateData(collection, key)* function is used. In this function, a *collection* name has to be provided to read the data from the correct PDC. In addition, a peer who tries to read the private data from a particular PDC must have the right to do so according to the collection level endorsement policy, *i. e. ,* the list of BOs who can read from PDC. Next, in order to write to PDC, the *PutPrivateData(collection, key, value)* function is used.

Ultimately, private data introduces additional constraints for the execution of the SC. Indeed, only peers with access to a PDC will be able to execute, and endorse, a transaction reading from this PDC. Hence, the PDC membership and endorsing policies need to be crafted in such a way as the endorsement policies be possible to fulfill by the members of the PDC. Different endorsement policies are applied when public or private data are added or updated, requiring certain groups of BOs to endorse the transaction. Thus, to correctly endorse the transaction where public and private records are modified, the endorser peers need to be communicated some values from the private data as part as the transaction input, even if they are not members of the PDC [27].

The data asset on the ledger or PDC can also be marked as deleted, *i. e. ,* the history of the

Algorithm 2 Register Prosumer Electricity

Require: *Prosumer*, *Energy Provider*, *RES*, *Aggregator*

```
1: function RegisterProsumerElectricity(prosumerId string, resId string, amount double, prosumer-
   PDC string)
2:     prosumer ← GetPrivateData(prosumerPDC, prosumerId)
3:     res ← GetPrivateData(prosumerPDC, resId)
4:     if res.OwnerID == prosumerId then
5:         if res.IsCertified == True then
6:             energyProvider ← GetPrivateData(prosumerPDC, prosumer.EnergyProviderID)
7:             if prosumer.AggregatorID ≠ NULL then
8:                 aggregator ← GetPrivateData(prosumerPDC, prosumer.AggregatorID)
9:                 availableCapacity ← aggregator.MaximumCapacity − aggregator.StoredElectricity
10:                if amount > availableCapacity then
11:                    autoSellAmount ← amount − availableCapacity
12:                    finalSellPrice ← autoSellAmount ∗ energyProvider.EnergyPrice
13:                    AutoSellElectricity(prosumer.WalletId, finalSellPrice, prosumerPDC) ▷ c. f. , Al-
                       gorithm 3
14:                    if availableCapacity > 0 then
15:                        prosumer.Electricity ← prosumer.Electricity + availableCapacity
16:                        aggregator.StoredElectricity ← aggregator.StoredElectricity+availableCapacity
17:                        PutPrivateData(prosumerPDC, prosumerId, prosumer)
18:                        PutPrivateData(prosumerPDC, aggregatorId, aggregator)
19:                else
20:                    prosumer.Electricity ← prosumer.Electricity + amount
21:                    aggregator.StoredElectricity ← aggregator.StoredElectricity + amount
22:                    PutPrivateData(prosumerPDC, prosumerId, prosumer)
23:                    PutPrivateData(prosumerPDC, aggregatorId, aggregator)
24:            else
25:                finalSellPrice ← amount ∗ energyProvider.EnergyPrice
26:                AutoSellElectricity(prosumer.WalletId, finalSellPrice, prosumerPDC)
27:        else
28:            return "Prosumer RES is not certified"
29:    else
30:        return "Prosumer is not owner of the RES"
```

Algorithm 3 Automatic Electricity Selling to Energy Provider

Require: *Prosumer Wallet*

```
1: function AutoSellElectricity(prosumerWalletId string, currency double, prosumerPDC string)
2:     prosumerWallet ← GetPrivateData(prosumerPDC, prosumerWalletId) ▷ Retrieving the seller's
       wallet record.
3:     prosumerWallet.Currency ← prosumerWallet.Currency + currency
4:     PutPrivateData(prosumerPDC, prosumerWalletId, prosumerWallet) ▷ Saving seller wallet
       record in the energy provider and regulator PDC, c. f. , PDC2 or PDC3 in Figure 9.2.
```

asset remains on the immutable ledger, but the record itself cannot be used in transactions. The *DeleteState(key)* function is used for public data and the *DeletePrivateData(collection, key)* is used for private data.

By design, the input parameters of the transaction proposal are visible to peers who endorse the transaction. Thus, all input parameters of the transaction where private data is manipulated are stored in the *Transient* storage within our marketplace SC functions. Transient storage is not included in the transaction proposal and is used to hide the input parameters from peers who are not members of the PDC.

Algorithm 4 Issue GO

```
1: function IssueGO(prosumerId string, regulatorId string, resId string, resOwnerId string, isRESCertified
   boolean, amount double)
2:     if resOwnerId == prosumerId then
3:         if isRESCertified == True then
4:             go ← NewRecord()
5:             go.OwnerID ← prosumerId
6:             go.RegulatorID ← regulatorId
7:             go.RESID ← resId
8:             go.ElectricityAmount ← amount
9:             go.IsConsumed ← False
10:            PutState({prosumerId}_{regulatorId}_go[increment], go) ▷ Saving newly issued GO in the
               Public Data ledger, c. f. , Figure 9.2.
11:        else
12:            return "Prosumer RES is not certified"
13:    else
14:        return "Prosumer is not owner of the RES"
   Regulator Executes CompleteGO(prosumerId, amount, prosumerPDC)          ▷ c. f. , Algorithm 5.
```

Algorithm 5 Complete GO

Require: *Prosumer*

```
1: function CompleteGO(prosumerId string, amount double, prosumerPDC string)
2:     prosumer ← GetPrivateData(prosumerPDC, prosumerId)
3:     prosumer.Electricity ← prosumer.Electricity − amount
4:     PutPrivateData(prosumerPDC, prosumerId, prosumer) ▷ Saving prosumer record in the energy
       provider and regulator PDC, c. f. , PDC2 or PDC3 in Figure 9.2.
```

8.4.2.2 Marketplace Actors Registration

Within the marketplace, the regulator takes responsibility for the creation of a range of data records and their updating. The peers of regulator BO create the regulator, RES, and aggregator data records. The regulator data record is created with the governmentally regulated upper energy sell price limit saved in *PriceLimit* field. This field is saved in the public data ledger and can be updated only by the regulator BOs. Further, the regulator creates RES records within the marketplace. The *IsCertified* field signifies whether the RES is certified by the regulator. This field also serves as a regulatory mechanism to restrict prosumer actions within the marketplace in case of RES manipulation. In such a case, the regulator revokes the certificate and changes *IsCertified* value to *False* until the dispute is resolved. Thus, the prosumer is not further able to generate new GOs and automatically sell electricity to the energy provider. The RES record identifier is constructed as a concatenation of a *RegulatorID* field, a "res" string, and an integer, which is incremented each time a new record is created. Next, to address the **FR4**, the aggregator record is created for the certified equipment only, and the regulator guarantees the matching between the real-world equipment serviceability and correctness of blockchain records. If the aggregator is degraded and cannot be used anymore by the marketplace prosumers, its record can be marked as deleted, *i. e. ,* all prosumers who were connected to it must subscribe to another one. Both RES and aggregator records have their owner identifiers attached to them to conduct validation of ownership during SC functions execution. The aggregator record identifier is constructed

Algorithm 6 Create Sell Electricity Order

```
Require: GO, Regulator
function CreateSellElectricityOrder(sellerWalletId string, goId string, price double, amount double)
    go ← GetState(goId)                                          ▷ Retrieving the GO record.
    regulator ← GetState(go.RegulatorId)                         ▷ Retrieving the regulator record.
    if go.ElectricityAmount == amount then
        if price ≤ (amount * regulator.PriceLimit) then
            order ← NewRecord()
            order.Type ← "Sell"
            order.Price ← price
            order.ElectricityAmount ← amount
            order.GOID ← goId
            order.SellerWalletID ← sellerWalletId
            PutState({sellerWalletId}_order[increment], order) ▷ Saving new order in the Public Data
            ledger, c. f. , Figure 9.2.
        else
            return "Incorrect selling price"
    else
        return "Insufficient GO electricity amount"
```

Algorithm 7 Create Buy Electricity Order

```
function CreateBuyElectricityOrder(buyerWalletId string, price double, amount double)
    order ← NewRecord()
    order.Type ← "Buy"
    order.Price ← price
    order.ElectricityAmount ← amount
    order.BuyerWalletId ← buyerWalletId
    PutState({buyerWalletId}_order[increment], order) ▷ Saving new order in the Public Data ledger,
    c. f. , Figure 9.2.
```

as a concatenation of a *RegulatorID* field, an "aggregator" string, and an integer, which is incremented each time a new record is created. Both RES and aggregator records are saved in the energy provider and regulator PDC, *c. f. ,* PDC2 or PDC3 in Figure 9.2.

The energy provider creates its data record within the PDC, which it shares with the regulator. The energy provider record is created with the predefined market electricity price saved in the *EnergyPrice* field. This field is updated by the energy provider and monitored by the regulator to prevent electricity price deregulation.

8.4.2.3 Enroll New Prosumer

When the new prosumer is registered in the marketplace by the MSP of his/her BO, new blockchain data records must be created, *i. e. ,* prosumer and wallet. First, the MI generates a new prosumer record identifier as a unique value and stores it in the prosumer's X.509 public certificate. During every interaction with the MI, the identifier stored in the corresponding X.509 public certificate is used for SC transaction proposal submission. The process of new prosumer enrollment is shown in Algorithm 1. This process can be executed only by the energy provider and regulator BOs of the prosumer. During the prosumer enrollment process, the wallet record, an integral part of the prosumer, is also created. Its unique identifier is generated randomly since it is linked with the main prosumer record through the *WalletID*

Algorithm 8 Fulfill Sell Electricity Order

Require: *Order* ($Type = Buy$), *GO*, *Seller Wallet*, *Buy er Wallet*

```
1: function SellElectricity(orderId string, goId string, buyerRegulatorID string, sellerId string, sellerWalletID string, sellerPDC string, buyerPDC string, tradePDC string)
2:     order ← GetState(orderId)                                   ▷ Retrieving the buying order record.
3:     go ← GetState(goId)                                         ▷ Retrieving the record of GO proposed by seller.
4:     regulator ← GetState(buyerRegulatorID)                      ▷ Retrieving the buyer regulator record.
5:     sellerWallet ← GetPrivateData(sellerPDC, sellerWalletId)    ▷ Retrieving the seller wallet record.
6:     buyerWallet ← GetPrivateData(buyerPDC, order.BuyerWalletID) ▷ Retrieving the buyer wallet record.
7:     if go.OwnerID == sellerId & go.IsConsumed == False then
8:         if go.ElectricityAmount == order.ElectricityAmount then
9:             if buyerWallet.Currency ≥ order.Price then
10:                order.SellerWalletID ← sellerWalletID
11:                order.GOID ← goId
12:                sellerWallet.Currency ← sellerWallet.Currency + order.Price
13:                buyerWallet.Currency ← buyerWallet.Currency − order.Price
14:                buyerWallet.Electricity ← buyerWallet.Electricity + order.ElectricityAmount
15:                PutPrivateData(tradePDC, orderId, order)    ▷ Saving order in PDC1, c. f. , Figure 9.2.
16:                PutPrivateData(sellerPDC, sellerWalletId, sellerWallet)
17:                PutPrivateData(buyerPDC, order.BuyerWalletID, buyerWallet) ▷ Save wallets in the energy provider and regulator PDCs, c. f. , PDC2 and PDC3 in Figure 9.2.
18:            else
19:                return "Insufficient buyer currency"
20:        else
21:            return "Insufficient electricity amount"
22:    else
23:        return "Invalid GO"
    Regulator executes FinalizeOrder(orderId, goId)                ▷ c. f. , Algorithm 10.
```

field. The enrollment process is completed by saving the prosumer and wallet in the energy provider and regulator PDC, *c. f. ,* PDC2 or PDC3 in Figure 9.2. Finally, the energy provider assigns a newly created prosumer ID to the RES record, which was certified and created by the regulator.

8.4.2.4 Register Prosumer Electricity

To address the **FR1**, the electricity generated by the RES is registered in the prosumer data record, *i. e. , Electricity* field. It is the responsibility of the regulator to certify the RES and enable generated electricity registration. Further, the generated electricity has to be stored in the aggregator in case it has available capacity. The process of registering prosumer-generated electricity is shown in Algorithm 2. This process can be executed only by the energy provider or the regulator BOs of the prosumer. The generator RES has to be certified and owned by the prosumer. Further, the prosumer has to have an aggregator assigned to it. If all conditions are met, the aggregator's available capacity is checked. If it can store the entire amount of generated electricity, both prosumer's *Electricity* and aggregator's *StoredElectricity* are updated and saved in the energy provider and regulator PDC, *c. f. ,* PDC2 or PDC3 in Figure 9.2. If only a portion of electricity can be stored, the rest is automatically sold to the energy provider at a predefined market price, *i. e. , EnergyPrice* field in Table 8.8. This addresses the **FR2**. The process of automatic energy

Algorithm 9 Fulfill Buy Electricity Order

Require: *Order* (*Type* = *Sell*), *GO*, *Buyer Wallet*, *Seller Wallet*

function BUYELECTRICITY(orderId string, buyerWalletId string, buyerPDC string, sellerPDC string, tradePDC string)
  $order \leftarrow GetState(orderId)$ ▷ *Retrieving the selling order record.*
  $go \leftarrow GetState(order.GOID)$ ▷ *Retrieving the record of GO attached to the selling order.*
  $buyerWallet \leftarrow GetPrivateData(buyerPDC, buyerWalletId)$ ▷ *Retrieving the buyer wallet record.*
  $sellerWallet \leftarrow GetPrivateData(sellerPDC, order.SellerWalletID)$ ▷ *Retrieving the seller wallet record.*
  if $buyerWallet.Currency \geq order.Price$ **then**
    if $go.IsConsumed == False$ **then**
      $order.BuyerWalletID \leftarrow buyerWalletId$
      $sellerWallet.Currency \leftarrow sellerWallet.Currency + order.Price$
      $buyerWallet.Currency \leftarrow buyerWallet.Currency - order.Price$
      $buyerWallet.Electricity \leftarrow buyerWallet.Electricity + order.ElectricityAmount$
      $PutPrivateData(tradePDC, orderId, order)$ ▷ *Saving order in PDC1,* c. f. , *Figure 9.2.*
      $PutPrivateData(buyerPDC, buyerWalletId, buyerWallet)$
      $PutPrivateData(sellerPDC, order.SellerWalletID, sellerWallet)$ ▷ *Saving wallets in the energy provider and regulator PDCs,* c. f. , *PDC2 and PDC3 in Figure 9.2.*
    else
      return "*Invalid GO attached to order*"
  else
    return "*Insufficient buyer currency*"
  Regulator Executes $FinalizeOrder(orderId, goId)$ ▷ c. f. , *Algorithm 10.*

Algorithm 10 Finalize Order

Require: *GO*, *Order*

function FINALIZEORDER(orderId string, goId string)
  $order \leftarrow GetState(orderId)$ ▷ *Retrieving the order record to delete.*
  $go \leftarrow GetState(goId)$ ▷ *Retrieving the GO record to consume.*
  $go.IsConsumed \leftarrow True$ ▷ *Mark GO as consumed.*
  $PutState(goId, go)$ ▷ *Saving consumed GO in the Public Data ledger,* c. f. , *Figure 9.2.*
  $DeleteState(orderId)$ ▷ *Deleting fulfilled order.*

selling is shown in Algorithm 3.

8.4.2.5 Issue GO

To address the **FR3** and **NR3** the *IssueGO* function is defined, *c. f. ,* Algorithm 4. The GO is issued solely by the regulator BO within the marketplace. It plays a central role in the electricity trade process, *i. e. ,* without valid GO, prosumer cannot sell electricity. Thus, RES ownership by the prosumer and valid certification is checked during GO generation. The GO generation process consists of two stages to separate public and private data processing. In the first stage, the GO record is created on a given amount of electricity and saved in the public data ledger. This process is executed only by the regulator BO. It contains the identifiers of the regulator that issued the GO, the RES that generated the electricity, and the prosumer-owner. The GO identifier is constructed as a concatenation of a prosumerId and regulatorId fields, "go" string, and an integer, which is incremented each time a new record is created. In the second stage, the GO amount of electricity is subtracted from the prosumer account and saved in the respective PDC, *c. f. ,* Algorithm 5. This process can be

executed only by the regulator.

8.4.2.6 Electricity Trade Order Creation

To address the **FR2**, the electricity order creation functions are defined. The electricity order enables the P2P trade settlement within the marketplace. When posted, all orders are saved in the public data ledger. Depending on the type of the order, *i. e. ,* buy or sell, certain security checks are executed. For *sell* order, the valid GO has to be presented. The prosumer has to be the owner of GO, and the electricity amount of the order should match the one in GO. Further, the price of this order should not exceed the one established by the prosumer's regulator, *c. f. ,* Algorithm 6. For *buy* order, the buyer currency value has to be greater or equal to the order price *c. f. ,* Algorithm 7. Finally, in both order types, the identifier is generated as a concatenation of issuer prosumer wallet id, "order" string, and an integer, which is incremented each time a new record is created.

8.4.2.7 Electricity Trade

When an electricity sell or buy order is saved on the ledger, prosumers may fulfill this order by utilizing *SellElectricity* or *BuyElectricity* SC functions, *c. f. ,* Algorithms 8 and 9. The trade settlement operation execution has two stages. In the first stage, either sell or buy settlements are executed. Both types of trade settlements work only with prosumer wallets, and during the transaction, the participant BOs peers can see both prosumer wallets. The wallet is a necessary partial private data disclosure to assure the selling side that buyer has enough currency to purchase the electricity. The *SellElectricity* function takes a buy order that was posted by a prosumer-buyer, *c. f. ,* Algorithm 8. Since the seller brings a new GO into the order, its validity is checked, *i. e. ,* ownership, consumption, and electricity amount fields. Further, the buyer's currency is checked to be greater or equal to the order price. The *BuyElectricity* function takes a sell order with a GO attached to it, *c. f. ,* Algorithm 9. Thus, the algorithm checks if the GO is consumed and if the buyer has enough currency in the wallet. Further, in both cases, the buyer currency is decreased by the order price and increased by the electricity amount. The buyer can only consume the bought electricity, *i. e. ,* cannot resell it. The seller currency is increased, and both wallets are saved in the respective PDC, *c. f. ,* PDC2 or PDC3 in Figure 9.2. Finally, the order record is saved in the energy providers and regulator PDC, *c. f. ,* PDC1 in Figure 9.2. In the second stage the *FinalizeOrder* function is executed, *c. f. ,* Algorithm 10. First, this function takes the GO, sets its *IsConsumed* value to True, and saves it in the public ledger. Further, it marks the fulfilled order as deleted, *i. e. ,* it does not appear in the pool of orders for prosumers. This addresses the **FR1** and **NR2**.

8.4.3 Marketplace Interface Implementation

The MI is implemented with Node.js version 10.13. It contains the necessary functionality required to communicate with the SC functions described in Section 8.4.2. As the focus of this work is the investigation of the marketplace SC performance, the MI implementation is kept simple, assuming that this component is trusted and uncompromised, *i. e. ,* contains

no critical software defects [39] compromising the entire marketplace. To address **SR6**, the communication with MI is encrypted using the public key infrastructure supported by the HF's CA. Within the MI discussed in this study, the following operations have been identified:

8.4.3.1 Registration

The prosumer uses MI to request registration in the marketplace. First, the prosumer sends a request to MI where his/her RES identifier may be included if present, *i. e.,* RES can be registered and assigned to the prosumer after registration. MI uses electronic identification (eID), *e. g.,* eIDAS, which is tied to the prosumer's physical identity to confirm activation of enrolment action. Upon confirmation, a nonce (number to be used just once) is returned over the eID application. The received nonce is used in the CA application on the prosumer's device. First, the CA application generates a public and private key pair. Further, the CA application encloses the public key in the CSR, which is sent to MI, along with the nonce. This way, MI can be confident that the signature request comes from a legitimate prosumer. MI signs the CSR and returns the digital certificate to the client CA application. Next, MI generates a new prosumer record identifier as a unique value and stores it in the prosumer's public certificate. Finally, the CA signs the certificate, and it is sent back to the prosumer. Further, the MI triggers the enroll user algorithm on the peer of the energy provider BO, *c.f.,* Algorithm 1.

8.4.3.2 Personal Data Read

To check the generated electricity balance, the prosumer first needs to authenticate towards the MI. This is done using the X.509 certificate that was generated during the prosumer registration. In order to authenticate, prosumers must provide proof of private key ownership. The MI sends a cryptographic challenge, *e. g.,* a blob of data, which the prosumer has to encrypt with the private key. The MI decrypts the received back cyphertext with the public key stored in the X.509 certificate. The prosumer is authenticated if decrypted data and challenges match. If true, the prosumer data record identifier is read from the X.509 certificate and relayed to a peer to execute a read query. The prosumer can read its data, wallet values, owned res, aggregators, personally created orders, and owned GOs.

8.4.3.3 Electricity Trade

Prior to trade settlement execution, the issuer has to submit the order to the ledger where the amount of electricity to buy or sell is specified. The prosumer that fulfills the order authenticates towards the MI and triggers the trade settlement SC function on the energy provider BO peer. After all the necessary verifications, the buyer transfers the agreed amount of currency to the seller's account and gains the electricity. Finally, the transaction is submitted to the orderer, where it is put into a block and then submitted to all peers to be saved in the ledger.

8.4.3.4 Metering Device Data Registration

In a real-world scenario, the digital layer of the marketplace is connected to an energy grid and receives the data from metering devices that measure generated electricity in households. Such metering devices register the generated electricity and send it to the energy provider periodically. According to the energy provider involved in our study, this happens every 10 minutes, but the system is prepared for second or subsecond frequency. Such metering device updates have to be stored securely in order to be used in electricity trade transactions within the blockchain-based marketplace. In our implementation, we emulate the energy grid with the *orchestrator node*, *c. f. ,* Figure 9.4. An orchestrator emulates prosumers' metering devices by registering a certain amount of kWh to user accounts on a ledger per a certain period of time (depending on the configuration).

In addition, the orchestrator is used as a measurement tool. It calculates the number of trade settlements executed in the marketplace per second, as well as the number of concurrent market users. Such performance evaluation is discussed in the next section.

8.5 Performance Evaluation

In order to investigate the performance of blockchain-based infrastructure, a number of quantitative characteristics of an implemented marketplace were evaluated, *e. g. ,* transaction throughput and latency. The performance evaluation was conducted on a distributed computing infrastructure instantiated in the Microsoft Azure[17]. The performance evaluation was run on the marketplace SC and concentrated on electricity trade settlement transactions as they require the most computations and are the largest in terms of disk space. The marketplace test implementation structure is depicted in Figure 8.6. The test implementation is deployed on four virtual machines (VMs) where each VM size is 16 vCPUs, 64 GB RAM, and 256 GB high throughput (150MB/s) disk space. Energy providers A and B run VM1 and VM2 respectively. The regulator runs VM3 and VM4, *i. e. ,* acts as a BO and orderer nodes operator. All VMs are connected with a 10Gbit/s network interface. In our experimental implementation, we use Hyperledger Fabric (HF) version 2.1 without any modifications to the core code. As a state DB, we use LevelDB[18] due to its performance advantage [40]. All nodes within the infrastructure are deployed as docker containers.

8.5.1 Data Gathering Techniques

In order to collect reliable and correct performance evaluation data, specialized and certified data gathering techniques have to be applied. In this evaluation, the process of data gathering is executed with two software tools, *c. f. , Metrics Observer* in Figure 8.6.

First, the *Prometheus*[19] real-time metrics database is used as the main blockchain operation data collector. It is an open-source system monitoring and alerting toolkit originally built at SoundCloud. It is connected directly to the HF's peer and orderer nodes

[17] https://azure.microsoft.com
[18] https://dbdb.io/db/leveldb
[19] https://prometheus.io/docs/introduction/overview/

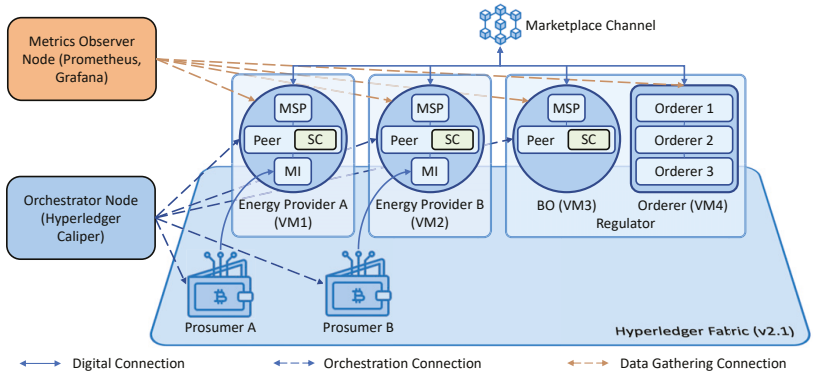


Figure 8.6: *Implemented P2P Blockchain-based Energy Marketplace.*

and collects real-time data on numerous HF characteristics, *e. g.*, number of generated blocks or number of successful / failed transactions per unit of time.

Second, the *Grafana*[20] software is used as a data presentation tool. It enables the composition of rich dashboards, where data is presented in a comprehensive manner and allows exporting of the processed data into an in-depth report. The Prometheus is plugged into Grafana as a data source and feeds the data to a dashboard.

Together, Prometheus and Grafana give a comprehensive overview of the blockchain network operation data. They also enable a comparison of the historical data that has been collected over time from different HF configurations and marketplace endorsement policies.

8.5.2 Transaction Load Generator

In order to test the throughput limits of the implemented marketplace, the constant number of transactions sent to the blockchain has to be maintained. In this case, multiple requests have to be sent to the marketplace simultaneously. Thus, the Hyperledger Caliper[21] (HC) performance evaluation tool was used, for transaction load generation. HC was developed by the Hyperledger Foundation to measure the performance of a blockchain implementation with a set of predefined use cases. It allows to highly parallelize transactions sent to the marketplace, thus, providing a consistent load. HC also allows observing such performance metrics as transaction throughput, latency, and resource consumption (CPU, Memory, Network). However, HC is limited in comparison to Prometheus and Grafana in terms of specific variables that detail the throughput of all transaction life-cycle phases, *i. e.*, endorse-order-validate, and ledger blocks-related information. In Figure 8.6, the HC is part of the Orchestrator Node which is a separate VM. HC bypasses the MI, hence the MI was not part of the load testing.

[20] https://grafana.com/
[21] https://hyperledger.github.io/caliper/

8.5.3 Configured Policy

In HF, an EPCF specifies a set of peer nodes that must execute and endorse each transaction in the network in order for it to be considered valid. In our setup, the EPCF requires one peer per energy provider and regulator to endorse and validate the trade settlement. In this way, the system ensures that both energy providers and the regulator replicate the transaction result, verify the produced data, and save it in the respective PDC, *c. f. ,* PDC1 in Figure 9.2. This is done due to the specific assumptions about the energy marketplace, *i. e. ,* electricity trade transactions must be regulated by governmental bodies. Thus, strong guarantees are needed that the transaction data is valid and have not been tampered with. In addition, the configured policy specifies three orderers, which execute the RAFT consensus protocol. Three orderers are the minimum needed amount to execute RAFT, per the HF developers' recommendation. Orderer nodes operate in crash fault-tolerant mode, *i. e. ,* even if one orderer node fails, the blockchain network can operate normally.

8.5.4 Throughput

The throughput, *i. e. ,* transactions per second (TPS), is an essential characteristic of the blockchain-based marketplace, as far as it affects the quality of service (QoS) [41] for the prosumers. QoS is aimed to maximize the user experience in terms of response time and transaction success rate by addressing throughput and scalability issues. In addition, it affects the granularity of the updates from the metering devices that can be saved on the blockchain within a certain period of time.

Throughput is calculated as a number of *successful* transactions that have been processed by the blockchain network during one second. Such successful transactions may end up in different blocks. During an evaluation, a constant number of TPS is sent to the marketplace per one 5-minute test execution. Then, the average value of successful TPS is calculated and taken as a final throughput value. During an evaluation, next fixed transaction send rates were chosen: *10, 25, 50, 100, 150, 200, 300, 400, 500, 600.* The transaction send rate step was gradually increased to land on a maximum possible throughput for an HF-based marketplace.

A number of configurable metrics were manipulated within HF to investigate the throughput. These metrics were selected based on the performance tests conducted by the HF developers and research studies [42, 43]. The first configurable metric, which was manipulated to investigate the changes in TPS, is the HF *block configuration.* Such configuration consists of a number of characteristics that affect the size and timeout of orderer-produced blocks. First, it is a *BatchTimeout*, which specifies the timeout to receive incoming transactions that go into one block. Further, it is a *batch size* which is defined by three components: *MaxMessageCount, AbsoluteMaxBytes, PreferredMaxBytes.* The MaxMessageCount specifies the maximum number of transactions that can go into one block. After the transaction limit is reached, even if BatchTimeout has not elapsed, the block is finalized and sent to committing peer. Further, the AbsoluteMaxBytes specifies the maximum block size in bytes. Finally, the PreferredMaxBytes specifies the preferred block size in bytes. If a blockchain network processes transactions that weigh less than PreferredMaxBytes, then the block must never exceed the preferred size. If one large transaction was generated that outweighs the

PreferredMaxBytes, the block must never weigh more than AbsoluteMaxBytes. In case the transaction size is larger than AbsoluteMaxBytes, it will be discarded.

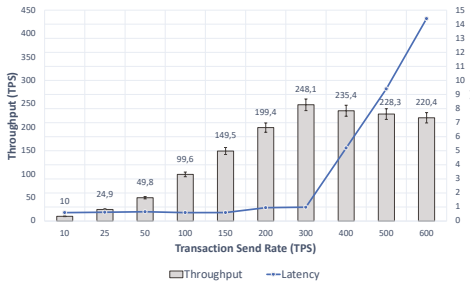
During performance evaluation, a *1 second* BatchTimeout was chosen. It is an acceptable waiting time, even if not a large number of transactions are being processed by the marketplace, *i. e.,* block does not reach MaxMessageCount before BatchTimeout elapses. Such timeout also should maintain an optimal QoS, such as the prosumer has to wait not more than 1 second for the trading transaction to be recorded on the ledger (if no other delays are present). For the PreferredMaxBytes, a *3MB* size was chosen. This is done because the trade transaction approximate size is *4KB*. In this way, we can assume that a 3MB block should host up to approximately 750 transactions, which is an amount that exceeds measured maximum trade settlement throughput. The AbsoluteMaxBytes was set to 99MB to make sure that with such a 4KB transaction size, the maximum block size is never reached, *i. e.,* no limit in terms of max size is set. In contrast to the aforementioned block characteristics, which remain static, the MaxMessageCount, *i. e.,* block size, was varied throughout experiment execution in order to investigate its effect on the TPS. During performance evaluation a large variety of block sizes were investigated. In this study, we demonstrate the results for he next block sizes: *50, 100, 200, 300, 500*. The reasoning behind these specific block sizes is to demonstrate the TPS increase as well as the maximum possible throughput. Entire performance evaluation parameters configuration is summarized in Table 8.12.

Table 8.12: *Performance Evaluation Parameters Configuration*

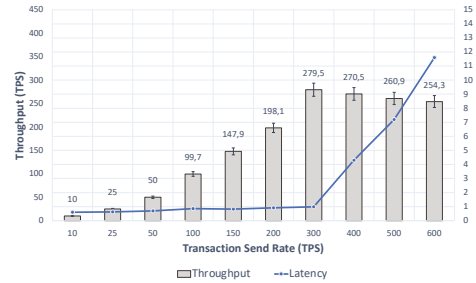
Parameter	Value
BatchTimeout	1 second
AbsoluteMaxBytes	99 MB
PreferredMaxBytes	3MB
Transaction Send Rate (Write)	10, 25, 50, 100, 150, 200, 300, 400, 500, 600 *(fixed-rate in duration of 5 minutes)
Block Size (MaxMessageCount)	50, 100, 200, 300, 500
Transaction Send Rate (Read)	10, 50, 100, 200, 300, 400, 500, 650, 800, 1000, 1300, 1600, 1900, 2200 *(fixed-rate in duration of 5 minutes)
Asset Size	512B, 1KB, 2KB, 4KB, 8KB
State Database	LevelDB
Storage Type	Disk, RAM
Endorsement Nodes	1 per Energy Provider
Smart contract language	Go

8.5.4.1 Write - Trade Settlement Execution

In this study, the Algorithm 8 is executed as an SC function to test maximum write TPS. The Algorithm 8 was chosen as a load generator due to having the highest computational complexity out of all defined SC functions. The rest of the defined algorithms are not a part of performance evaluation due to the lower amount of computations, *i. e.,* transaction time, needed in comparison to the investigated sell electricity function, *i. e.,* Algorithm 8. In order to write to the blockchain ledger, both endorsing policy, *i. e.,* specified in EPCF, and consensus protocol, *i. e.,* RAFT, must be executed. The performance evaluation results of write ledger transactions are described in Figure 8.7. As can be seen from Figure 8.7e,

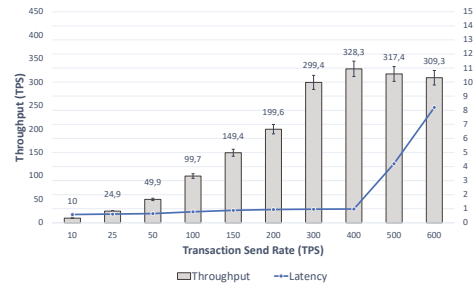


(a) Block Size = 50

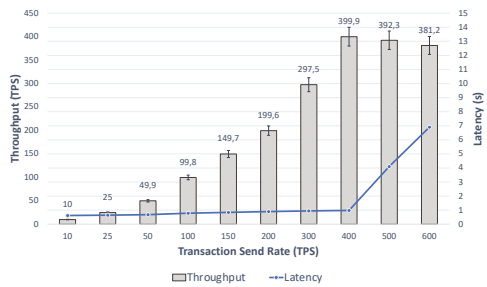


(b) Block Size = 100

(c) Block Size = 200



(d) Block Size = 300



(e) Block Size = 300

Figure 8.7: *Trade settlement transaction throughput.*

the maximum number of transactions is achieved with the block size of 500 (390 - 400 TPS), which is approximately five times more than in blockchain-based applications built on Ethereum private ledger [44]. For each block size, after the throughput limit is reached, the increase in the transaction send rate causes a throughput decrease. The throughput decrease continues with the increase of the transaction send rate. Such an effect is caused by the overflow of the transaction queue and a considerable increase in transaction latency, *c. f.*, Section 8.5.5. Furthermore, the queue overflow results in a failure of a number of transactions due to queue build-up and eventual transaction timeout, which affects the throughput result (only successful transactions are counted). The obtained results show that the block size has a significant impact on throughput and has to be tailored to the needs of an application scenario, *i. e.*, depending on transaction size, blockchain network load, and QoS.

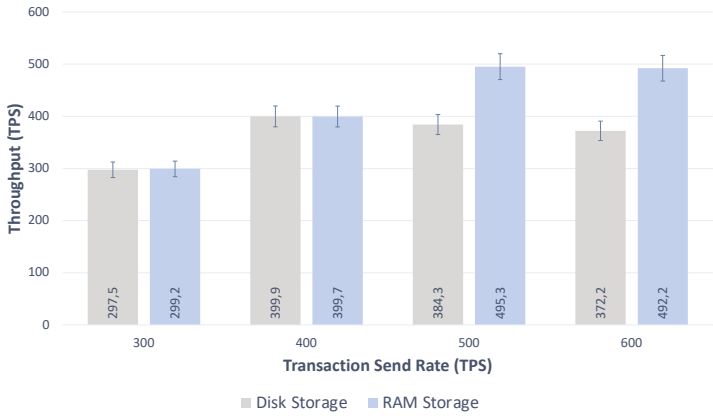


Figure 8.8: *Trade settlement transaction throughput for Disk and RAM storage.* (Block Size = 500)

The tests which are described next were executed with the *block size of 500 transactions* since it shows the best performance for energy marketplace use-case, *i. e. ,* for the developed smart contract operations. In order to identify the ledger write throughput bottleneck, the implemented solution has been tested with a RAM-based disk. In this approach, the HF docker containers were stored in the RAM disk only. Hence, the latency component due to disk I/O is avoided resulting in a higher number of input operations per second (IOPS). The usage of RAM disk increased the maximum write throughput by 25% (from 400 to 500) for the block size of 500 transactions, *c. f. ,* Figure 8.8. Such behavior leads to the conclusion that the disk IOPS limits the performance of the HF-based marketplace, representing a considerable performance bottleneck. However, such limitation is dependent on the hardware setup, which can be regulated by the marketplace channel's actors.

Further, in order to identify a bottleneck, the effect of orderer nodes, *i. e. ,* consensus execution nodes, has been investigated. In the main experimental setup, there are three orderer nodes, which are located on a separate VM and execute consensus. In order to investigate the consensus execution impact, a single orderer node was used. In this case, the consensus protocol is not executed as there is only one orderer node that does not have to agree on a transaction order with others. The decrease to a single orderer node does not demonstrate any write throughput increase, indicating that the RAFT consensus execution is not the performance bottleneck. Furthermore, the increase to 5 orderer nodes demonstrates HF's ability to scale and does not affect write throughput significantly, *i. e. ,* remains at approximately 400 TPS. However, when the CFT algorithm in RAFT is replaced by BFT, the consensus may represent a critical performance bottleneck due to increased complexity and computations [45].

Finally, to investigate the impact of different transaction life-cycle phases, *i. e. ,* endorse-order-validate, each phase throughput was measured individually, *c. f. ,* Figure 8.9. The results indicate that the *validate* phase represents the main and major performance bottleneck of the entire transaction life-cycle. This indicates that the queue buildup is concentrated in the validation phase, which sets the limit for maximum transaction throughput. The endorsement and ordering phases do not show any limitations as their throughput increases and matches the transaction send rate.

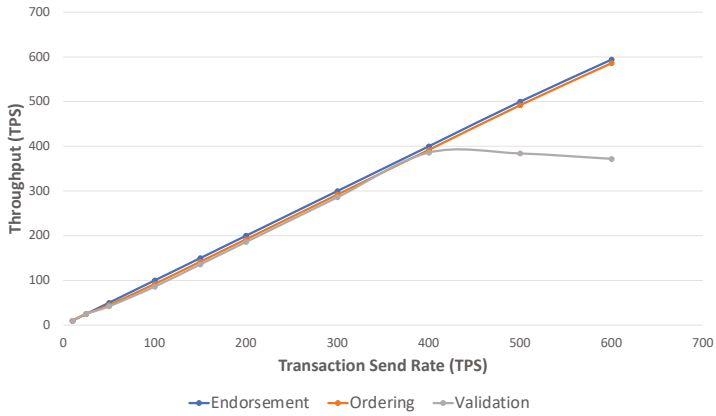


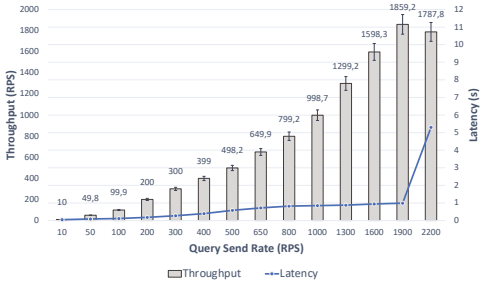
Figure 8.9: *Transaction life-cycle phase throughput.* (Block Size = 500. Validation phase is a bottleneck)

8.5.4.2 Ledger Data Read

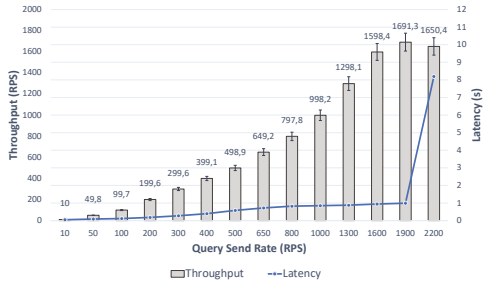
The performance evaluation results of read transactions per second (RPS) are shown in Figure 8.10. In order to read from the ledger, no policy or consensus execution is required. Thus, a read does not require the same amount of computations as for write transactions. Hence, it is not the block configuration but the *asset size* that affects the RPS. Asset size is the amount of information in *bytes* that is read from the ledger according to a specific query request. Depending on the query, different assets can be extracted from the ledger. Furthermore, an asset can be extracted from one or from different transactions. Since query size can vary, this study evaluates read throughput for default asset sizes to give the reader an idea on the marketplace's RPS. The following asset sizes were tested: *512B, 1KB, 2KB, 4KB, 8KB*. The testing query was built such as to read the data from a block where the data record was modified the last. As can be seen from Figure 8.10e, the maximum investigated asset size of 8KB can be read from the ledger with approximately 1112 RPS. Further, smaller transactions that result in the 512B asset size can be read from the ledger with approximately 1859 RPS, *c. f. ,* Figure 8.10a. After the maximum asset read throughput is reached, a further query send rate increase results in a throughput drop. This happens due to read queue buildup and subsequent failure of a number of queries. The obtained data demonstrate that the ledger read is dependent predominantly on the asset size, network latency, and speed of the storage solution where HF is deployed. Block size does not affect ledger read throughput since the consensus and the endorsement policy are not executed.

8.5.5 Transaction Latency

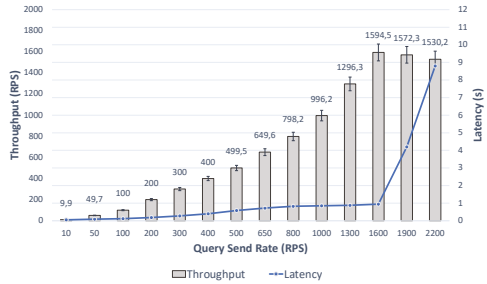
Transaction latency (TL) is an important characteristic that affects the QoS and the user experience. TL denotes the time (in seconds) a transaction needs to go through the entire endorse-order-validate life-cycle and be recorded in the ledger. In our performance evaluation, the TL is calculated for each aforementioned block size and transaction send rate. In addition, for each experiment, maximum, minimum, and average latencies were calculated to provide a full picture of the delays that may occur in the production environment.



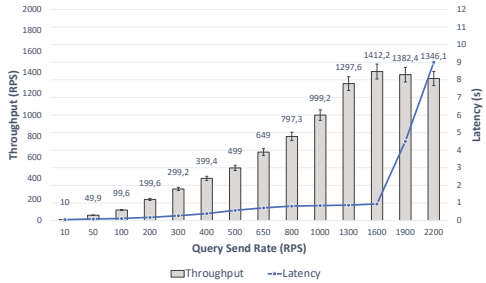
(a) Query Size = 512Bytes



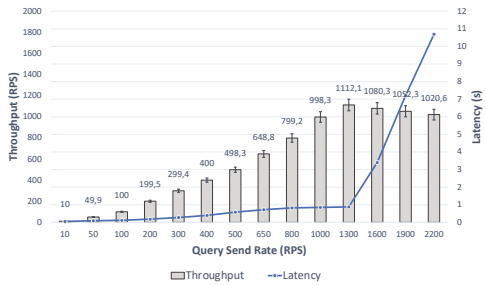
(b) Query Size = 1KByte



(c) Query Size = 2KBytes



(d) Query Size = 4KBytes



(e) Query Size = 8KBytes

Figure 8.10: *Read transaction throughput.*

8.5.5.1 Average Latency

The average transaction latency for write and read throughput is shown in Figures 8.7 and 8.7e, 8.10, and 8.10e. It represents the sum of all individual transaction latencies divided by the number of successful transactions submitted to the blockchain network. No latency restrictions were specified in the HF configuration in order to observe the effect of queue buildup. As can be seen from the results, the average latency stays under 1 second before we approach the maximum throughput for the respective block size. When the maximum throughput is reached, the average latency rapidly increases. This happens due to queue buildup, and thus, the more transactions are in the queue, the more time it takes to get ordered, validated, and written to the ledger. The TL rapid increase after maximum throughput is reached is a major QoS concern that may result in marketplace availability

degradation [46].

8.5.5.2 Maximum Latency

The maximum latency behaves in a similar manner to the average latency, *i. e. ,* increases rapidly after a peak throughput for the respective block size is reached. However, the difference between the average and maximum latency becomes smaller if send rate continues to be increased after the peer throughput is reached. For example, the average and maximum latencies for block size 500 tx and send rate 300 tx are 2 and 10 seconds respectively. Here, the difference is 400% or 8s. Further, for send rate 400 tx, the average and maximum latencies are 12 and 22 seconds respectively. Here, the difference is less than 100% or 10s. Such observation indicates that the higher the deviation from the maximum throughput, the higher the number of queued transactions, thus, consequently, the higher is the number of failed transactions due to timeout. The transaction remains in the queue until the 5-minute test is finalized and then discarded. The time interval from the moment transaction is placed in the queue and eventually failed at the end of the test is set as a maximum latency. In an industrial system, where execution is not limited by a 5-minute time span, all transactions eventually are recorded on the ledger, after being processed by the blockchain network. However, depending on the configuration, *e. g. ,* if no retry mechanism is implemented, a portion of transactions may fail due to timeout.

8.5.5.3 Minimum Latency

The minimum latency evaluation results indicate that the minimum latency is not affected by the send rate since the first received transactions are processed by the HF network instantaneously and do not requires additional time to set up a queue. Thus, the minimum latency does not exceed 0.21s.

8.6 Results and Observations

The implementation and execution of the decentralized blockchain-based energy marketplace demonstrate that defined requirements, *c. f. ,* Sections 8.2 and 8.3.3, can be fulfilled by HF. It permits the trade settlement process's automation by utilizing distributed ledger and SC. In addition, the blockchain consensus mechanism makes the energy trading process transparent, trusted, and secure. Throughout the process of the requirements definition, architecture design, and system implementation, a number of observations were made. Such observations are discussed next.

8.6.1 The Regulator is the De Facto TTP

The energy market is a field that is highly regulated by the government. Thus, the investigated use-case of P2P energy trading has to take into account the presence of the governmental entity and its regulatory control over the marketplace. Here, the regulator oversees all processes and acts as the main trust anchor in marketplace operations. The issuing and consumption of GO is controlled by the regulator. In addition, it regulates the process of

RES certification and generated electricity registration within the marketplace. Further, the regulator takes part in the electricity trade settlement as a guarantor that the GO is not sold more than once. Thus, the regulator can be viewed as a TTP which is the middleman of the trade settlement operations. However, considering the number of limitations that are put on the energy trading process, the marketplace enables automation of trade settlement process and connects the end-users of the energy grid, *i. e.,* prosumers, in a trusted and transparent way.

The GO management may be done outside the ledger, considering a certain level of centralization around the regulator. However, synchronizing the ledger with the outside GO management system becomes challenging and opens for new attack vectors, *e. g.,* spoofing and tampering attacks on the communication channel. It also puts an additional operational burden on the regulator as the outside GO management system would have to be maintained and secured. Hence, the challenge is to find the balance between system complexity, resilience, and operational overhead. In the marketplace described in this study, all assets remain on the ledger to secure the data integrity and reduce the attack surface. In addition, the blockchain ledger serves as an irrefutable source of truth and increases the credibility of auditing marketplace assets and processes.

8.6.2 Robust Interface Between the Blockchain-Based Marketplace and the Electricity Grid

The energy grids, *i. e.,* electricity distribution systems, nowadays represent a critical infrastructure that citizens rely on in their everyday life. In the case of the energy grid failing, the societal and environmental impact may be catastrophic. Thus, the blockchain-based marketplace, when applied in the context of electricity distribution systems, requires the implementation of a robust and safe interface between these two entities. To this end, the energy grid operator has to make sure that the marketplace gets accurate and timely updates on the generated electricity. However, the marketplace has to be developed in a way that it is unable to tamper with the energy grid's operations. Thus, the blockchain-based marketplace should be built as a standalone entity that just operates on the data, which is transmitted securely from the energy grid digital infrastructure.

8.6.3 Limitation of the Trade Settlement Operations Per Second

One of the blockchain's quantitative characteristics is throughput, *i. e.,* the number of transactions that can be processed and stored on a ledger per second. This characteristic has to be considered at a system design phase since the initially chosen blockchain platform may, in the end, have unsuitable throughput that may not fulfill the scalability requirements of the release system. Consequently, throughput may be affected by the blockchain architecture, *i. e.,* private or public, computational requirements, *e. g.,* validation in HF, and consensus mechanisms, *e. g.,* Proof of Work (PoW) [21] or RAFT. The performance evaluation of implemented blockchain-based marketplace has shown a throughput of approximately 400 TPS, which is a considerable improvement compared with other blockchain solutions such as layer 1, *i. e.,* base networks, for Ethereum [47] or Bitcoin [21]. However, HF still requires efficient mechanisms that increase the maximum throughput and scalability.

The maximum measured throughput of 400 TPS allows for calculating an estimated number of the prosumers that the designed energy marketplace may support. The authors acknowledge that such calculation is simplified, and it is done to give an approximation regarding the maximum supported prosumer number. In the real-world system, the maximum prosumer number may vary due to other factors not taken into consideration here. For our calculation, the following metrics were outlined. The T_{max} parameter is the system's maximum throughput, *i. e.,* 400 TPS in our case. The m parameter is the number of daily (24 hours) metering device updates. The o parameter is the number of orders created during a day by a prosumer. The t parameter is the number of trade transactions executed during a day. The g parameter is the number of GOs issued for a prosumer during a day. Finally, dividing the maximum daily amount of transactions by the sum of *m, o, t, g* yields an approximate number of prosumers that are able to operate within a marketplace, *c. f.,* (8.1).

$$Prosumers_{max} = \frac{T_{max} * 24 * 60 * 60}{m + o + t + g} \tag{8.1}$$

The throughput of 400 TPS allows the execution of 34.560.000 transactions per 24 hours. Thus, the *m, o, t, g* configuration solely defines the final result of the maximum supported prosumer number. In order to trade, the energy must first be generated and registered in the marketplace. Thus, the m parameter has to be set first and affects the rest of the parameters. If the m parameter is set to 24, *i. e.,* hourly metering device update, it gives the prosumer the possibility to trade generated electricity 23 times during the day (minus 1 hour for initial generation). Hence, the worst-case scenario is that the prosumer trades electricity every time the metering device update happens. For each trade to be executed, the order must be placed and GO issued. In such a case, the final number of daily prosumer transactions equals 93 (24+23+23+23). Thus, 34.560.000 / 93 $\approx$ 370.000 prosumers. Such a number of prosumers corresponds to a medium size energy community.

8.6.4 Marketplace Concurrent Operations Impact Overall System Performance

In Hyperledger Fabric, each record on the ledger is stored in the same manner as an ordinary record stored in a conventional database, *e. g.,* MySQL. Thus, when two processes at once try to modify the same record on the ledger, one of these modifications fails. However, the inherent characteristic of Hyperledger Fabric is that although the failed transaction does not modify the ledger, it is still stored in an ordered block. To this end, an application requires functionality that identifies failed attempts and triggers a retry mechanism. At this point, HF does not have such functionality, which requires its implementation within the application layer of a blockchain-based solution. Such functionality is also interesting to investigate as it may lead to a new kind of non-blockchain-related delays.

8.6.5 Constant Transaction Load Leads to a Significant Blockchain Growth

As was discussed in Section 9.3, the metering device updates are registered in the blockchain-based marketplaces at a certain frequency. Measured maximum throughput of 400 TPS with a constant full load can support 1,440,000 transactions per hour. With the approximate

transaction size of 4KB, the maximum transaction load can grow the blockchain by 5.76GB of data per hour, by 138.24GB of data every 24 hours, or by 4.14TB per month. This approach is not sustainable, as over time this may introduce a major economic issue to store the entire blockchain, *i. e.,* increased operational costs. Thus, one optimization solution is to offload the metering device updates from the blockchain and process and store them in a conventional database. However, this raises issues of trust and validity of metering device data. Here, a separate type of consensus must be executed between the energy providers, who must agree on the metering device data before executing the trading transaction. Moreover, the metering device data can be stored in integrity-guaranteed data storage, which utilizes provable data possession technique [48]. Further, the aggregated metering device updates can be recorded in the marketplace with a smaller granularity, *e. g.,* once per hour or day. In this way, the ledger is not overloaded with frequent metering device updates, and the marketplace trade transactions operate on valid electricity generation data.

Another solution to the blockchain growth issue is *pruning* of old blockchain blocks. It allows deleting old blocks after a certain period, saving storage space and making the blockchain-based solution more sustainable. The HF does not have such functionality at the moment, although it was investigated and tested in [49].

8.6.6 Private Blockchain Lesser Energy Consumption

Several blockchain implementations, *e. g.,* Bitcoin, have raised concern from environmentalists regarding abnormal energy consumption and subsequent carbon emissions [50]. Large energy consumption may lead to carbonization of Earth's atmosphere and, in the long-term perspective, cause a harmful effect on humankind. There are two major blockchain architectures, *i. e.,* public and private. Both architectures assume a certain level of system decentralization. Some public systems, *e. g.,* Bitcoin, provide a level of decentralization that allows for making systems entirely digitally sovereign [51]. However, this comes at the cost of computationally heavy consensus algorithms such as PoW. Further, the private blockchain architecture assumes a certain degree of centralization within the system, *i. e.,* reliance on TTP such as a regulator in the energy marketplace. However, an execute-order-validate consensus mechanism consumes minimal to non-existent computational power, which is more sustainable in a long-term approach [52].

8.6.7 Undesirable Energy Market Manipulation

While the blockchain-based energy marketplace brings many benefits to prosumers, there may be some drawbacks with the deregulation of the electricity trade market presented in this paper. In particular, we are concerned that when prosumers are allowed to compete in the marketplace, some may have enough RES to affect the prices unfairly for other actors, to manipulate the market to their advantage. This is akin to effects seen on the stock market when actors dump shares to lower prices or aggressively buy to increase prices, *e. g.,* pump-and-dump strategy. This sort of behavior may also be possible when a large number of small and medium prosumers collude in order to control the pricing. Whereas it is difficult to assess the likelihood and impact of this phenomenon at this time, we nonetheless think it is an interesting study for future work.

8.7 Related Work

The area of decentralized blockchain-based marketplaces generates research interest in academia and industry due to the widespread adoption of blockchain technology. Blockchain enables decentralized governance, identity management, and trade settlement execution. Thus, the energy marketplaces had a number of proposals in recent years in terms of blockchain technology incorporation. Such proposals are discussed next.

8.7.1 Blockchain-based Energy Trading

In [53], the authors propose a blockchain-based trading mechanism where locally generated electricity can be sold between prosumers via a marketplace system. The system is based on the Ethereum blockchain and utilizes its SCs to automate and execute the trade settlement process. According to the authors, blockchain technology enables P2P trading between energy prosumers. In addition, the authors conclude that SCs enable real-time trade settlement with minimum oversight. Authors of [3] investigate the possibility of blockchain-based marketplace creation for P2P energy trading. According to the authors, they aim to bring flexibility and transparency to all actors involved in the energy marketplace. In the proposed model, blockchain registers the amount of generated electricity and enables electricity prices regulation based on the prosumer generation rate. However, this prototype, as well as the one from [53], relies on the Ethereum blockchain, which is built as a public permissionless platform. In contrast, HF is a private permissioned platform that is geared toward organizational deployment. It includes better privacy-preserving characteristics, and its architecture is designed to cater to business needs. In [54], authors propose a blockchain-based platform for energy trading based on HF blockchain. In the proposed platform, blockchain technology helps reduce the need for TTP in electricity trading. The main aim of this proposal is to make the consumption and generation of electricity by prosumers more efficient and eventually reduce electricity bills. Authors of [55] propose a platform for blockchain-based P2P energy trading. The main aim of this platform is to provide prosumers with the ability to trade electricity after the generation period and not before, as it is done in energy markets today. According to the authors, blockchain technology enables bilateral energy trading after the generation period by enabling prosumers to trade electricity in P2P mode without the need for TTP. In [56], the authors propose a blockchain-based energy trading platform with the aim of enhancing the distribution of the energy generated by the DER. In addition, this study uses machine learning (ML) to analyze the generation data and propose better energy production and distribution strategies. Authors claim that the proposed model enhances energy crowdsourcing while maintaining QoS. Authors of [57] propose a P2P blockchain-based energy trading model that employs a double auction model. The double auction enables prosumers to place buying and selling orders. Further, the order price can be adjusted according to the market need, which, according to the authors, promotes the bidding strategies improvement and supply and demand alignment. Authors claim that the proposed trading model increases hourly social welfare by 22.3%. In [58], authors design a decentralized energy marketplace to enable P2P prosumer trade settlements. The main aim is to allow prosumers who own RES to sell electricity to neighboring households at a better price than the energy company offers. First, the authors use the mathematical game theory

for prosumers' behavior within the energy marketplace. Further, they utilize the blockchain to enable P2P trade settlements. Authors claim that the proposed marketplace can reduce electricity prices under certain conditions, *e. g.*, the energy generation is much higher than the demand in the grid. Authors of [59] propose an energy trading model which caters to prosumers' data privacy needs. This model utilizes consortium blockchain to preserve prosumers' data privacy without restricting trade settlement execution. Authors claim that the experimental evaluation demonstrates the effectiveness of the proposed approach, *i. e.*, automated trade settlement with data privacy preservation. In [12], authors propose a P2P energy marketplace for tokenized energy assets. All assets, *i. e.*, energy, batteries, and trade transactions, are expressed as either FTs or NFTs within the marketplace. Further, these assets are traded within the marketplace where each actor can benefit monetarily depending on its role. Authors claim that their implementation achieved a throughput of 448.3 TPS for the slowest SC function. Authors of [60] propose a P2P energy trading system where multiple market models are applied to provide high flexibility for prosumers. First, the prosumers can utilize P2P trade settlement with other market participants. Further, the prosumers can trade with the distribution system operator. Each market model is utilized depending on the electricity pricing during the day. The authors claim that the data collected can be analyzed to produce energy distribution optimization techniques, eventually resulting in the decarbonization of the grid. In [61], authors propose an automated P2P energy marketplace based on blockchain technology and a multi-agent system paradigm. The usage of permissioned blockchain brings such benefits as reduced transaction cost, elimination of single point of failure, and enables micro-transactions. According to the authors, blockchain technology enables the creation of a democratic energy marketplace while being compliant with current data regulations. For further read on the developments in blockchain-based energy marketplaces see [62].

8.7.2 Energy Grid Management

The topic of energy grid management has also gained traction in recent years, investigating topics such as efficient energy distribution strategies, demand-supply matching, and balancing electricity prices. Authors of [63] propose a blockchain-based virtual power plant (VPP) management platform. Here, authors address such aspects of VPP management as energy aggregation flexibility, operation of community microgrids, and P2P electricity trading. Authors claim that the proposed platform can successfully address the deployment and operation challenges connected to small-scale VPPs and promote grid decarbonization. In [13], authors present a framework to optimize the capacity of DERs aggregators using mixed non-linear programming. The increased resolution of the energy demand data improves optimization accuracy significantly, resulting in increased prosumer revenue. Authors claim that the energy trade revenue increases up to 29.8% in comparison to the currently used heuristic approach for capacity optimization.

8.7.3 Performance Evaluation

In [64], the authors conduct the performance evaluation of HF 1.2, where they aim to substitute conventional distributed databases with the blockchain distributed ledger technol-

ogy. The authors propose their HF modification called HF++, and according to them, it outperforms the original HF 1.2 by a factor of 3. However, such a performance evaluation is not put in a context of a use-case, where it would represent an execution of a SC tailored for a specific application. Such specific characteristics as the transaction size, infrastructure distribution, and a number of blockchain organizations may heavily affect the evaluation results. Moreover, our solution uses a more up-to-date HF 2.1 which underwent a number of revisions and modifications. Finally, HF 2.1 has a similar performance as that is proposed by authors modification HF++. The authors of [65] conduct the evaluation of HF versions 0.6 and 1.0, with the aim to compare their performance characteristics. The authors conclude that HF 1.0 performs better than HF 0.6. However, according to the authors, even 1.0 can not reach the performance level needed in current traditional database systems. In [66], the authors perform a performance evaluation of HF 1.4, aiming to identify the main bottlenecks. Authors investigate different ordering services such as Solo, Kafka, and RAFT as well as different endorsement policies. Authors conclude that the validation phase is the major bottleneck of HF 1.4, with ordering not making a big impact on performance. However, the authors do not specify such external bottlenecks as storage IOPS and do not investigate ledger read throughput. Finally, our HF 2.1 based solution shows approximately 25% better performance in terms of maximum throughput. The authors of [67] describe their HF-based proposal to scale the throughput up to 20000 transactions per second. According to the authors, they achieve such a throughput increase by focusing on the existing performance bottlenecks and solving them through modifications to the core code of HF. However, the changes made to the HF modify the core code so much that the resulting blockchain software has lost any resemblance to HF. In our work, we specifically investigated the original HF to evaluate its capabilities for our use case. Furthermore, the evaluation of original technology enables us to judge the improvement which is introduced in future versions of HF.

The related works demonstrate that blockchain applicability in an energy marketplace has been relatively well defined at an abstract level. However, all works mentioned above lack the detailed requirements definition, implementation details, and discussion on the technical limitations of blockchain technology incorporation. In addition, the aforementioned related works lack discussion on the GOs and the regulator role. In this work, we discuss the decentralized energy marketplace from a technical perspective to provide insights into challenges encountered during our test implementation and operation.

8.8 Summary and Outlook

In this work, we describe a decentralized blockchain-based P2P energy marketplace. It utilizes the private permissioned blockchain platform Hyperledger Fabric. The main aim of such a marketplace is to enable trusted and transparent P2P energy trading. Also, the marketplace provides data privacy to its actors, catering to the needs of prosumers and energy providers. To achieve this aim, the following methodology was used. First, with advice from an operating energy provider, we define a set of *regulatory* and *operational requirements*, which have to be met by the marketplace. Next, based on defined requirements, we describe the proposed *marketplace architecture*. Further, we produce the marketplace's *threat model* and define *security requirements*. Next, we detail the implementation and map

security requirements to appropriate countermeasures provided by HF. Further, we present the marketplace's performance evaluation with the SC tailored for energy trading.

The functional and non-functional requirements defined in Section 8.2 are aligned with the Directive 2018/2001 of the European Parliament on the promotion of the use of energy from renewable sources. The energy marketplace, as a platform, should attract prosumers into installing RES due to the opportunity to become energy-independent and prosper from generated energy selling. The ultimate goal of RES promotion and widespread adoption is the decarbonization of the atmosphere. Further, with the marketplace, prosumers can choose where, when, and at what price they sell or buy electricity. Also, the marketplace opens new opportunities for energy providers. With the blockchain ledger, the energy providers obtain a decentralized database that can be used for accounting and efficient energy distribution strategies discovery. Further, it is used as a source of irrefutable evidence in case of legal disputes between marketplace actors. Finally, it is used as a source of digital trust, which energy providers can rely on when executing energy trade transactions.

The regulator represents a governmental entity that regulates the GO issuing and consumption within the marketplace. It also regulates the process of electricity generation and registration within the marketplace. Further, the regulator participates in the electricity trade settlement as a guarantor that the GO is not sold more than once. Thus, the regulator can be viewed as a TTP who oversees all the processes that are executed within the marketplace. However, even with a certain level of centralization around the regulator, the marketplace enables automation of P2P trade settlement process and connects the end-users of the energy grid, *i. e.,* prosumers, in a trusted and transparent way. Although, it should be noted that the maximum electricity selling price is established by the government, *i. e.,* the regulator.

The performance evaluation shows that HF has better performance characteristics than other well-known blockchain solutions such as Ethereum. However, a number of performance bottlenecks remain, such as the validation phase and possible BFT consensus scalability concerns. The validation phase requires a considerable amount of computational capabilities. Thus, optimization is required to increase the throughput and scalability of HF overall. With the current HF's development state, when designing the blockchain-based system, one has to carefully consider the data which is stored on the ledger. Since the HF is limited in throughput and restricted by data regulations, *e. g.,* GDPR, the ledger only has to save the data, which is critically needed to establish digital trust between system actors. In the case of the P2P energy marketplace, it is the GO and prosumer wallet records.

The proposed approach to designing a blockchain-based marketplace can also be applied in the case of telecommunication services marketplaces [33]. The communication service providers can participate in the decentralized marketplace where they can sell access to their network hardware via automated SC that ensures trusted transaction execution and dispute resolution in case of service-level agreement violations. Further, we believe that the system design described here can be generalized toward digital marketplaces in other domains.

Future work will focus on the investigation of possible improvements for consensus and policy execution of such a blockchain-based marketplace in order to improve its capabilities and efficiency of operation. Quantitative characteristics such as throughput and scalability are the main indicators of performance. Thus, they are used as main improvement indicators.

In addition, the investigation of a trade-off between the increased performance and security of a blockchain solution is of interest.

Acknowledgment

The authors would like to acknowledge the funding project supporting the work presented in this paper: The work was partly sponsored by the Swedish Knowledge Foundation through the project *Symphony - Supply-and-Demand-based Service Exposure using Robust Distributed Concepts*. The project partners in Symphony are Ericsson AB (Stockholm, Sweden) and Affärsverken Energi AB (Karlskrona, Sweden).

References

[1] B. Jasim and P. Taheri. "An Origami-Based Portable Solar Panel System". In: *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. 2018, pp. 199–203. DOI: 10.1109/IEMCON.2018.8614997.

[2] Y. Yang, S. Zhang, and Y. Xiao. "Optimal design of distributed energy resource systems coupled with energy distribution networks". In: *Energy* 85 (2015), pp. 433–448. DOI: 10.1016/j.energy.2015.03.101.

[3] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. "Blockchain Based Decentralized Management of Demand Response Programs in Smart Energy Grids". In: *Sensors* 18.2 (2018), pp. 1–21. DOI: 10.3390/s18010162.

[4] L. Strezoski, H. Padullaparti, F. Ding, and M. Baggu. "Integration of Utility Distributed Energy Resource Management System and Aggregators for Evolving Distribution System Operators". In: *Journal of Modern Power Systems and Clean Energy* 10 (2 Mar. 2022), pp. 277–285. DOI: 10.35833/MPCE.2021.000667.

[5] EU Parliament. *Directives Directive (EU) 2018/2001 of the European Parliament*. 2022, pp. 82–209. URL: http://data.europa.eu/eli/dir/2018/2001/2022-06-07 (visited on 06/18/2023).

[6] Á. Hamburger. "Is guarantee of origin really an effective energy policy tool in Europe? A critical approach". In: *Society and Economy* 41 (2019), pp. 487–507. DOI: 10.1556/204.2019.41.4.6.

[7] R.-V. Tkachuk, D. Ilie, and K. Tutschku. "Orchestrating Future Service Chains in the Next Generation of Clouds". In: *Proceedings of 15th Swedish National Computer Networking Workshop*. June 2019, pp. 1–5. URL: https://urn.kb.se/resolve?urn=urn:nbn:se:bth-18785.

[8] B. Hertz-Shargel, D. Livingston, and A. C. of the United States. *Assessing Blockchain's future in transactive energy*. 2019. ISBN: 9781619775992. URL: https://www.atlanticcouncil.org/in-depth-research-reports/report/assessing-blockchains-future-in-transactive-energy/ (visited on 06/18/2023).

[9] T. Kollmann, S. Hensellek, K. de Cruppe, and A. Sirges. “Toward a renaissance of cooperatives fostered by Blockchain on electronic marketplaces: a theory-driven case study approach”. In: *Electronic Markets* 30.2 (2020), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

[10] F. Rahimi, S. Nikhil, G. Gourisetti, J. Kempf, E. Alejandro, A. Flores, C. Lima, H. Albright, P. D. Heitmann, and T. Martinez. *IEEE Blockchain Transactive Energy (BCTE)*. 2021. URL: https://blockchain.ieee.org/verticals/transactive-energy (visited on 06/18/2023).

[11] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick. “Hyperledger Fabric”. In: *Proceedings of the Thirteenth EuroSys Conference*. 2018, pp. 1–15. DOI: 10.1145/3190508.3190538.

[12] N. Karandikar, A. Chakravorty, and C. Rong. “Blockchain Based Transaction System with Fungible and Non-Fungible Tokens for a Community-Based Energy Infrastructure”. In: *Sensors* 21 (11 May 2021), pp. 1–32. DOI: 10.3390/s21113822.

[13] U. Rai, G. Oluleye, and A. Hawkes. “An optimisation model to determine the capacity of a distributed energy resource to contract with a balancing services aggregator”. In: *Applied Energy* 306 (2022), pp. 1–22. DOI: 10.1016/j.apenergy.2021.117984.

[14] Ausgrid. *Community Batteries*. Australia, 2022. URL: https://www.ausgrid.com.au/In-your-community/Community-Batteries (visited on 06/18/2023).

[15] H. Akay and S.-G. Kim. “Reading functional requirements using machine learning-based language processing”. In: *CIRP Annals* 70 (1 Jan. 2021), pp. 139–142. DOI: 10.1016/j.cirp.2021.04.021.

[16] K. B. Wilson, A. Karg, and H. Ghaderi. “Prospecting non-fungible tokens in the digital economy: Stakeholders and ecosystem, risk and opportunity”. In: *Business Horizons* 65 (5 Sept. 2022), pp. 657–670. DOI: 10.1016/j.bushor.2021.10.007.

[17] Q. Wang, R. Li, Q. Wang, and S. Chen. *Non-Fungible Token (NFT): Overview, Evaluation, Opportunities and Challenges*. 2021. URL: http://arxiv.org/abs/2105.07447.

[18] K. Khatter and DevanjaliRelan. “Non-functional requirements for blockchain enabled medical supply chain”. In: *International Journal of System Assurance Engineering and Management* 13 (3 June 2022), pp. 1219–1231. DOI: 10.1007/s13198-021-01418-y.

[19] A. Qazi, F. Hussain, N. A. Rahim, G. Hardaker, D. Alghazzawi, K. Shaban, and K. Haruna. “Towards Sustainable Energy: A Systematic Review of Renewable Energy Sources, Technologies, and Public Opinions”. In: *IEEE Access* 7 (2019), pp. 63837–63851. DOI: 10.1109/ACCESS.2019.2906402.

[20] M. Binjubeir, A. A. Ahmed, M. A. B. Ismail, A. S. Sadiq, and M. K. Khan. “Comprehensive Survey on Big Data Privacy Protection”. In: *IEEE Access* 8 (2020), pp. 20067–20079. DOI: 10.1109/ACCESS.2019.2962368.

[21] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. URL: https://bitcoin.org/bitcoin.pdf (visited on 06/18/2023).

[22] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and Y. Liu. "A Survey of Blockchain Technology Applied to Smart Cities: Research Issues and Challenges". In: *IEEE Communications Surveys and Tutorials* 21.3 (2019), pp. 2794–2830. DOI: 10.1109/COMST.2019.2899617.

[23] M. Liu, K. Wu, and J. J. Xu. "How Will Blockchain Technology Impact Auditing and Accounting: Permissionless versus Permissioned Blockchain". In: *Current Issues in Auditing* 13.2 (2019). DOI: 10.2308/ciia-52540.

[24] IBM. *Hyperledger Fabric Internals*. 2020. URL: https://ibm.github.io/hlf-internals/ (visited on 06/18/2023).

[25] R. M. Parizi, Amritraj, and A. Dehghantanha. "Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability and Security". In: *Lecture Notes in Computer Science*. Vol. 10974 LNCS. 2018, pp. 75–91. DOI: 10.1007/978-3-319-94478-4_6.

[26] E. Abad-Segura, A. Infante-Moro, M.-D. González-Zamar, and E. López-Meneses. "Blockchain Technology for Secure Accounting Management: Research Trends Analysis". In: *Mathematics* 9.14 (2021), pp. 1–26. DOI: 10.3390/math9141631.

[27] S. Wang, M. Yang, Y. Zhang, Y. Luo, T. Ge, X. Fu, and W. Zhao. "On Private Data Collection of Hyperledger Fabric". In: *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, July 2021, pp. 819–829. DOI: 10.1109/ICDCS51616.2021.00083.

[28] D. Woos, J. R. Wilcox, S. Anton, Z. Tatlock, M. D. Ernst, and T. Anderson. "Planning for change in a formal verification of the raft consensus protocol". In: *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs*. ACM, 2016, pp. 154–165. DOI: 10.1145/2854065.2854081.

[29] S. Nathan, C. Govindarajan, A. Saraf, M. Sethi, and P. Jayachandran. "Blockchain Meets Database: Design and Implementation of a Blockchain Relational Database". In: (2019). URL: http://arxiv.org/abs/1903.01919.

[30] Hyperledger. *Hyperledger Fabric: The Ordering Service*. 2022. URL: https://hyperledger-fabric.readthedocs.io/en/release-2.2/orderer/ordering_service.html (visited on 06/18/2023).

[31] R.-V. Tkachuk, D. Ilie, and K. Tutschku. "Towards a Secure Proxy-based Architecture for Collaborative AI Engineering". In: *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*. Naha, Japan: IEEE, 2020, pp. 373–379. DOI: 10.1109/CANDARW51189.2020.00077.

[32] A. Shostack. *Threat Modeling: Designing for Security*. Wiley, 2014.

[33] R.-V. Tkachuk, D. Ilie, K. Tutschku, and R. Robert. "A Survey on Blockchain-Based Telecommunication Services Marketplaces". In: *IEEE Transactions on Network and Service Management* 19.1 (2022), pp. 228–255. DOI: 10.1109/TNSM.2021.3123680.

[34] S. P. Otta and S. Panda. “Decentralized Identity and Access Management of Cloud for Security as a Service”. In: *2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS)*. IEEE, Jan. 2022, pp. 299–303. DOI: 10.1109/COMSNETS53615.2022.9668529.

[35] A. S. Wazan, R. Laborde, D. W. Chadwick, R. Venant, A. Benzekri, E. Billoir, and O. Alfandi. “On the Validation of Web X.509 Certificates by TLS Interception Products”. In: *IEEE Transactions on Dependable and Secure Computing* 19 (1 Jan. 2022), pp. 227–242. DOI: 10.1109/TDSC.2020.3000595.

[36] EU Parliament. *Regulation (EU) 2016/679 of the European Parliament (General Data Protection Regulation)*. 2016, pp. 1–99. URL: https://gdpr-info.eu/ (visited on 06/18/2023).

[37] R. Mukta, H.-y. Paik, Q. Lu, and S. S. Kanhere. “A survey of data minimisation techniques in blockchain-based healthcare”. In: *Computer Networks* 205 (Mar. 2022), p. 108766. DOI: 10.1016/j.comnet.2022.108766.

[38] G. Caldarelli. “Real-world Blockchain Applications Under the Lens of the Oracle Problem. A Systematic Literature Review”. In: *2020 IEEE International Conference on Technology Management, Operations and Decisions (ICTMOD)*. IEEE, 2020, pp. 1–6. DOI: 10.1109/ICTMOD49425.2020.9380598.

[39] A. Vescan, C. Serban, and G. C. Crisan. “Software Defects Rules Discovery”. In: *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*. IEEE, Apr. 2021, pp. 101–109. DOI: 10.1109/ICSTW52544.2021.00028.

[40] Y.-F. Wen and C.-M. Hsu. “A performance evaluation of modular functions and state databases for Hyperledger Fabric blockchain systems”. In: *The Journal of Supercomputing* 79 (3 Feb. 2023), pp. 2654–2690. DOI: 10.1007/s11227-022-04762-3.

[41] W. Chen and I. Paik. “Toward Better Quality of Service Composition Based on a Global Social Service Network”. In: *IEEE Transactions on Parallel and Distributed Systems* 26.5 (2015), pp. 1466–1476. DOI: 10.1109/TPDS.2014.2320748.

[42] Hyperledger Fabric. *Performance Evaluation Report (v2.1)*. 2022. URL: https://hyperledger.github.io/caliper-benchmarks/fabric/performance/2.1.0/goContract/nodeSDK/configuration/ (visited on 06/18/2023).

[43] L. Hang and D.-H. Kim. “Optimal blockchain network construction methodology based on analysis of configurable components for enhancing Hyperledger Fabric performance”. In: *Blockchain: Research and Applications* 2 (1 Mar. 2021), pp. 1–12. DOI: 10.1016/j.bcra.2021.100009.

[44] M. Schäffer, M. di Angelo, and G. Salzer. “Performance and Scalability of Private Ethereum Blockchains”. In: *Lecture Notes in Business Information Processing* 361 (2019), pp. 103–118. DOI: 10.1007/978-3-030-30429-4_8.

[45] M. Vukolić. “The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication”. In: *Lecture Notes in Computer Science* 9591 (2016). Ed. by J. Camenisch and D. Kesdoğan, pp. 112–125. DOI: 10.1007/978-3-319-39028-4_9.

[46] A. S. Asrese, S. J. Eravuchira, V. Bajpai, P. Sarolahti, and J. Ott. "Measuring Web Latency and Rendering Performance: Method, Tools, and Longitudinal Dataset". In: *IEEE Transactions on Network and Service Management* 16 (2 June 2019), pp. 535–549. ISSN: 1932-4537. DOI: 10.1109/TNSM.2019.2896710.

[47] G. Wood. *Ethereum: a secure decentralised generalised transaction ledger*. Ethereum Project Yellow Paper, 2014, pp. 1–32. URL: https://gavwood.com/paper.pdf (visited on 06/18/2023).

[48] J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, and R. Buyya. "Ensuring Security and Privacy Preservation for Cloud Data Services". In: *ACM Computing Surveys* 49 (1 Mar. 2017), pp. 1–39. DOI: 10.1145/2906153.

[49] E. Palm, O. Schelen, and U. Bodin. "Selective Blockchain Transaction Pruning and State Derivability". In: IEEE, June 2018, pp. 31–40. DOI: 10.1109/CVCBT.2018.00009.

[50] J. Sedlmeir, H. U. Buhl, G. Fridgen, and R. Keller. "Recent Developments in Blockchain Technology and their Impact on Energy Consumption". In: *Informatik Spektrum* 43 (6 Feb. 2021), pp. 391–404. DOI: 10.1007/s00287-020-01321-z.

[51] Morrow and Zarrebini. "Blockchain and the Tokenization of the Individual: Societal Implications". In: *Future Internet* 11 (10 Oct. 2019), pp. 1–12. DOI: 10.3390/fi11100220.

[52] A. O. Bada, A. Damianou, C. M. Angelopoulos, and V. Katos. "Towards a Green Blockchain: A Review of Consensus Mechanisms and their Energy Consumption". In: *2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, July 2021, pp. 503–511. DOI: 10.1109/DCOSS52077.2021.00083.

[53] M. Sabounchi and J. Wei. "Towards resilient networked microgrids: Blockchain-enabled peer-to-peer electricity trading mechanism". In: *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*. IEEE, Nov. 2017, pp. 1–5. DOI: 10.1109/EI2.2017.8245449.

[54] S. Saxena, H. Farag, A. Brookson, H. Turesson, and H. Kim. "Design and Field Implementation of Blockchain Based Renewable Energy Trading in Residential Communities". In: *2019 2nd International Conference on Smart Grid and Renewable Energy (SGRE)*. IEEE, Nov. 2019, pp. 1–6. DOI: 10.1109/SGRE46976.2019.9020672.

[55] J. Mello, J. Villar, R. J. Bessa, M. Lopes, J. Martins, and M. Pinto. "Power-to-Peer: a blockchain P2P post-delivery bilateral local energy market". In: *2020 17th International Conference on the European Energy Market (EEM)*. 2020, pp. 1–5. DOI: 10.1109/EEM49802.2020.9221901.

[56] F. Jamil, N. Iqbal, Imran, S. Ahmad, and D. Kim. "Peer-to-Peer Energy Trading Mechanism Based on Blockchain and Machine Learning for Sustainable Electrical Power Supply in Smart Grid". In: *IEEE Access* 9 (2021), pp. 39193–39217. DOI: 10.1109/ACCESS.2021.3060457.

[57] C. Zhang, T. Yang, and Y. Wang. "Peer-to-Peer energy trading in a microgrid based on iterative double auction and blockchain". In: *Sustainable Energy, Grids and Networks* 27 (2021). DOI: 10.1016/j.segan.2021.100524.

[58] G. Yun, V. Zhygulin, and Q. P. Zheng. “Residential energy trading with blockchain technology”. In: *Energy Systems* 12 (3 Aug. 2021), pp. 619–636. DOI: 10.1007/s12667-021-00426-y.

[59] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen. “Privacy-Preserving Energy Trading Using Consortium Blockchain in Smart Grid”. In: *IEEE Transactions on Industrial Informatics* 15 (6 June 2019), pp. 3548–3558. DOI: 10.1109/TII.2019.2893433.

[60] H. Görgülü, Y. Topçuoğlu, A. Yaldız, T. Gökçek, Y. Ateş, and O. Erdinç. “Peer-to-peer energy trading among smart homes considering responsive demand and interactive visual interface for monitoring”. In: *Sustainable Energy, Grids and Networks* 29 (Mar. 2022), pp. 1–19. DOI: 10.1016/j.segan.2021.100584.

[61] Y. Mezquita, A. B. Gil-González, A. M. del Rey, J. Prieto, and J. M. Corchado. “Towards a Blockchain-Based Peer-to-Peer Energy Marketplace”. In: *Energies* 15 (9 Apr. 2022), pp. 1–20. DOI: 10.3390/en15093046.

[62] S. Gawusu, X. Zhang, A. Ahmed, S. A. Jamatutu, E. D. Miensah, A. A. Amadu, and F. A. J. Osei. “Renewable energy sources from the perspective of blockchain integration: From theory to application”. In: *Sustainable Energy Technologies and Assessments* 52 (2022), pp. 1–26. DOI: 10.1016/j.seta.2022.102108.

[63] T. Cioara, C. Pop, R. Zanc, I. Anghel, M. Antal, and I. Salomie. “Smart Grid Management Using Blockchain: Future Scenarios and Challenges”. In: *2020 19th RoEduNet Conference: Networking in Education and Research (RoEduNet)*. 2020, pp. 1–5. DOI: 10.1109/RoEduNet51892.2020.9324874.

[64] A. Sharma, F. M. Schuhknecht, D. Agrawal, and J. Dittrich. *How to Databasify a Blockchain: the Case of Hyperledger Fabric*. 2018. URL: https://arxiv.org/abs/1810.13177.

[65] Q. Nasir, I. A. Qasse, M. A. Talib, and A. B. Nassif. “Performance Analysis of Hyperledger Fabric Platforms”. In: *Security and Communication Networks* 2018 (Sept. 2018), pp. 1–14. DOI: 10.1155/2018/3976093.

[66] C. Wang and X. Chu. “Performance Characterization and Bottleneck Analysis of Hyperledger Fabric”. In: *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. Vol. 2020-Novem. IEEE, Nov. 2020, pp. 1281–1286. DOI: 10.1109/ICDCS47774.2020.00165.

[67] C. Gorenflo, S. Lee, L. Golab, and S. Keshav. “FastFabric: Scaling Hyperledger Fabric to 20,000 Transactions per Second”. In: *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. IEEE, May 2019, pp. 455–463. DOI: 10.1109/BLOC.2019.8751452.

Chapter Nine

On the Performance and Scalability of Consensus Mechanisms in Privacy-Enabled Decentralized Renewable Energy Marketplace

Abstract

Renewable energy sources were introduced as an alternative to fossil fuel sources to make electricity generation cleaner. However, today's renewable energy markets face a number of limitations, such as inflexible pricing models and inaccurate consumption information. These limitations can be addressed with a decentralized marketplace architecture. Such architecture requires a mechanism to guarantee that all marketplace operations are executed according to predefined rules and regulations. One of the ways to establish such a mechanism is blockchain technology. This work defines a decentralized blockchain-based peer-to-peer (P2P) energy marketplace which addresses actors' privacy and the performance of consensus mechanisms. The defined marketplace utilizes private permissioned Ethereum-based blockchain client Hyperledger Besu (HB) and its smart contracts to automate the P2P trade settlement process. Also, to make the marketplace compliant with energy trade regulations, it includes the regulator actor, which manages the issue and consumption of guarantees of origin and certifies the renewable energy sources used to generate traded electricity. Finally, the proposed marketplace incorporates privacy-preserving features, allowing it to generate private transactions and store them within a designated group of actors. Performance evaluation results of HB-based marketplace with three main consensus mechanisms for private networks, *i. e.*, Clique, IBFT 2.0, and QBFT, demonstrate a lower throughput than another popular private permissioned blockchain platform Hyperledger Fabric (HF). However, the lower throughput is a side effect of the Byzantine Fault Tolerant characteristics of HB's consensus mechanisms, *i. e.*, IBFT 2.0 and QBFT, which provide increased security compared to HF's Crash Fault Tolerant consensus RAFT.

9.1 Introduction

Energy distribution systems play a vital role in the modern world. The dependency on electricity supply transcends every aspect of a society's operation, making it a necessity. However, the electricity production conducted by power plants that work on fossil fuels results in atmosphere carbonization. In order to make electricity generation cleaner, renewable energy sources (RESs), *e. g.*, solar panels, were introduced as an alternative to fossil fuel ones. Consequently, the introduction of RES opened opportunities for electricity prosumers, *i. e.*, producers/consumers, to become a part of the grid as a distributed energy resource

(DER) [1]. This allows prosumers to not only consume energy as a conventional node but also to produce and output it to the energy grid [2]. Further, prosumers can also trade the produced electricity through the energy marketplace, which incentivizes the installation of RES and the production of green electricity. However, today's energy markets face a number of challenges when it comes to management and operation.

The first is the *inflexible pricing model* of today's marketplaces. In such a model, the prosumer is limited to selling the generated electricity to a single buyer without any other options, *e. g. ,* it is sold to an energy provider who owns the grid to which the prosumer's RES is connected. In addition, the generated electricity is sold at a price set by the buyer through a governmental body, *e. g. ,* country's energy agency regulates the margins for the RES-produced electricity trade and does not provide any room for negotiation. This creates a number of limitations for prosumers within an energy marketplace. It limits the volume of consumers that the prosumer can reach to sell their RES-produced electricity. Further, the seller cannot reach consumers belonging to a different electricity provider. Finally, this challenge results in a value distribution imbalance, where the prosumer side is losing a part of electricity sale profits due to price inflexibility [3].

The second challenge is *inaccurate green consumption information*, *i. e. ,* buyers receive unreliable information about the sources of the electricity they consume. Ultimately, the consumers are ready to pay higher electricity prices for RES-produced electricity to support the decarbonization of the atmosphere. Due to the inflexibility of the energy grid and inaccurate national regulatory frameworks, consumers frequently end up using electricity generated by fossil fuel sources while paying for RES-generated energy. This results in the devaluation of RES-produced electricity as prosumers do not see the benefit in buying it while being supplied with fossil fuel produced energy. Nowadays, the information about RES-produced electricity is recorded in the *guarantee of origin* (GO). GO is proof to the buyer that the electricity at a given quantity was produced by the RES [4]. Typically, the GO is issued by the governmental *regulator*, who certifies the prosumer-owned RES and an associated electricity generation metering device. However, due to the inflexibility of energy distribution systems, *e. g. ,* unavailability of RES in close proximity to consumers, they still end up using the electricity which was produced by fossil fuel energy sources while having the GO [5].

These limitations can be alleviated by introducing the *peer-to-peer* (P2P) *electricity trading*, which is an automated sale process for renewable energy between market participants using a contract with pre-determined conditions [4]. A P2P energy trade settlement allows prosumers to trade electricity directly with each other, enabling them to control when, where, and for what price the electricity is bought or sold. The ultimate goal of P2P energy trading is to create an incentive for the widespread adoption of RESs, resulting in the decarbonization of the energy distribution systems [6].

Today's marketplaces are built as centralized systems. Thus, a *trusted third-party* (TTP) (typically a prosumer's energy provider) has to be present to guarantee that the predetermined conditions of a P2P energy trading contract are followed. *However, trust issues are raised, when it comes to scaling the marketplace to more than one energy provider.* Energy providers want to keep their operations private to maintain a competitive advantage in the electricity market. This requires the introduction of an external TTP that can be trusted by all

energy providers within the marketplace [7], *i. e.,* allowing individuals belonging to different energy providers to trade with each other. To remediate these limitations, a *decentralized marketplace architecture* can be used to distribute control over the marketplace operations to multiple energy providers. However, all organizations require an efficient and robust consensus-reaching mechanism that provides guarantees that P2P trade settlement conditions are followed while maintaining actors' data privacy. Such capabilities can be provided by blockchain technology [8]. Blockchain provides marketplace participants with distributed storage, *i. e.,* the ledger, and brings such benefits as provenance, accountability, and privacy to all data processed in a system. It also acts as a consensus-reaching platform, allowing initially non-trusting energy providers and prosumers to establish a trusted relationship and conduct P2P trade settlements without needing a single TTP acting as a middleman [9].

Based on the challenges discussed above, the main contributions of this study can be summarized as follows. This study defines a decentralized blockchain-based P2P energy marketplace that utilizes *Hyperledger Besu* (HB) [10] as the blockchain platform. The proposed marketplace utilizes HB's *smart contracts* (SCs) to automate P2P energy trade settlement and issue and consume GOs. To make the marketplace compliant with energy trade regulations, it incorporates the *regulator* actor, which manages the issue and consumption of GO and certifies the RES used to generate traded electricity. Further, the marketplace utilizes Tessera private transaction manager to ensure actor data privacy. The following methodology was used to define the proposed marketplace. First, with advice from an energy provider, we define a set of regulatory and operational requirements. Further, we define the marketplace's architecture and detail its implementation. Next, we present the performance evaluation with the SC tailored for P2P energy trading. We investigate in-depth the performance of the main Proof of Authority (PoA) consensus mechanisms supported by HB, *i. e.,* QBFT, IBFT 2.0, and Clique. Finally, we summarize observations on the mechanisms that lead to secure consensus while preserving actors' data privacy. This paper is an extension of [11], and provides extended discussions on system architecture, implementation, performance evaluation results, and a summary.

The remainder of the paper is structured as follows. Section 9.2 describes the actors for the proposed marketplace and details its blockchain platform. Section 9.3 details the marketplace implementation, data structure, and smart contract definition. Section 9.4 details the performance evaluation process and results. Section 9.5 describes the observations from marketplace implementation. Section 9.6 describes related work on energy marketplaces and their performance evaluation. Finally, Section 9.7 summarizes the investigation of the proposed marketplace and provides an outlook.

9.2 Blockchain-based Energy Marketplace

The energy marketplace is subject to several regulatory constraints, which must be met to satisfy current P2P energy trade regulations, *i. e.,* GOs and an automated trade contract. Thus, the proposed marketplace requirements are aligned with regulations described in Directive 2018/2001 (D2018/2001) of the European Parliament [4] regarding the issuing, trading, and consumption of GOs. To align with D2018/2001, we introduce a *regulator* role in the proposed marketplace. The regulator is an actor that manages the issue and consumption

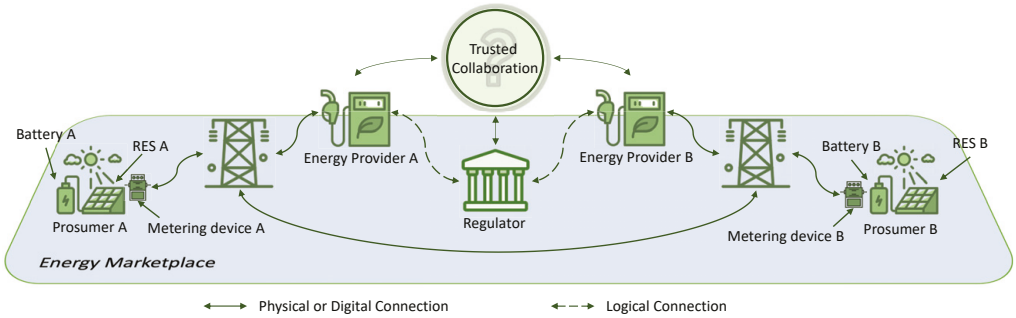


Figure 9.1: *Energy Marketplace Actors.* (The "Trusted Collaboration" component signifies the need to enforce rules that all actors of the marketplace follow).

of GOs, which are required to execute a trade settlement contract. Further, the regulator certifies the RES used to generate traded electricity. To the best of our knowledge, none of the other energy marketplace studies consider such governmental regulatory requirements in conjunction with actors' data privacy. Finally, the marketplace actors and requirements were defined in collaboration with the authors' local energy provider, which has DERs as a part of their grid infrastructure.

9.2.1 Marketplace Actors and Requirements

The actors and requirements were defined in collaboration with the authors' local energy provider. which has DERs as part of its grid infrastructure. Further, the requirements were defined in compliance with the regulations described in D2018/2001. Energy marketplace actors and their respective places in the grid infrastructure are depicted in Figure 9.1.

The *prosumer* represents a DER in an energy grid with an installed RES. The prosumer's main interest in becoming a part of the marketplace is to control the conditions of energy trade settlement, *e. g.*, to sell electricity at a better price. Further, prosumers want to get GO for the electricity they produce within the marketplace's automated system.

The *energy provider* is an actor that manages the energy grid to which the prosumer is connected. As a local central point in the energy distribution scheme, the energy provider collects data on electricity consumption fluctuations to optimize distribution and conduct an accounting of the electricity and money flows in its network. Further, energy providers want to expand their DER infrastructure to meet customer demand for RES-generated energy delivery.

The *regulator* is the representative of governmental authority who manages the issue and consumption of GOs. The GO acts as proof that the electricity was generated with RES and must be presented during the trade transaction by the prosumer-seller. Further, the regulator is the entity that certifies prosumers' RES and ensures the correct mapping between the generated and marketplace-traded electricity.

9.2.1.1 Functional Requirements

To enable renewable energy trade settlement, a set of operations must be defined. The *functional requirements* [12] define the operations which can be executed by the prosumers, energy providers, and regulators within the marketplace.

Electricity Operations: The prosumers must have the ability to *manage the generated electricity*, which is represented by virtual kilowatt-hours (kWh) on the level of the marketplace trade operations. Within the marketplace, generated electricity acts as a *fungible token* (FT) [13]. First, the prosumer *registers generated electricity* by adding a number of virtual kWh to their marketplace account. This happens automatically via a metering device connected to the prosumer's RES. Further, the prosumer should be able to *trade the generated electricity*, *i. e.* , sell/buy electricity at a given price. The associated GO is consumed when the electricity is sold, making it impossible to further sell it to another prosumer.

Ordering System Operations: Prosumers propose the energy trade through a *marketplace ordering system*. First, the marketplace should enable the creation of *offer to sell electricity* of a given quantity at a given price. Further, the marketplace should enable the creation of an *offer to buy electricity* of a given quantity at a given price.

GO Operations: The regulator issues the GO on the electricity generated by a specific RES. The GO acts as a *non-fungible token* (NFT) [14]. When the electricity is generated by a RES, the regulator should be able to *issue the GO* on the name of the prosumer, which is presented when the energy is sold. Further, when the electricity is sold, the GO should be *consumed*, to disable the double selling of energy.

9.2.1.2 Non-Functional Requirements

The *non-functional requirements* [15] define the global constraints which affect the marketplace system's reliability and data assurance.

Data Correctness: The marketplace must ensure that the virtual kWh must only be *issued following the actual generation of electricity*. When an order is executed, the marketplace has to *make sure that appropriate resources*, *i. e.* , virtual kWh and currency, *are available* for both seller and buyer. In addition, the *energy trade must be executed to a set of conditions that were previously approved by marketplace actors*. Further, the *GO must only be issued following the generation of electricity from renewable source*, *e. g.* , hydro, wind, or solar [16]. Finally, *it must be impossible to sell consumed GO*.

Data Privacy: To ensure data privacy [17] *all transactions from a prosumer, including generation, selling, and purchase, should not be disclosed to other prosumers*. Further, the *details of P2P energy trade should be disclosed only to the prosumers, their respective energy providers, and the regulator*. Finally, *prosumer energy generation information has to be visible for the regulator* to ensure correct mapping between virtual kWh and actual generated electricity.

9.2.2 Blockchain Platform

Blockchain technology [8] can be used to provide the technical building block allowing for meeting the requirements defined in Section 9.2.1. Blockchain provides marketplace

participants with distributed storage, *i. e.,* the ledger, and brings accountability and provenance to all data processed in a network. With a consensus mechanism, blockchain allows initially non-trusting energy providers, regulators, and prosumers to establish a trusted relationship [18]. Thus, blockchain technology removes the need for a single TTP accepted by all marketplace actors.

To enable marketplace actors to conduct P2P trade settlement, a blockchain platform has to be chosen such that it meets the identified requirements. Considering the privacy requirements, the proposed marketplace utilizes a private permissioned blockchain platform. Permissioned blockchain network has an identity and access management (IAM) [19] mechanism that defines a set of entities, *i. e.,* collaborating organizations and users, which are allowed to access the network. Further, permissioned blockchain requires that after entering the network, the entity has to be authorized to execute new transactions and add them to the global ledger. Finally, private blockchains enable data privacy and better address the demands of the business use-cases [20]. Here, the data privacy mechanism is defined as the ability to keep blockchain transactions private for a certain group of participants. *Hyperledger Besu* (HB) [10] is representative of private permissioned blockchain platform. It is an open-source Ethereum [21] client that PegaSys[1] first developed and later handed over to the Hyperledger Foundation[2]. From the beginning, the Ethereum blockchain was designed as a public permissionless platform, *i. e.,* opened for everyone to join and generate transactions. HB can be considered an adaptation of the original public Ethereum blockchain to the private context. Here, HB implements the *Enterprise Ethereum Alliance Protocol* to enable such functionality as private transactions, IAM, and permissioning. In the HB network, the *validator* nodes order, execute and verify transactions in the blockchain network. However, validator nodes cannot be used to initiate transactions. All transactions in the HB network are initiated by *user accounts*, representing a public and private key pair that can be generated off-chain. The smart contract (SC) defines functions that a user account can call to operate on the data in the ledger. First, the SC has to be installed in the blockchain network. Once installed, it serves as a predefined trade settlement contract where fixed, agreed-upon rules are enforced during every execution.

9.2.3 Identity and Access Management

The IAM in HB can be implemented using *local* and *on-chain* permissioning. The local permissioning is defined in a *permissions configuration file* and can be specified on each individual blockchain node. This permissioning type does not require consensus from the rest of the network. Local permissioning allows the specification of the list of valid nodes to which the validator can connect. In addition, it allows specifying the user accounts that can use the validator to execute transactions. In contrast, the on-chain permissioning is defined by the *permissioning management SC* and requires consensus of all nodes in the network. The SC acts as a program within a marketplace that collaborating organizations, *i. e.,* admins, install in the HB network to govern the IAM. Through the SC, admins can specify a list of nodes authorized to be a part of the network and perform consensus mechanism, *i. e.,*

[1] A team of engineers within ConsenSys company.
[2] https://www.hyperledger.org/

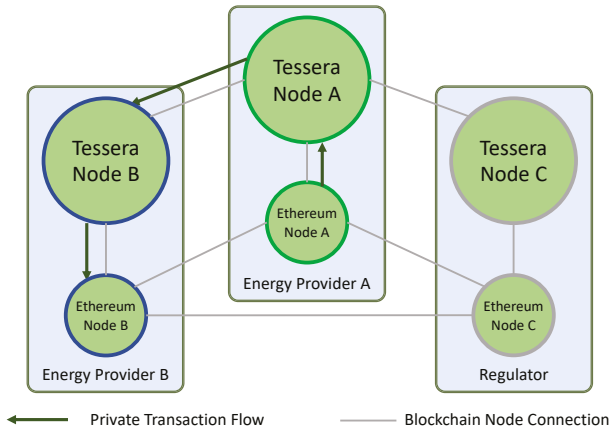


Figure 9.2: *Implemented Energy Marketplace.*

order, execute, and verify transactions. Further, through the SC, admins can specify the user accounts that can initiate transactions in the network. The advantage of on-chain permissioning is the ability to collectively govern access to the HB network and track the changes to access lists of full nodes and user accounts.

9.2.4 Data Privacy

In HB, private data is stored in transactions that are disclosed only to a subset of network participants (further referred to as privacy group (PG)), while the rest of the network does not have access to the contents. Further, the rest of the network does not know the list of nodes that belong to PG. The private transactions in HB are handled by the *Tessera* private transaction manager. Each organization in HB must have a Tessera node to participate in private transactions. When a new private transaction is generated, it is passed from the Ethereum node to the Tessera node associated with it. Further, the Tessera node encrypts the transaction and distributes it to the PG. Recipient Tessera Nodes from the PG decrypt the transaction and pass it to their Ethereum Nodes. The rest of the nodes outside of the PG receive the record confirming that the private transaction was executed. Such a record consists of a hash of the encrypted transaction data and a privacy marker, *i. e.* , indicator that the transaction is private. Further, this record is written into the global ledger.

The HB private transaction flow is shown in Figure 9.2. *Energy Provider A* and *Energy Provider B* are in the same PG. When *Ethereum Node A* generates the private transaction, it is passed to the *Tessera Node A* for encryption. Further, *Tessera Node A* transmits the encrypted transaction to *Tessera Node B*, where it is decrypted and written to the private storage of *Ethereum Node B*. Finally, the *Regulator's Ethereum Node C* receives the record with the encrypted transaction data and a privacy marker which is written to the global ledger.

In the public Ethereum network, *gas* is the unit of measurement for the number of computations needed to execute a transaction. The user is required to pay a certain amount

of *gas*, *i. e.,* depending on complexity, in order to execute a transition. In contrast, privacy-enabled HB Ethereum networks allow disabling gas spending to execute both ordinary and private transactions. This requires a certain level of trust among the blockchain network transacting nodes, *i. e.,* that none of the participants will act maliciously and perform a denial of service (DoS) attack by flooding the network with transactions. Thus, privacy-enabled networks must have off-chain trust-enabling mechanisms, including SC deployment recommendations and legal consequences for malicious activity.

9.2.5 Smart Contract

The smart contract (SC) in HB represents a concrete entity with functions that a user account can call. An SC cannot be triggered by any other internal HB network event or entity. Further, SCs are isolated in terms of storage, *i. e.,* each SC has its namespace and operates on the records saved there. However, one SC can invoke the functions of other SCs.

Initially, an SC is written in a high-level programming language, *e. g.,* Solidity[3] or Vyper[4]. These are domain-specific languages (DSLs), *i. e.,* defined to work specifically with Ethereum SCs. One prominent Ethereum DSL is Solidity, which is influenced by JavaScript in terms of syntax and structure. There are various open-source Solidity libraries that can be reused and adapted to a specific use case. Solidity enables the development of complex SCs, *i. e.,* the syntax and code constructions facilitate the implementation of complex function routines. One disadvantage is this may lead to an introduction of security vulnerabilities since complex routines may not behave as expected after the SC compilation [22], *i. e.,* due to the inexperience of the developer. Vyper is a DSL designed specifically for Ethereum to improve the auditability and security of SCs. In addition, Vyper DSL has a simpler syntax in comparison to Solidity. An SC code written in Vyper has built-in controls which prevent the introduction of security vulnerabilities. Further, Vyper SC can be more comprehensively reviewed by all collaborating parties relying on built-in security controls. After the SC is finalized in DSL, it is compiled into the *runtime bytecode*, *i. e.,* a state in which the SC is saved on the ledger. Further, SCs are executed in the Ethereum Virtual Machine (EVM), *i. e.,* an executable environment that is deployed locally for each validator.

9.2.6 Consensus Mechanisms

The consensus mechanism defines an algorithm by which all nodes in the network can agree on the validity of transaction order in the block. While proof of work (PoW) [8] worked in a public blockchain, it was unsuitable for private deployment, *i. e.,* low transaction throughput and high energy consumption to mine new blocks. Hence, a new approach was followed in private Ethereum called proof of authority (PoA). The blocks in PoA consensus mechanisms are not mined but signed by the designated pool of validators, *i. e.,* avoid wasting energy by delegating block creation to the trusted nodes.

Within the available consensus mechanisms, some are identified as Byzantine Fault Tolerant (BFT) and/or Crash Fault Tolerant (CFT) [23]. CFT consensus mechanisms are protected only from node failure, *i. e.,* if less than 50% of the nodes fail, the network can

[3] https://soliditylang.org/
[4] https://docs.vyperlang.org/

Table 9.1: *Besu consensus mechanisms comparison.*

Property	Ethash	PoS	Clique	IBFT 2.0	QBFT
Type	PoW	PoS	PoA	PoA	PoA
Finality	No	No	No	Yes	Yes
Quorum	1/2	1/2	NA	2/3	2/3
BFT	Yes	Yes	No	Yes	Yes
Liveness	1/2	1/2	1/2	1/3	1/3
Network	Public	Public	Private	Private	Private

operate successfully. BFT consensus provides the same level of protection as CFT and in addition can operate in the presence of adversaries, *e. g. ,* nodes that manipulate transactions and try to disrupt the blockchain network operation. However, there are limitations to the BFT consensus mechanisms in terms of the number of adversaries, *i. e. ,* consensus is jeopardized if more than *1/3* of the nodes collude. In practice, when the blockchain user account initiates the transaction, it must wait until the moment the *2m+1* responses are received, where m is the maximum number of allowed failed or malicious nodes. When *2m+1* responses are successfully received, the consensus is achieved and the state of the network is updated. The improved security of BFT consensus mechanisms may come at the cost of decreased performance compared to CFT ones.

The consensus mechanisms supported by the HB are PoW (Ethash), Proof of Stake (PoS), and PoA (Clique, IBFT 2.0, and QBFT). A brief summary of all HB consensus mechanisms characteristics is listed in Table 9.1. This study concentrates on PoA consensus mechanisms used in private HB networks. When comparing consensus mechanisms, such characteristics as *immediate finality*, *quorum*, *liveness*, and *throughput* have to be considered. Immediate finality refers to the ability to avoid forks, *i. e. ,* alternative blockchains or chain reorganizations. Quorum refers to the minimum number of validator nodes in the blockchain network. Liveness refers to how many failed validators it can sustain and continue normal operation. Throughput refers to the maximum number of write or read transactions, *c. f. ,* Section 9.4. The characteristics of each investigated consensus mechanism are discussed next.

Clique is a PoA consensus algorithm that was first proposed in the Ethereum Improvement Proposal (EIP) [24]. In Clique, a designated node pool of trusted *signers*, *i. e. ,* validators, creates and adds a new block to the ledger. Further, the existing pool of signers in Clique can vote to include a new or exclude an existing signer. The list of trusted signers is saved on the ledger to ensure that the correct pool is always accessible to every signer in the network. The block creation process is called *sealing*, where signers create new blocks at a fixed time interval defined in *blockperiodseconds*. When blockperiodseconds time is up, the block is cut and embedded into the ledger. To prevent malicious activity, every signer is allowed to seal a block once per *n/2+1* blocks, where n is the total number of signers. Thus, there are only *n-(n/2+1)* signers at a time that can seal a block. Clique consensus does not have immediate finality due to the possibility of creating a fork by proposing two different blocks at a time. Forks occur due to the process called *out-of-order* sealing. It implies that if the current block was not sealed in time, a new block could be proposed by another signer that waited for blockperiodseconds. Out-of-order sealing occurs if blockperiodseconds are configured to be too short for the network configuration, *i. e. ,* high latency between nodes and poor network

performance. Further, the higher the number of signers, the higher the chance of producing a fork in the blockchain network. Next, since Clique is not BFT, the minimum number of signers for Clique to operate is one. Finally, in terms of liveness, Clique can tolerate up to *1/2* of failed signers in a network.

IBFT 2.0 [25] is the Istanbul Byzantine Fault Tolerant (IBFT) PoA consensus mechanism. It is a variation of Practical BFT [26], which is applicable in blockchain networks. Originally, IBFT 1.0 [27] attempted to bring immediate finality and BFT into the block generation process, which was missing in the Clique consensus. However, Saltini in [28] proved that IBFT 1.0 is not BFT and does not guarantee immediate finality while operating in synchronous networks. This was fixed in IBFT 2.0. Similarly to Clique, IBFT 2.0 also has a designated list of signers, called *validators*. IBFT 2.0 achieves immediate finality and prevents the occurrence of forks in the blockchain. However, the minimum number of validators, *i. e.*, quorum, for IBFT 2.0 increased to four. It achieves quorum and is BFT only if up to *(n-1)/3* validators are malicious, where n is the total number of validator nodes. IBFT 2.0 achieves consensus in three distinct phases: *pre-prepare, prepare, and commit*. A new block is disseminated to all validators with a *pre-prepare* message. Then, validators broadcast *prepare* message. When receiving prepared replies from *2/3+1* of validators and achieving a quorum, the validator broadcasts *commit message*. When *commit* message is received by *2/3+1* of validators, the new block is written to the ledger. IBFT 2.0 communication complexity is $O(n^2)$. Finally, in terms of liveness, the IBFT 2.0 network can sustain up to *1/3* of the validators to fail.

QBFT or Quorum BFT [29] is the latest PoA consensus mechanism for HB private networks. It was proposed as a solution to the liveness and safety concerns of IBFT 2.0, *i. e.*, blockchain network DoS when two legitimate validators lock on different blocks. QBFT is similar to IBFT 2.0 in terms of immediate finality, quorum, and liveness. Similar to IBFT 2.0, it has communication complexity of $O(n^2)$ and follows a three-phase commit strategy. However, the difference is that in QBFT, if validators do not achieve consensus before a certain, predefined time expires, the validation round will reset, triggering a new consensus attempt. QBFT achieves immediate finality and prevents the occurrence of forks in the blockchain. Further, it achieves quorum and is BFT only up to *(n-1)/3* of malicious validator nodes. Finally, in terms of liveness, the QBFT can sustain up to *1/3* of the validator nodes to fail. QBFT is recommended by HB developers as the enterprise-grade consensus protocol for HB private networks.

9.3 Marketplace Implementation

The energy marketplace is depicted in Figure 9.3. It consists of two layers: *physical* and *digital*. The physical layer represents the electricity generation and distribution infrastructure. The digital layer represents the network infrastructure between the energy providers, regulators, and prosumers which enables the electricity trade. This work investigates the digital layer exclusively. Regarding the physical layer, we assume the regulator can correctly map generated electricity to the virtual kWh by combining information from metering devices in the electrical grid with the information stored in the blockchain ledger. Further, the regulator ensures that only certified RES and metering devices are installed in the physical

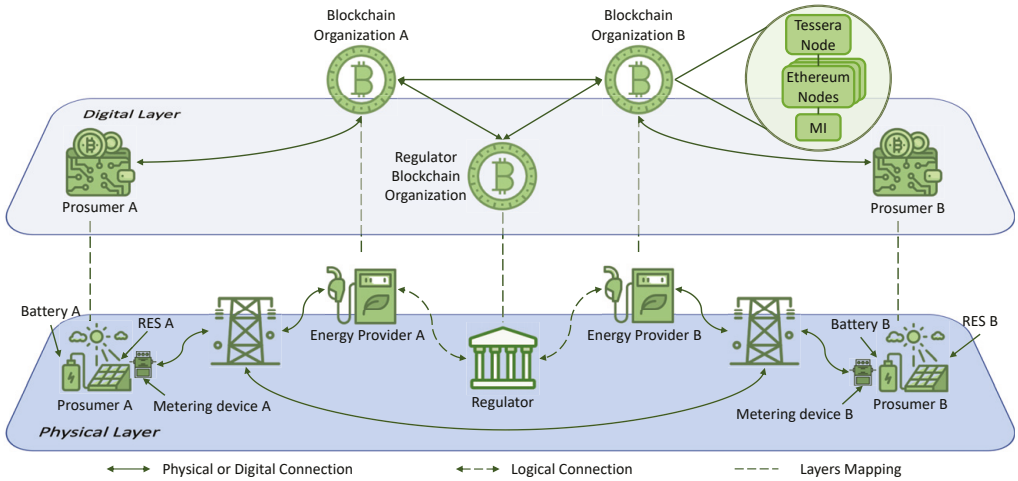


Figure 9.3: *Energy Marketplace*. (Physical layer, i.e., energy grid, is mapped to a digital blockchain-based layer, where the electricity trade operations are executed).

layer of the marketplace.

Each energy provider and regulator are represented within the marketplace as a *blockchain organization* (BO). Each BO must operate at least one validator node. The validator nodes are the main guarantors of valid transaction execution and require the most computational power. Further, each BO has a dedicated *Tessera* node to enable private transaction execution in the network. Finally, each BO has a marketplace interface (MI) that prosumers use to conduct P2P trade settlements.

In the marketplace implementation, prosumers are represented as user accounts. Since they do not operate any validators, they need to trust their energy provider's BO to execute transactions on their behalf. Before registration, the prosumer-owned RES must be certified by the regulator. Further, the RES is saved as a data record within PG, which includes the energy provider and regulator. During prosumer registration in the marketplace, the previously created RES record is attached to the prosumer record. In addition, the prosumer receives a personal wallet record where both fiat currency and bought electricity are stored. The energy provider registers the prosumer-generated electricity in the marketplace if the prosumer's RES is marked as certified. While trading, the prosumer utilizes an ordering system where buy or sell orders can be fulfilled according to a predefined marketplace SC.

9.3.1 Marketplace Execution Guarantees

Following this setup, HB provides the participants in the blockchain network, *i. e.*, the electricity providers and regulator, with two types of guarantees: *1)* the guarantee that the data stored in the ledger cannot be tampered with and *2)* the guarantee that it can only be modified following the rules implemented in the SC. These guarantees can be leveraged to fulfill the marketplace requirements. Firstly, by storing the GO in the ledger and encoding the rules governing their life-cycle in an SC, *i. e.*, issue and consumption, it is possible to

automate their management in a transparent fashion and guarantee that the legislation is followed. Secondly, the same principles can be applied to the management of electricity production, consumption, and trade settlement. By encoding the state of all the marketplace entities in the ledger, *i. e.,* prosumer, RES, and order, the marketplace ensures that there is always a consensus among all participants regarding the status of the marketplace as a whole. Further, by describing all the processes in the marketplace as a set of operations transforming this data and implementing these operations in the SC, it is possible to ensure that all the operations in the marketplace respect the agreed-upon rules.

One limitation of blockchain technology is that it can only provide guarantees after storing the data in the ledger. In other words, it cannot verify the validity of the data inserted in the ledger. In that regard, HB can only provide traceability for the data, recording which actor provided the information. The other marketplace actors must either trust that actor to provide correct information or rely on external processes to verify its validity. Within the marketplace, the regulator is trusted with the insertion of the GO, the certification of the prosumer-owner RES, and the energy providers are trusted with the report of the energy production. The SC guarantees that the implemented rules are followed for all the other operations. In this case, the challenge is to ensure that the SC implementation matches the legislation. Another limitation appears when designing a system respectful of the privacy of the actors. In this case, the complete state of the system can no longer be publicly stored and shared with all the actors. Instead, it needs to be split, and different parts are then stored in different PGs depending on which actor needs to access the data. Beyond weakening the tamper resistance guarantees, this also introduces additional complexity in the design and implementation of the SC, making it more challenging to ensure that the implementation correctly matches the legislation.

9.3.2 Marketplace Data Structure

Each record in HB is saved as *<key, value>* pairs. *Key* is a unique data identifier and must not repeat within a ledger. *Value* contains data associated with a specific key and all fields that the data record consists of. An underlying data structure is required to manipulate data in trade settlement transactions.

Table 9.2: *Prosumer Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Prosumer's record unique identifier
Electricity	*Double*	Amount of generated electricity *(kWh)*
WalletID	*String*	Prosumer's Wallet identifier
RESID	*String*	Prosumer's RES identifier

The *prosumer* record is described in Table 9.2. It is private for the PG which includes the energy provider and regulator. This record contains prosumer unique *ID*. The ID represents the *key* in *<key, value>* pair and contains a user blockchain identity *Address*. The *Electricity* field is updated by the energy provider and regulator based on the data from the prosumer's metering device. Further, it contains an associated wallet and RES IDs. The prosumer record intentionally does not contain any personally identifiable information (PII) to comply

with General Data Protection Regulation (GDPR) [30]. All PII needed for legal purposes can be saved in the conventional DB outside of the blockchain.

Table 9.3: *Wallet Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Wallet's unique identifier
Currency	*Double*	Amount of fiat currency, *e. g.*, USD, EUR
Electricity	*Double*	Amount of prosumer bought electricity *(kWh)*

The *wallet* record is described in Table 9.3. The *Currency* is the amount of fiat currency the prosumer has. It is used for trade settlement execution. The *Electricity* shows the amount of bought electricity. The wallet record *Electricity* and the prosumer record *Electricity* are separated to ensure that the bought electricity is not resold twice. The wallet record is visible to all energy providers to conduct cross-provider trade settlements.

Table 9.4: *GO Blockchain Data Record*

Field Name	Type	Description
ID	*String*	GO unique identifier
OwnerID	*String*	GO owner ID
RegulatorID	*String*	Issuer of GO
ElectricityAmount	*Double*	Amount of electricity *(kWh)*
IsConsumed	*Boolean*	Set *True* when electricity is sold

The GO record is described in Table 9.4. It is a significant asset that must be presented by the prosumer-seller during the trade settlement execution. The GO records are public for the entire blockchain network. Further, the GO record contains the respective ids of the prosumer who owns it and the regulator who issued it. Further, *ElectricityAmount* contains the amount of electricity certified by the regulator for further trading. Finally, when the energy is sold, the *isConsumed* field is set *True*.

Table 9.5: *Order Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Order unique identifier
Type	*String*	Order Type *(Sell or Buy)*
Price	*Double*	Price for the entire amount sold
ElectricityAmount	*Double*	Amount of electricity *(kWh)*
GOID	*String*	GO unique identifier
SellerWalletID	*String*	Seller wallet identifier
BuyerWalletID	*String*	Buyer wallet identifier

The *order* record is described in Table 9.5. *Type* shows what kind of order it is, *i. e.*, sell or buy. Further, the *Price* and *ElectricityAmount* contain the respective amounts of resources required from both parties. The *GOID* links a particular GO to the order. In *buy* order, the *GOID* is left empty to be filled by the seller. The *SellerWalletID* and *BuyerWalletID*

Algorithm 11 Fulfill Buy Electricity Order

```
1: function SellElectricity(Order (Type = Buy), GO, Seller Wallet, Buyer Wallet,
   SellerID)
2:     if GO.OwnerID == SellerId & GO.IsConsumed == False then
3:         if GO.ElectricityAmount == Order.ElectricityAmount then
4:             if BuyerWallet.Currency ≥ Order.Price then
5:                 Order.SellerWalletID ← SellerWallet.ID
6:                 Order.GOID ← GO.ID
7:                 SellerWallet.Currency ← SellerWallet.Currency + Order.Price
8:                 BuyerWallet.Currency ← BuyerWallet.Currency − Order.Price
9:                 BuyerWallet.Electricity ← BuyerWallet.Electricity +
                   Order.ElectricityAmount
10:                Commit
11:            else
12:                return Insufficient Buyer Currency.
13:        else
14:            return Insufficient Electricity Amount.
15:    else
16:        return Invalid GO Attached to the Order.
   Execute FinalizeOrder(Order, GO)                    ▷ c. f. , Algorithm 13.
```

Algorithm 12 Fulfill Sell Electricity Order

```
1: function BuyElectricity(Order (Type = Sell), GO, BuyerWallet, SellerWallet)
2:     if BuyerWallet.Currency ≥ Order.Price then
3:         if GO.IsConsumed == False then
4:             Order.BuyerWalletID ← BuyerWallet.ID
5:             SellerWallet.Currency ← SellerWallet.Currency + Order.Price
6:             BuyerWallet.Currency ← BuyerWallet.Currency − Order.Price
7:             BuyerWallet.Electricity ← BuyerWallet.Electricity + Order.ElectricityAmount
8:             Commit
9:         else
10:            return Invalid GO Attached to the Order.
11:    else
12:        return Insufficient Buyer Currency.
   Execute FinalizeOrder(Order, GO)                    ▷ c. f. , Algorithm 13.
```

fields contain identifiers of prosumer wallets. Depending on the type of the order, when it is created, one of the wallet identifiers is left empty, *i. e. , SellerWalletID* is empty for a buy order, and *BuyerWalletID* is empty for a sell order. When the order is fulfilled, it is private for prosumers and energy providers participating in trade settlement.

9.3.3 Trade Settlement Smart Contract

The implemented SC contains the necessary operations actors require to operate the marketplace and trade electricity. These operations include electricity registration, GO issue and consumption, order creation, and trade settlement. For the purposes of the performance evaluation, this study describes in detail trade settlement SC functions that fulfill the *buy*

Algorithm 13 Finalize Order

```
1: function FinalizeOrder(Order, GO)
2:     GO.IsConsumed ← True
3:     Delete(Order)
4:     Commit
```

and *sell* customer electricity orders, *c. f. ,* Algorithm 11 and 12. Before the order can be fulfilled, a number of prerequisites have to be met. First, electricity has to be generated and registered within the prosumer's marketplace account. Further, a GO has to be issued for the amount of electricity that is being sold. Finally, the order itself has to be created. For both order types, the trade settlement operation execution has two stages. This is required due to HB SC's inability to modify private, *i. e. ,* wallets and order, and public, *i. e. ,* GO, data in a single transaction.

In the first stage of Algorithm 11, the *SellElectricity* function takes a *buy* order posted by a prosumer-buyer. Further, the algorithm performs a number of security checks. Since the prosumer-seller provides the GO at the moment of trade settlement execution, it is verified to have the correct ownership. Further, the GO's *IsConsumed* field is checked to be *False*. Finally, the GO is checked to be issued for the amount of electricity listed in the buy order. Next, the buyer's wallet is verified to have an appropriate currency to buy the electricity. Finally, the resources are exchanged between the buyer and seller, *i. e. ,* electricity and currency. This transaction is private for PG, which includes trading prosumers' energy providers and the regulator.

In the first stage of Algorithm 12, the *BuyElectricity* function takes a *sell* order posted by a prosumer-seller. Further, the algorithm verifies if the GO is consumed and if the buyer has enough currency in the wallet. Finally, the resources are exchanged between the buyer and seller. The correct ownership of the GO and conformity of GO's and order's electricity amounts is not checked in *BuyElectricity* function. These checks are performed during sell order creation, *i. e. ,* GO has to be provided during the sell order creation.

In the second stage of both Algorithm 11 and 12, the *FinalizeOrder* function is executed by the buyer's energy provider, *i. e. ,* the actor interested in preventing electricity double-spending. First, this function takes the GO, sets its *IsConsumed* value to *True*, and saves it in the public ledger. Further, it marks the fulfilled order as deleted. Thus, the order is not visible in the order chart but can be found in the ledger history.

9.4 Performance Evaluation

The throughput of public transactions, *i. e. ,* visible to the entire private network, has already been investigated by the authors of [31]. The main aim of this study is to measure the performance of private transaction execution with the SC tailored to the energy marketplace needs. The performance evaluation was conducted on the test infrastructure described in Figure 9.4. The infrastructure consists of 4 virtual machines (VMs), where each VM size is 16 vCPUs, 64 GB RAM, and 256 GB high throughput (150MB/s) disk space. Energy providers A, B, and C run VM1, VM2, and VM3, respectively, while the regulator runs VM4. All

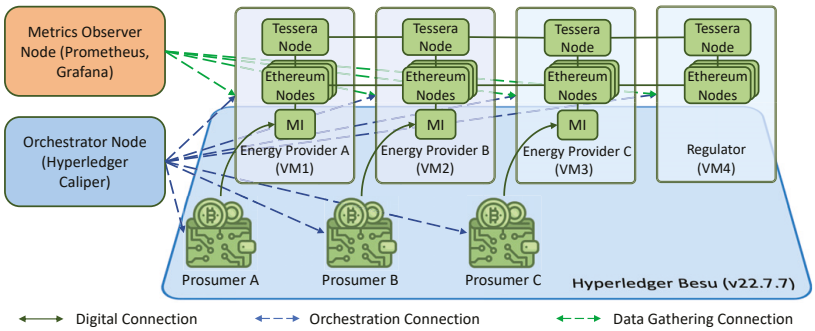


Figure 9.4: *Implemented Energy Marketplace.*

VMs are connected with a 10Gbit/s network interface. In our experimental implementation, we use HB version 22.7.7 and Tessera 22.1.7 without modifying the core code. All nodes within the infrastructure are deployed as docker containers. To collect reliable and correct performance evaluation data, *Prometheus*[5], *Grafana*[6], and *Hyperledger Caliper*[7] (HC) tools are utilized. The *Prometheus* is used as the main blockchain operation data collector. The *Grafana* is used as a data visualization tool. The *HC* performance evaluation tool is used as a transaction load generator. In Figure 9.4, the *HC* is part of the Orchestrator Node and is located on a separate VM. The *HC* executes transactions bypassing the MI. Thus, the MI is not a part of the performance evaluation.

Several performance metrics are considered in this study. First, the *throughput* is the number of successful transactions (TPS) or reads (RPS) executed per second in the blockchain network. The *latency* is the time it takes to finalize transaction execution and write it to the ledger or return a reply with the query result. The *scalability* is the behavior of the network with an increasing number of nodes. Scalability is also dependent on the size of PG.

Table 9.6: *Performance Evaluation Parameters*

Parameter	Value
Transaction Send Rate (Write)	10, 20 → 300 with step of 20 *(fixed-rate in duration of 5 minutes)
Block Period Seconds (BPS)	1 → 6 with step of 1
Transaction Send Rate (Read)	100, 300 → 3000 with step of 300 *(fixed-rate in duration of 5 minutes)
Validator Nodes	4 → 24 with step of 4
Privacy Group Size	2, 3, 4
Consensus Mechanism	Clique, IBFT 2.0, QBFT

This study manipulated several configuration parameters within HB to investigate the maximum throughput. These parameters were selected based on the performance tests conducted by the HB developers and research studies [31]. The *Block Period Seconds* (BPS) parameter defines the time validators accept transactions to add to the new block. When the

[5] https://prometheus.io/docs/introduction/overview/
[6] https://grafana.com/
[7] https://hyperledger.github.io/caliper/

BPS time is up, the block is cut and embedded into the ledger. Further, horizontal scalability is investigated by changing the number of validator nodes and PG size. To investigate write transaction throughput, 5-minute tests were executed with a constant send rate. To investigate read throughput, the 4KB asset was read from the local HB database, *i. e.,* state database, with varying query send rates. Table 9.6 summarizes the entire performance evaluation parameters configuration.

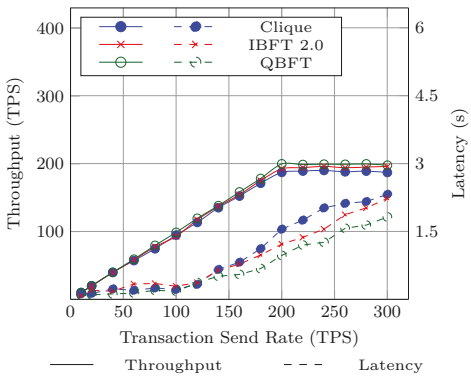
9.4.1 Write - Trade Settlement Execution

In this study, an Algorithm 11 was executed as an SC function to test maximum write TPS. An Algorithm 11 was chosen as a load generator due to having the highest computational complexity out of all defined SC functions. To write a transaction to the ledger, a respective consensus mechanism, *i. e.,* Clique, IBFT 2.0, or QBFT, must be executed. First, we test the baseline HB configuration, which included the minimum necessary setup to operate, *i. e.,* four validators, BPS = 1s. The PG size is 3, *i. e.,* energy providers A and B, and the regulator. The throughput measurement results are shown in Figure 9.5a. All consensus mechanisms show a similar performance of approximately 200 TPS. However, QBFT demonstrated the best latency. Clique and IBFT 2.0 demonstrate higher latency both for peak throughput and further increase of send rate exceeding the maximum TPS. The baseline test demonstrates the maximum sustainable network load with private transactions of around 200 TPS. Thus, further tests are conducted with a fixed send rate of 200 TPS.

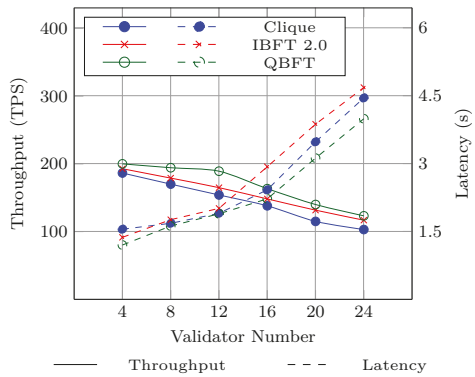
Next, the maximum TPS with a varying BPS was investigated, *c. f.,* Figure 9.5d. The results demonstrate that the BPS affects the maximum throughput of the HB network, *i. e.,* the BPS increase results in a steady throughput decrease. Further, the latency rises significantly, *e. g.,* up to approximately 6s latency for BPS = 6s. Here all investigated consensus mechanisms show similar performance under varying BPS, where QBFT is the best performer. The results demonstrate that BPS and eventual latency increase significantly affecting the Quality of Service (QoS) [32] provided by the marketplace. QoS is aimed to maximize the user experience in terms of response time and transaction success rate by addressing throughput and scalability issues. In this case, the BPS has to be considered an important metric for QoS provisioning [33].

The horizontal scalability was investigated with varying validators number and a PG size. The results of the validator scalability investigation are shown in Figure 9.5b. Here, the number of validator nodes was changed from 4 to 24 with a step of 4. Results demonstrate that the number of validator nodes significantly affects the maximum network throughput. It represents a significant performance bottleneck resulting in approximately 42% throughput reduction with 24 validators. Further, the latency increases significantly, reaching approximately 4.5s for IBFT 2.0. Here, all investigated consensus mechanisms demonstrate similar performance, with QBFT having the highest TPS and the lowest latency. In addition, QBFT demonstrates the best scalability by maintaining 190-200 TPS up to 12 validators.

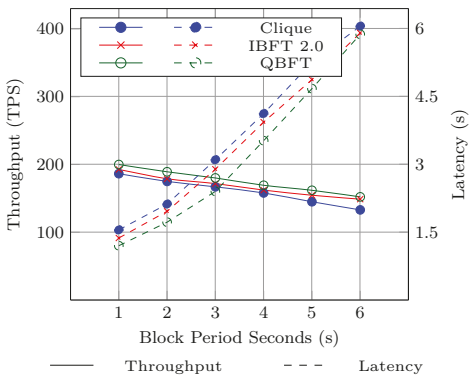
The results of PG size scalability are shown in Figure 9.5c. The investigated PG sizes are under four nodes because each BO can operate only one Tessera node, *i. e.,* this is an infrastructure limitation. The PG size increase does not result in a significant throughput decrease. However, the latency increases approximately by a half second for all investigated consensus mechanisms with a PG size equal to four. Here, the performance of consensus



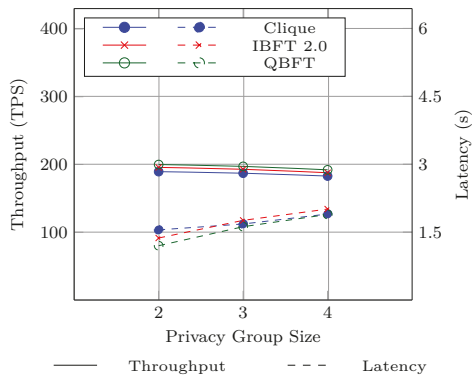
(a) *Transaction Throughput and Latency with varying send rate (*BPS = 1s, Validators = 4, PG size = 3*).*



(b) *Throughput and Latency with varying* Validator Nodes *number (*BPS = 1s, 200 TPS send rate*).*



(c) *Throughput and Latency with varying* PG Size *(*BPS = 1s, Validators = 4, 200 TPS send rate*).*



(d) *Throughput and Latency with varying* Block Period Seconds *(*200 TPS send rate*).*

Figure 9.5: *Hyperledger Besu performance and scalability evaluation results.*

mechanisms is similar, with QBFT showing the best results.

The performance evaluation results demonstrate that the maximum possible throughput depends significantly on BPS and network size, *i. e.*, the best throughput is achieved with BPS = 1s and 4 Validators configuration. Further, the QBFT has the best throughput, latency, and scalability characteristics out of all investigated consensus mechanisms. Finally, the performance evaluation shows that the HB-based marketplace demonstrates an approximately two times lower throughput and higher latency than the HF-based marketplace investigated in [34]. However, HF uses the RAFT consensus mechanism, which is only CFT, *i. e.*, does not protect from malicious nodes. In contrast, HB's IBFT 2.0 and QBFT consensus mechanisms are BFT, *i. e.*, protect the blockchain network for up to 1/3 of malicious nodes at the cost of increased computational complexity.

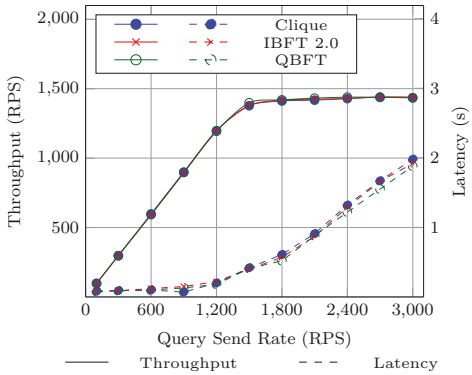


Figure 9.6: *Read Throughput and Latency (*4KB asset*).*

9.4.2 Ledger Data Read

In order to write any data to the blockchain ledger, a consensus mechanism has to be executed. The query, *i. e.*, read, request does not execute a consensus mechanism to get the requested data. Thus, the block or network configuration, *i. e.*, validator number, does not affect the read throughput. Here, it is the amount of data, *i. e.*, *asset size*, that is read from the individual blockchain node that affects reads per second (RDS). The read throughput is shown in Figure 9.6. To investigate read throughput, the query was constructed to read 4KB of data from the RockDB world state database. The results demonstrate the maximum throughput of approximately 1440 RPS for all investigated consensus mechanisms. The latency remains under 500ms until the moment we reach peak read throughput. From there on the latency starts to increase rapidly if the read queries send rate is over peak throughput.

The read throughput evaluation results show that the HB-based energy marketplace demonstrates a similar throughput to the HF-based marketplace investigated in [34]. This is an indicator that world state databases that are used in HB and HF, *i. e.*, RockDB and LevelDB respectively, demonstrate similar performance when it comes to reading assets of similar size.

9.5 Results and Observations

During the design, implementation, and performance evaluation of the proposed marketplace, a number of observations and conclusions were made, which bring an enhanced understanding of the advantages and limitations of HB. Such observations and conclusions are discussed next.

9.5.1 Limitations of Private Transaction Execution

System transaction throughput is an important performance characteristic that must be considered in the system design phase. If throughput requirements are not met, the production system QoS and scalability requirements will be impossible to meet. Different blockchain

architectures, *i. e. ,* public and private, demonstrate a considerable performance difference in terms of throughput and latency. Higher system decentralization and security come at a cost of additional computational complexity. Performance evaluation of the HB-based energy marketplace demonstrates a throughput of approximately 200 TPS which is a considerable improvement in comparison with public blockchains such as Bitcoin or Ethereum [21]. However, the HB still requires an improvement in throughput and scalability to get to the level of performance demonstrated by HF.

The measured HB throughput of 200 TPS can be used to estimate the maximum number of prosumers the energy marketplace can support [34]. However, this estimation is a first-order approximation and may not account for other factors that can affect the maximum number of prosumers in the real-world system. Several metrics were identified to make this estimation, including the maximum throughput of the system T_{max}, the number of daily electricity generation registrations m, the number of orders created by a prosumer per day o, the number of trade transactions executed per day t, and the number of GOs issued for a prosumer per day g. Metric t was identified as a full number of blockchain transactions needed to finalize energy trade. As trade settlement has two stages, it's every execution requires two blockchain transactions. Dividing the maximum daily amount of transactions by the sum of *m, o, t, g* yields an approximate number of prosumers that can operate within the marketplace, *c. f. ,* Eq.(9.1).

$$Prosumers_{max} = \frac{T_{max} * 24 * 60 * 60}{m + o + t + g} \tag{9.1}$$

The throughput of 200 TPS implies that the energy marketplace can execute 17,280,000 transactions in 24 hours. The configuration of the *m, o, t, g* determines the maximum number of prosumers the marketplace can support. The m needs to be set first because it affects the rest of the parameters. If the m parameter is set to 24, which means hourly registrations, the prosumer can trade their generated electricity 23 times daily (minus 1 hour for initial generation). This worst-case scenario assumes the prosumer trades every time the metering device updates. To execute a trade, the prosumer must place an order and have a GO issued. In this scenario, the total number of daily prosumer transactions equals 116 (24+23+23*2+23). Therefore, the maximum number of prosumers supported by the marketplace is 17.280.000 / 116 $\approx$ 150.000, corresponding to a small-to-medium-size energy community.

9.5.2 Limited Auditability of Private Transactions

The auditability and integrity of all data in the decentralized network are characteristics that affect the guarantees that the system can provide for its users. However, private transactions imply that only a portion of blockchain network participants see the contents and participate in consensus execution for a particular transaction. As the Ethereum blockchain was not designed to work with private transactions, the *Tessera* private transaction manager was adopted in HB. It is built as a separate entity and complements the implementation of the Ethereum Enterprise Client. However, it comes at the cost of private transactions' auditability. The Tessera nodes distribute private transactions to the members of PG. However, the rest

of the nodes outside of PG receive the record confirming that the private transaction was executed. Such an approach results in a limitation where the blockchain network participants outside of PG cannot verify the validity of the private transaction data. This is a result of the inability of non-PG members of the HB network to verify the correctness of private SC deployment and transaction execution.

9.5.3 Public and Private Data Modification

The integrity of data within a marketplace relies heavily on the correctly defined SC. In the case of private transaction execution, the data within the blockchain is split into public, *i. e.,* seen by all network participants, and private, *i. e.,* available only to the members of PG. However, in a function such as an energy trade settlement, we need to modify both public and private data within one operation. The HF allows such modifications within one transaction without the exposure of private data to non-PG members. In contrast, the HB does not have such a capability, and modification of public and private data has to be split into two different transactions. Such a limitation opens up additional security concerns where the delay between private and public data modification transactions may be used to disrupt trade process execution or attempt double spending of GOs.

9.5.4 Private Blockchain Lesser Energy Consumption

The excessive amount of computations needed for the transaction execution within a blockchain network raised concern amount environmentalists regarding subsequent carbon emissions [35]. Blockchain implementations such as Bitcoin with PoW consensus algorithm require the execution of computationally heavy tasks, which in the long-term perspective may lead to the carbonization of Earth's atmosphere and cause harmful effects on humankind. Bitcoin is representative of public blockchain architecture, which typically involves a vast number of computing machines involved in blockchain operation and transaction generation. Such public blockchains enable decentralized environments that provide digital sovereignty to their users [36]. The HB, as well as HF, are representatives of private blockchain architecture, which is typically deployed for a targeted business use case that involves selected actors. Consequently, private blockchains assume a certain degree of centralization within a blockchain system, which requires collaborating entities to have certain legal agreements outside of blockchain guarantees, *i. e.,* in the case of the energy marketplace, it is the reliance on TTP such as the regulator. However, private blockchain deployments with PoA consensus mechanisms consume a small fraction of computations when compared to Bitcoin's PoW. Thus, private blockchains are more sustainable in a long-term approach [37].

9.6 Related Work

Hyperledger Foundation has created several projects which employ different blockchain architectures, *i. e.,* public and private, to address industrial and business use-cases [38]. Thus, private blockchains like HF and HB became the main energy marketplace implementation and investigation tools. Recently, there were a number of proposals on blockchain-based

energy marketplaces in terms of system architecture, electricity trading framework, and performance evaluation. Such proposals are discussed next.

In [39], the authors propose an HB-based P2P marketplace for energy trading and payment settlement. The marketplace utilizes HB as a blockchain platform and IBFT 2.0 as a consensus mechanism. Further, the authors compare IBFT 2.0 with Clique, PoW, and HF's RAFT. According to the authors, their marketplace demonstrates better throughput and latency than PoW and Ethereum Clique. Further, the authors claim that the proposed unified energy trading model provides lower latency compared to similar systems based on PoW, Clique, and HF's RAFT. The authors of [40] propose an HF-based P2P energy marketplace for tokenized energy assets. Such assets are traded within the marketplace, where each actor can benefit monetarily depending on its role. Further, the authors define actors and requirements for the P2P energy marketplace. However, their marketplace does not include a regulator role, GO usage, and data privacy requirements intrinsic to energy market systems. The authors claim their implementation achieved a throughput of 448.3 TPS with transactions that modify public data. However, the authors do not consider private transaction execution and PG throughput impact. In [41], the authors propose an automated blockchain-based P2P energy marketplace based on a multi-agent system paradigm. Permissioned blockchain allows for reduced transaction costs, enables marketplace micro-transactions, and eliminates a single point of failure. According to the authors, blockchain technology enables prosumer self-sovereignty while allowing the marketplace to comply with current data regulations. The authors of [42] propose an HB-based framework for P2P energy trading. The proposed marketplace uses a flexible permission ascription scheme that utilizes HB permissioning and IBFT 2.0 consensus mechanism. According to the authors, the proposed framework provides an efficient scheme for P2P energy trading compared to other solutions. The authors claim that IBFT 2.0 has five times lower latency than Ethereum PoW and two times lower than HF RAFT and KAFKA. Further, performance evaluation demonstrated that IBFT 2.0 has 1.5 times higher throughput than HF's RAFT and Kafka and three times higher than Ethereum PoW. In [43], the authors propose an HF-based platform for the transactive energy marketplace. A proposed platform has a layered architecture consisting of physical, communications, and blockchain layers. Further, the authors use the energy generation data from a real-world energy provider and build its digital twin as a physical layer for their platform. The authors claim that the developed prototype allows trading electricity via SCs developed within the HF network. The authors of [44] propose a blockchain-based marketplace platform that enables energy trading between institutions and electric vehicle (EV) owners. Within the case study, institutions own RES and sell generated electricity to EV owners via a P2P trade contract. The authors claim that such a marketplace platform enables synergy between institutions and EV owners, providing clean and affordable energy. For further reading on the developments in blockchain-based energy marketplaces, the reader is referred to [45].

In [31], the authors conduct an in-depth performance evaluation of the HB platform and its three main consensus mechanisms for private blockchain, *i. e.,* Clique, IBFT 2.0, and QBFT. According to the authors, the performance of HB has a number of bottlenecks, such as transaction execution and blockchain state updates, which are influenced by node computation power and transaction complexity. Authors claim that QBFT consensus has

the best performance and scalability results, achieving a write throughput of approximately 450 TPS and scalability of up to 14 validator nodes. The authors of [46] compare the main proof-based consensus mechanisms, focusing on security and performance. The authors highlight the centralization tendency and the vulnerabilities of main proof-based consensus mechanisms, *i. e.,* PoW, PoS, PoA, and Delegated PoS (DPoS). According to the authors, DPoS consensus has the best balance between throughput, latency, and scalability. However, such a balance comes at the cost of increased centralization and reduced protection against malicious activity.

The related work demonstrate that the application of blockchain technology in the context of an energy marketplace has been defined at a level of abstract entities and operations. However, all works mentioned above lack requirements definition and alignment with the existing regulation on P2P energy trading. Further, the related work lack implementation details and discussion on the technical limitations of blockchain technology incorporation. This work discusses the energy marketplace from regulatory and technical perspectives to provide insights into challenges encountered during the implementation of private transactions execution and system operation.

9.7 Summary and Outlook

This work proposes a decentralized blockchain-based P2P energy marketplace that addresses actors' privacy and the performance of consensus mechanisms. The main aim of the marketplace is to automate the P2P trade settlement process while preserving actors' privacy. The novelty of the proposed marketplace is its alignment with the current energy trade regulations defined in D2018/2001 of the European Parliament. More specifically, our marketplace incorporates the *regulator* actor. The regulator represents a governmental authority that controls renewable energy trading via GO issue and price regulation. In addition, the regulator certifies the RES used to generate traded electricity. Hence, with current regulations, the marketplace is partially centralized around the regulator actor but still improves the automation of energy trading.

Performance evaluation results of an HB-based marketplace private transaction execution with three main consensus mechanisms, *i. e.,* Clique, IBFT 2.0, and QBFT, demonstrate a throughput of approximately 200 TPS with baseline configuration. The QBFT consensus mechanism shows the best throughput and latency. Further, QBFT demonstrates the best scalability by maintaining 190-200 TPS throughput for up to 12 validators. However, HB's QBFT consensus mechanism demonstrates lower throughput than another popular private permissioned blockchain platform HF. This is a side effect of BFT and, thus, increased computations of QBFT. In contrast, HF executes the RAFT consensus mechanism, which is CFT, *i. e.,* more centralized and vulnerable to collusion between malicious nodes. However, the inherent centralization around the regulator mitigates this issue, making HF better suited for such a use case.

Future work will focus on investigating possible improvements for consensus mechanisms for blockchain-based marketplaces to improve the efficiency of their operation. In addition, investigating a trade-off between the performance and security of private blockchains is of interest.

Acknowledgment

The work was partly sponsored by the Swedish Knowledge Foundation through the project *Symphony - Supply-and-Demand-based Service Exposure using Robust Distributed Concepts*. The project partners in Symphony are Ericsson AB (Stockholm, Sweden) and Affärsverken Energi AB (Karlskrona, Sweden).

References

[1] Y. Yang, S. Zhang, and Y. Xiao. "Optimal design of distributed energy resource systems coupled with energy distribution networks". In: *Energy* 85 (2015), pp. 433–448. DOI: 10.1016/j.energy.2015.03.101.

[2] B. Jasim and P. Taheri. "An Origami-Based Portable Solar Panel System". In: *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. 2018, pp. 199–203. DOI: 10.1109/IEMCON.2018.8614997.

[3] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. "Blockchain Based Decentralized Management of Demand Response Programs in Smart Energy Grids". In: *Sensors* 18.2 (2018), pp. 1–21. DOI: 10.3390/s18010162.

[4] EU Parliament. *Directives Directive (EU) 2018/2001 of the European Parliament*. 2022, pp. 82–209. URL: http://data.europa.eu/eli/dir/2018/2001/2022-06-07 (visited on 06/18/2023).

[5] Á. Hamburger. "Is guarantee of origin really an effective energy policy tool in Europe? A critical approach". In: *Society and Economy* 41 (2019), pp. 487–507. DOI: 10.1556/204.2019.41.4.6.

[6] B. Hertz-Shargel, D. Livingston, and A. C. of the United States. *Assessing Blockchain's future in transactive energy*. 2019. ISBN: 9781619775992. URL: https://www.atlanticcouncil.org/in-depth-research-reports/report/assessing-blockchains-future-in-transactive-energy/ (visited on 06/18/2023).

[7] T. Kollmann, S. Hensellek, K. de Cruppe, and A. Sirges. "Toward a renaissance of cooperatives fostered by Blockchain on electronic marketplaces: a theory-driven case study approach". In: *Electronic Markets* 30.2 (2020), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

[8] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. URL: https://bitcoin.org/bitcoin.pdf (visited on 06/18/2023).

[9] J. Singh and J. D. Michels. "Blockchain as a Service (BaaS): Providers and Trust". In: *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 2018, pp. 67–74. DOI: 10.1109/EuroSPW.2018.00015.

[10] H. Foundation. *Hyperldger Besu Ethereum client*. 2022. URL: https://besu.hyperledger.org/en/stable/ (visited on 06/20/2023).

[11] R.-V. Tkachuk, D. Ilie, R. Robert, V. Kebande, and K. Tutschku. “On the Performance of Consensus Mechanisms in Privacy-Enabled Decentralized Peer-to-Peer Renewable Energy Marketplace”. In: *26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*. IEEE, 2023, pp. 179–186. DOI: 10.1109/ICIN56760.2023.10073510.

[12] H. Akay and S.-G. Kim. “Reading functional requirements using machine learning-based language processing”. In: *CIRP Annals* 70 (1 Jan. 2021), pp. 139–142. DOI: 10.1016/j.cirp.2021.04.021.

[13] K. B. Wilson, A. Karg, and H. Ghaderi. “Prospecting non-fungible tokens in the digital economy: Stakeholders and ecosystem, risk and opportunity”. In: *Business Horizons* 65 (5 Sept. 2022), pp. 657–670. DOI: 10.1016/j.bushor.2021.10.007.

[14] Q. Wang, R. Li, Q. Wang, and S. Chen. *Non-Fungible Token (NFT): Overview, Evaluation, Opportunities and Challenges*. 2021. URL: http://arxiv.org/abs/2105.07447.

[15] K. Khatter and DevanjaliRelan. “Non-functional requirements for blockchain enabled medical supply chain”. In: *International Journal of System Assurance Engineering and Management* 13 (3 June 2022), pp. 1219–1231. DOI: 10.1007/s13198-021-01418-y.

[16] A. Qazi, F. Hussain, N. A. Rahim, G. Hardaker, D. Alghazzawi, K. Shaban, and K. Haruna. “Towards Sustainable Energy: A Systematic Review of Renewable Energy Sources, Technologies, and Public Opinions”. In: *IEEE Access* 7 (2019), pp. 63837–63851. DOI: 10.1109/ACCESS.2019.2906402.

[17] M. Binjubeir, A. A. Ahmed, M. A. B. Ismail, A. S. Sadiq, and M. K. Khan. “Comprehensive Survey on Big Data Privacy Protection”. In: *IEEE Access* 8 (2020), pp. 20067–20079. DOI: 10.1109/ACCESS.2019.2962368.

[18] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and Y. Liu. “A Survey of Blockchain Technology Applied to Smart Cities: Research Issues and Challenges”. In: *IEEE Communications Surveys and Tutorials* 21.3 (2019), pp. 2794–2830. DOI: 10.1109/COMST.2019.2899617.

[19] R.-V. Tkachuk, D. Ilie, K. Tutschku, and R. Robert. “A Survey on Blockchain-Based Telecommunication Services Marketplaces”. In: *IEEE Transactions on Network and Service Management* 19.1 (2022), pp. 228–255. DOI: 10.1109/TNSM.2021.3123680.

[20] M. Liu, K. Wu, and J. J. Xu. “How Will Blockchain Technology Impact Auditing and Accounting: Permissionless versus Permissioned Blockchain”. In: *Current Issues in Auditing* 13.2 (2019). DOI: 10.2308/ciia-52540.

[21] G. Wood. *Ethereum: a secure decentralised generalised transaction ledger*. Ethereum Project Yellow Paper, 2014, pp. 1–32. URL: https://gavwood.com/paper.pdf (visited on 06/18/2023).

[22] M. Kaleem et al. “Vyper: A Security Comparison with Solidity Based on Common Vulnerabilities”. In: *BRAINS*. 2020, pp. 107–111. DOI: 10.1109/BRAINS49436.2020.9223278.

[23] M. Vukolić. “The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication”. In: *Lecture Notes in Computer Science* 9591 (2016). Ed. by J. Camenisch and D. Kesdoğan, pp. 112–125. DOI: 10.1007/978-3-319-39028-4_9.

[24] P. Szilágyi. *EIP-225: Clique proof-of-authority consensus protocol*. 2017. URL: https://eips.ethereum.org/EIPS/eip-225 (visited on 04/20/2023).

[25] R. Saltini et al. “IBFT 2.0: A Safe and Live Variation of the IBFT Blockchain Consensus Protocol for Eventually Synchronous Networks”. In: (2019).

[26] M. Castro and B. Liskov. “Practical Byzantine Fault Tolerance”. In: *Proceedings of the Third Symposium on Operating Systems Design and Implementation*. OSDI ’99. New Orleans, Louisiana, USA: USENIX Association, 1999, pp. 173–186. ISBN: 1880446391.

[27] Y.-T. Lin. *Istanbul Byzantine Fault Tolerance*. 2017. URL: https://github.com/ethereum/EIPs/issues/650 (visited on 04/20/2023).

[28] R. Saltini et al. “Correctness Analysis of IBFT”. In: (2019). URL: http://arxiv.org/abs/1901.07160.

[29] H. Moniz. “The Istanbul BFT Consensus Algorithm”. In: (2020). URL: http://arxiv.org/abs/2002.03613.

[30] EU Parliament. *Regulation (EU) 2016/679 of the European Parliament (General Data Protection Regulation)*. 2016, pp. 1–99. URL: https://gdpr-info.eu/ (visited on 06/18/2023).

[31] C. Fan, C. Lin, H. Khazaei, and P. Musilek. “Performance Analysis of Hyperledger Besu in Private Blockchain”. In: *2022 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*. 2022, pp. 64–73. DOI: 10.1109/DAPPS55202.2022.00016.

[32] W. Chen and I. Paik. “Toward Better Quality of Service Composition Based on a Global Social Service Network”. In: *IEEE Transactions on Parallel and Distributed Systems* 26.5 (2015), pp. 1466–1476. DOI: 10.1109/TPDS.2014.2320748.

[33] A. Vaghani, K. Sood, and S. Yu. “Security and QoS issues in blockchain enabled next-generation smart logistic networks: A tutorial”. In: *Blockchain: Research and Applications* 3.3 (2022), p. 100082. ISSN: 2096-7209. DOI: https://doi.org/10.1016/j.bcra.2022.100082. URL: https://www.sciencedirect.com/science/article/pii/S2096720922000239.

[34] R.-V. Tkachuk, D. Ilie, R. Robert, V. Kebande, and K. Tutschku. “Towards Efficient Privacy and Trust in Decentralized Blockchain-Based Peer-to-Peer Renewable Energy Marketplace”. In: *Sustainable Energy, Grids and Networks* (2023), pp. 1–27. DOI: 10.1016/j.segan.2023.101146.

[35] J. Sedlmeir, H. U. Buhl, G. Fridgen, and R. Keller. “Recent Developments in Blockchain Technology and their Impact on Energy Consumption”. In: *Informatik Spektrum* 43 (6 Feb. 2021), pp. 391–404. DOI: 10.1007/s00287-020-01321-z.

[36] Morrow and Zarrebini. “Blockchain and the Tokenization of the Individual: Societal Implications”. In: *Future Internet* 11 (10 Oct. 2019), pp. 1–12. DOI: 10.3390/fi11100220.

[37] A. O. Bada, A. Damianou, C. M. Angelopoulos, and V. Katos. "Towards a Green Blockchain: A Review of Consensus Mechanisms and their Energy Consumption". In: *2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, July 2021, pp. 503–511. DOI: 10.1109/DCOSS52077.2021.00083.

[38] J. Li, A. Grintsvayg, J. Kauffman, and C. Fleming. "LBRY: A Blockchain-Based Decentralized Digital Content Marketplace". In: *2020 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*. Aug. 2020, pp. 42–51.

[39] J. Abdella, Z. Tari, A. Anwar, A. Mahmood, and F. Han. "An Architecture and Performance Evaluation of Blockchain-Based Peer-to-Peer Energy Trading". In: *IEEE Transactions on Smart Grid* 12 (4 2021), pp. 3364–3378. DOI: 10.1109/TSG.2021.3056147.

[40] N. Karandikar, A. Chakravorty, and C. Rong. "Blockchain Based Transaction System with Fungible and Non-Fungible Tokens for a Community-Based Energy Infrastructure". In: *Sensors* 21 (11 May 2021), pp. 1–32. DOI: 10.3390/s21113822.

[41] Y. Mezquita, A. B. Gil-González, A. M. del Rey, J. Prieto, and J. M. Corchado. "Towards a Blockchain-Based Peer-to-Peer Energy Marketplace". In: *Energies* 15 (9 Apr. 2022), pp. 1–20. DOI: 10.3390/en15093046.

[42] N. R. Pradhan, A. P. Singh, N. Kumar, M. M. Hassan, and D. S. Roy. "A Flexible Permission Ascription (FPA)-Based Blockchain Framework for Peer-to-Peer Energy Trading With Performance Evaluation". In: *IEEE Transactions on Industrial Informatics* 18.4 (2022), pp. 2465–2475. DOI: 10.1109/TII.2021.3096832.

[43] A. Boumaiza, M. Z. C. Wanik, and A. Sanfilippo. "Modeling a Blockchain-enabled Transactive Energy System for Community Microgrids". In: *2022 IEEE 16th International Conference on Compatibility, Power Electronics, and Power Engineering (CPE-POWERENG)*. 2022, pp. 1–6. DOI: 10.1109/CPE-POWERENG54966.2022.9880874.

[44] I. Cavalcante, J. Júnior, J. A. Manzolli, L. Almeida, M. Pungo, C. P. Guzman, and H. Morais. "Electric Vehicles Charging Using Photovoltaic Energy Surplus: A Framework Based on Blockchain". In: *Energies* 16.6 (2023). ISSN: 1996-1073. DOI: 10.3390/en16062694.

[45] M. Choobineh, A. Arabnya, B. Sohrabi, A. Khodaei, and A. Paaso. "Blockchain technology in energy systems: A state-of-the-art review". In: *IET Blockchain* 3 (1 Mar. 2023), pp. 35–59. ISSN: 2634-1573. DOI: 10.1049/blc2.12020. URL: https://onlinelibrary.wiley.com/doi/10.1049/blc2.12020.

[46] G. A. F. Rebello et al. "A security and performance analysis of proof-based consensus protocols". In: *Annals of Telecommunications* 77 (7-8 Aug. 2022), pp. 517–537. ISSN: 19589395. DOI: 10.1007/s12243-021-00896-2.

Chapter Ten

On the Application of Enterprise Blockchains in Decentralized Renewable Energy Marketplaces

Abstract

The energy distribution infrastructure is a vital part of any modern society. Thus, renewable energy sources are becoming increasingly important as a substitute for energy produced with fossil fuels. However, renewable energy production faces several challenges in the energy market and its management, such as inflexible pricing models and inaccurate green consumption information. A decentralized electricity marketplace may address these challenges. However, such a platform must guarantee that the transactions follow the market rules and regulations, cannot be manipulated, and are consistent with the energy generated. One of the ways to provide these guarantees is to leverage blockchain technology. Our previous studies demonstrate the current energy trade regulations result in partial marketplace centralization around governmental authority. The governmental authority, *i. e.*, the regulator, oversees marketplace operations and requires energy providers to share private data about electricity generation and energy trade settlement. This study proposes amendments to D2018/2001 legislation and the governmental regulator actor to improve marketplace flexibility and data privacy. Further, we propose a new blockchain-based P2P energy marketplace model with increased flexibility and scalability while addressing actors' privacy and trust requirements. The marketplace utilizes a private permissioned blockchain Hyperledger Fabric (HF) due to its privacy-preserving and trust-enabling capabilities. This study provides HF comparison with Ethereum-based competitor Hyperledger Besu (HB). Further, based on the identified advantages and limitations, we discuss the rationale for the choice of HF. We utilize HF's smart contracts to enable P2P energy trade settlement orchestration and management. Based on previous studies, we propose an improvement towards HF security by utilizing a Byzantine Fault Tolerant (BFT) consensus mechanism, which is protected against malicious system actors. The results demonstrate that while protecting the blockchain network from malicious system actors, the BFT mechanism shows a similar throughput to the RAFT Crash Fault Tolerant consensus in the context of the P2P energy marketplace. Finally, BFT consensus enables legislation enhancements, resulting in increased flexibility and data privacy in the energy trade marketplace.

10.1 Introduction

Nowadays, the energy distribution infrastructure is a vital part of any modern society. It plays a crucial role in enabling the usage of devices and vehicles that are utilized by a large number of individuals. Also, the energy infrastructure is an integral part of the industry sector,

enabling day-to-day enterprise operations. Such dependency on the energy infrastructure requires making it robust and flexible, *i. e. ,* enabling efficient electricity supply and demand strategies. Nowadays, the reliance on big power plants with high electricity output creates centralized energy supply points, which, if fail, result in a disruption of social and economic activities of entire regions. In addition, utilization of large power plants results in *suboptimal energy distribution*, *i. e. ,* challenges in matching supply and demand changes with the power plant's capabilities [1]. Further, today's energy infrastructure consists of many power plants that use fossil fuels to produce electricity, resulting in the *atmosphere's carbonization*. The recent adoption of renewable energy sources (RES), *e. g. ,* wind, solar, and hydro [2], aims to reduce the carbon footprint of fossil fuel usage and eventually substitute it. Further, the widespread adoption of RES should improve energy production and distribution flexibility by having the energy sources closer to the end consumers. Finally, privately owned RESs enable the establishment of self-sufficient energy communities that do not have to rely on large power plants with centralized electricity distribution. However, the percentage of renewable energy production is still very insignificant, which requires the development of methods for the promotion and dissemination of RES.

The introduction of RES and developments in Smart Grid technologies [3], *i. e. ,* automation and orchestration of energy distribution, opened opportunities for individuals in terms of private RES ownership and operation. The individuals can install RES on their premises, *e. g. ,* homes or private spaces, and become electricity *prosumers*, *i. e. ,* producers/consumers. Here, the surplus of RES-produced electricity is supplied back to the energy grid, making the prosumer a distributed energy source (DER) [4]. This opens an opportunity for the prosumer to make a profit on the surplus electricity by selling it to their energy provider (EP) via a bilateral contract, *i. e. ,* immediate sell on a surplus production according to previously agreed conditions. However, this electricity trading model leads to a number of limitations. The *inflexible pricing model* limits prosumers to sell the electricity only to their original EP. This leads to the prosumer side suffering monetarily as the electricity trading process is settled on a fixed price, leaving no room for negotiation. Further, prosumers have only one EP to trade with, which leads to a value distribution imbalance where contract conditions are dictated by one side of the transaction. However, in the case of energy trading, price margins are regulated by the governmental body, *i. e. ,* country's energy agency.

These limitations can be alleviated by the peer-to-peer (P2P) energy trading model. According to *Directive 2018/2001* (D2018/2001) *of the European Parliament*, a *"peer-to-peer trading of renewable energy means the sale of renewable energy between market participants by means of a contract with predetermined conditions governing the automated execution and settlement of the transaction..."* [5]. P2P trading enables prosumers to trade with different actors and establish an energy *marketplace*. The marketplace serves as a platform for energy market participants who wants to both buy or sell electricity. Within the P2P energy marketplace, prosumers are able to decide where, when, and for what price[1] to buy or sell electricity. This way, the P2P energy trade transforms the previous bilateral agreement with EP into an energy market economy. Ultimately, the P2P energy marketplace aims to incentivize energy grid users to become DER. The financial profitability of privately owned

[1] The price margins are regulated by the government. However, these margins leave room for negotiation for the prosumer to make a higher or lower profit on generated electricity.

DERs promotes RES installation and consequently results in the decarbonization of energy distribution systems [6].

The source of the consumed energy is a matter of concern for the end customer of the marketplace due to the possibility of *inaccurate green consumption information* [7]. In such a case, buyers receive inaccurate information on the sources of electricity they consume, *i. e.*, receive fossil fuel energy when promised renewable. This requires enforcement of a number of certification and regulatory activities, which ensure correct consumption information and RES operation. First, the RES have to be installed and certified by an authorized body to confirm operational capabilities and electricity output into the energy grid. This is done due to the energy grid being part of critical infrastructure where any malfunctioning component may lead to a power outage. The authorized body may be an EP or a privately owned company that has a governmental mandate for RES installation. Further, the marketplace has to ensure correct mapping between produced and traded electricity. The information about the produced electricity is contained in a *guarantee of origin* (GO). According to D2018/2001, the GO is *"an electronic document which has the sole function of providing proof to a final customer that a given share or quantity of energy was produced from renewable sources"* [5]. Thus, when trading renewable energy within a marketplace, the seller has to provide a valid GO, assuring the buyer that the capacity of electricity has been produced with the seller-owned RES. Within the European Union (EU), the GO trade market is regulated via D2018/2001 and CSN EN 16325 standard[2]. However, current legislation has a number of constraints that limit prosumers in their ability to conduct energy trade settlements. The first limitation in trading GOs is that they are primarily issued for substantial quantities of renewable energy, with a minimum threshold of 1MWh. This may be too large for small DERs that produce a couple of dozens of kWh per day. Further, the regulation of renewable energy production and consumption within a country is centralized on the governmental level and regulated by the National Regulatory Authority (NRA). The main role of an NRA is to ensure that the energy sector operates in a safe, reliable, and sustainable manner and to establish and enforce rules and regulations that promote the development of RES. As a result, the NRA is responsible for authorizing a GO issuing body, *i. e.*, the *regulator*. This creates a partial market centralization around the regulator as it has to be involved in the electricity trade process to ensure that GO is properly consumed after energy is sold. Eventually, partial centralization may become a privacy concern, *e. g.*, necessity to disclose prosumer trading data to the regulator to prevent GO double-spending [8].

The current energy marketplaces have a *centralized architecture* [9], where they act as the sole provider of core services such as identity and access management, data assurance, governance, and trade settlement [10]. The marketplace acts as a mediator in all trading activities and needs to be run by a trusted third party (TTP) to ensure compliance with pre-agreed trade settlement conditions and to execute orders from prosumers. This means that prosumers have to rely on their EP to properly operate the marketplace, making the EP the de-facto TTP. Scaling of the marketplace in terms of the number of DERs requires geographic expansion to include multiple energy communities that are operated by different EPs. *The centralized approach becomes problematic when multiple EPs are involved in marketplace operation*, as competition for revenue can lead to a lack of transparency and

[2] https://www.document-center.com/standards/show/BS-EN-16325

raise concerns about fair play. Since EPs are likely privately-owned corporations, they have the incentive to keep their operations confidential to gain competitive advantages. The lack of transparency may cast doubts regarding conformance to negotiated energy trade settlement conditions. Moreover, any EP acting as a TTP would be in a supreme position to profit disproportionately from running the marketplace, creating an asymmetry in value generation [11]. Therefore, this requires the introduction of an external TTP that all the other marketplace actors can trust. In order to overcome the necessity of external TTP, an alternative solution is the adoption of a *decentralized marketplace architecture* that distributes the control of the marketplace over multiple organizations, e.g., the EPs. In this case, all organizations collectively run the marketplace in accordance with protocols that ensure its proper operation, rather than relying on a single TTP [12].

The motivation of this study is to provide an improvement towards today's model of a P2P energy marketplace and regulations that control GO issue and consumption. Further, the aim is to propose a framework of a decentralized P2P energy marketplace adapted to the improved regulations and standards considering trading actors' privacy requirements. Within the marketplace, blockchain technology is used as a main trust-enabling mechanism that allows the establishment of a decentralized energy trading platform. This study aims to address the scalability and trade transaction throughput limitations of today's enterprise blockchain solutions in the context of P2P energy marketplaces. Further, the aim is to propose possible blockchain improvements and provide a system performance evaluation. Blockchain improvements aim to eliminate partial centralization around the regulator, *i. e.* , as a separate governmental authority that oversees all marketplace operations. To effectively address future energy systems, the *regulator* actor is enhanced to address marketplace flexibility and privacy requirements. With the introduced improvements towards the regulator, the prosumers' data can be kept private while preserving the integrity and reliability of P2P trade settlement and the GO issue process. To our knowledge, the improvements towards GO regulations and the regulator role were not studied in the context of any other decentralized energy marketplace.

Based on the discussed energy marketplace challenges and outlined motivations, the main contributions of this paper can be summarized as follows. The study proposes a new model of the blockchain-based P2P energy marketplace with increased flexibility and scalability while addressing actors' privacy and trust requirements. The marketplace utilizes a private permissioned blockchain Hyperledger Fabric (HF) [13] due to its privacy-preserving and trust-enabling capabilities. Further, this study provides a comparison with HF's competitor Hyperledger Besu (HB) [14], and based on identified advantages and limitations, we discuss the rationale for the choice of HF. We utilize HF's *smart contracts* (SC), *i. e.* , chaincodes, to enable P2P energy trade settlement orchestration and GOs management. Based on previous studies conducted in [8] and [15], we propose an improvement towards the HF security by utilizing a Byzantine Fault Tolerant (BFT) consensus mechanism which is protected against malicious system actors [16]. Consequently, to improve marketplace flexibility and data privacy, this study proposes enhancements to D2018/2001 legislation and the regulator actor. *1.)* The minimum quantity of renewable energy that GO can be issued for is 1kWh. This should enable prosumers to trade small amounts of electricity and bring flexibility into the marketplace ordering system. *2.)* In order to become a part of the marketplace, the EP has

to be the NRA-authorized regulator who can issue and consume GOs. In this way, there is no third party participating in the energy trade process, *c. f. ,* the regulator in [8]. Further, such an improvement preserves prosumer's and EP's data privacy and enables P2P energy trade settlement. Next, this study proposes a new multilayered marketplace model to address scalability requirements. In the multilayered model, different energy communities run their own blockchains and are interconnected through layer *interface*, *i. e. ,* a marketplace actor who can communicate with both underlying and overlying layers. Every layer corresponds to an expanding geographical and economic unit, *e. g. ,* energy community, city, region, or country. Finally, we summarize our new approach and talk about future prospects for improvement.

10.2 Energy Trade Regulations

The energy distribution system (EDS) is a critical infrastructure and a vital part of modern society. The power grid is responsible for delivering electricity to individuals' homes, businesses, and industries. Thus, government and private entities invest significantly to maintain and upgrade the EDS to further enable societal development. From a societal perspective, the EDS provides reliable delivery and access to cheap energy that supplies homes and industries. Thus, reliable and affordable electricity is essential for socioeconomic growth. From a governmental perspective, the EDS represents a critical infrastructure and directly affects national security and environmental sustainability. Thus, governments are responsible for establishing laws and regulations that enable EDS monitoring to ensure their security and reliability. Further, governments are responsible for installing regulations that ensure environmental standards, *i. e. ,* promoting green energy and reducing atmosphere carbonization. Consequently, this results in significant investment and regulatory efforts to make the country's EDS self-sufficient, *i. e. ,* relying on the country's own resources and production capabilities, and environmentally sustainable.

One of the strategies to enhance EDSs' sustainability and reliability is exploiting Smart Grid (SG) [17] technologies. SG improves EDS's reliability by implementing and establishing real-time monitoring of energy flows. Further, SG accelerates the integration of RES into EDS, *i. e. ,* making energy production more sustainable. In the SG, certified metering devices deliver real-time information on the electricity generation/consumption to the EP, which enables the development of demand-response programs [18], allowing prosumers to balance their energy usage during peak demand times. Here, demand-response programs allow balancing energy distribution by incentivizing prosumers to output more energy into the grid, *i. e. ,* by reducing consumption and selling it at a higher price. However, such a bilateral energy trade model leads to fixed prices and forces prosumers to trade exclusively with their EP. Thus, the EDS can be further enhanced by establishing an energy marketplace where the prosumers trade the energy via P2P contracts. Such a marketplace should enable price flexibility for prosumers, *i. e. ,* choosing when and whom to sell, and promote RES installation through the profitability of energy trade.

There are a number of regulatory efforts within the energy legislative framework produced by countries to address the promotion and dissemination of RES and the rules of P2P energy trading. The *New York Public Service Commission's REV Connect* [19] program

supports the development of technologies and business modes for the state's EDS. This program concentrates on the dissemination of RESs in the state's EDS to promote a more diverse clean energy ecosystem. Further, the *Federal Energy Regulatory Commission's Order No. 2222* [20] established rules for energy markets and regulates DERs installation and operation. This regulation aims to establish a decentralized and flexible clean energy system by enabling the DERs' participation in electricity markets. Next, the *Australian Energy Market Commission's (AEMC) Wholesale Demand Response Rule* [21] describes the rules and regulations of energy trading. This regulation focuses on the development of efficient demand-response programs by incentivizing owners of RESs to trade their energy in the electricity marketplace. Finally, *Germany's Renewable Energy Sources Act (EEG)* [22] facilitates the development of RES infrastructure and enables P2P energy trading through virtual power plants [23]. The main focus of this regulation is the expansion of RES infrastructure and overall EDS decarbonization.

In European Union, the promotion of RESs and regulation of P2P energy trade is defined in Directive 2018/2001 (D2018/2001) of the European Parliament [5]. The main objective of D2018/2001 is to promote the widespread adoption and dissemination of residential RESs and the establishment of a P2P energy marketplace. This work investigates and proposes amendments towards specifically D2018/2001 due to several reasons. The first reason is that the D2018/2001 defines the energy marketplace platform and traded assets, *i. e.,* electricity and Guarantees of Origin (GOs). Further, it defines the framework for the P2P energy trading process and ensures the protection of prosumers' rights, data privacy, and marketplace actors' responsibilities. Finally, it defines the rules for issuing, transferring, selling, and cancellation of GOs, *i. e.,* GO life-cycle, which are accepted within EU member states. Recognizing GOs management life-cycle rules establishes compliance within EDSs of all EU-member states and enables collaboration towards common renewable energy goals. Further, D2018/2001 is discussed in detail.

10.2.1 Directive 2018/2001 of European Parliament

D2018/2001[3] is a comprehensive regulatory framework that aims to establish an energy marketplace in the EU. One of the key aspects of D2018/2001 is the definition and promotion of the P2P energy trade. It is defined as *the sale of renewable energy between market participants by means of a contract with predetermined conditions governing the automated execution and settlement of the transaction..."* [5]. Every EU member must develop a legal framework and power grid infrastructure to enable prosumers to participate in the P2P energy trade. The important note is that the P2P energy trade can be executed directly by the prosumer or through the third-party EP.

Further, D2018/2001 requires EU member states to establish a framework for an energy marketplace where such a P2P trade settlement can be executed. Such a marketplace should promote RES installation and integration into an EDS as a DER. Ultimately, the marketplace aims to make EDS more flexible, reliable, efficient, and sustainable. The marketplace framework also requires EU members to establish an NRA, which is responsible for implementing the D2018/2001 on a national level. Further, NRA is responsible for

[3] Is a part of a legislative effort called the Clean Energy for All Europeans Package.

overseeing the energy marketplace and ensuring its proper functioning, *i. e. ,* only certified RES and metering devices can become a part of the system. Finally, the NRA is responsible for the GOs management life-cycle and ensures correct mapping between the amounts of virtual kWh for which GOs were issued and actual RES-produced electricity.

Considering the scope of responsibilities of NRA, it is possible to delegate part of the responsibilities to the special actor called the *regulator*, who is subject to a number of requirements. Regulators must comply with the rules established by the NRA and receive its accreditation, *i. e. ,* licensing. Further, regulators must be a legal and independent entity, *i. e. ,* no conflict of interest (malicious intent) towards GO generation, that has technical resources to be part of the marketplace. When licensed, regulators can certify RES and metering devices installed by prosumers and manage the life cycle of GOs. In case of violations regarding GOs management by the regulator, NRA can enforce legal penalties.

Within the D2018/2001, the GO is defined as *"an electronic document which has the sole function of providing proof to a final customer that a given share or quantity of energy was produced from renewable sources"* [5]. The GO can be issued for the amount of energy that is measurable by the metering devices and traceable by the actors responsible for electricity generation registration. The D2018/2001 requires issuing GOs for the amounts above or equal to 1 MWh due to good traceability and sufficient generation measurement accuracy. However, the NRA can adjust this limit based on the capabilities of the EDS and the requirements of P2P trade settlement.

The previous studies [8] and [15] defined HF and HB-based decentralized energy marketplaces compliant with D2018/2001. However, while the compliance led to potential real-world applicability, it resulted in a partial centralization of the marketplace around the regulator. In detail, it resulted in the need to involve the regulator in every energy trade transaction, which makes the regulator de-facto TTP. In addition, high reliance on the certification of RESs and metering devices resulted in the regulator's involvement in energy generation registration within the marketplace, disclosing prosumer generation data to a third party. Considering the presence of these limitations, this paper proposes several amendments in the D2018/2001 targeted at improving the decentralized blockchain-based energy marketplace model and flexibility of the electricity trade process.

10.2.1.1 NRA-licensed EP

In D2018/2001, the regulator has to be an independent organization to issue GOs for generated electricity. Further, it has to be a part of the energy marketplace and oversee all marketplace operations, *i. e. ,* energy registration and trade settlement, to ensure correct mapping of generated electricity and virtual kWh. Finally, when the generated electricity is registered within a marketplace, the regulator has to issue a GO and participate in the energy trade transaction to ensure correct GO consumption, *i. e. ,* preventing double-spending. This requires the prosumer's EP to expose generation and trade transaction data to the regulator, violating data privacy requirements. To preserve prosumers' generation and trade data privacy, this study proposes to *expand NRA regulator certification to the EP within national EDS*. With EP acting as a regulator for the prosumers connected to its locally managed part of the energy grid, there is no need to involve TTP in the electricity registration and trade processes. Further, while having regulator licensing, an EP is able to issue and consume GOs.

Further, in the case of any violations from EP, the NRA is able to impose legal penalties.

10.2.1.2 Increased Flexibility of GOs and Energy Trade

D2018/2001 requires issuing GOs for the amounts above or equal to 1MWh. However, such a limitation may not be sufficient for residential RESs, which can produce smaller amounts of electricity, *i. e.,* several kWh per day. Further, it limits the flexibility of the marketplace energy transactions by requiring prosumers to sell/buy a minimum of 1 MWh per trade. To increase the flexibility of the marketplace, this study proposes to *decrease the minimum amount of energy that GO can be issued for to 1 kWh.* With the current capabilities of residential RES, a change to 1 kWh GO increases prosumers' energy trade flexibility, *i. e.,* allows them to trade more frequently. Further, it allows prosumers to better adjust to price fluctuations and diversify the pool of buyers, *i. e.,* trade with different consumers. Finally, it enables more flexible renewable energy storage strategies for private individuals, *i. e.,* battery size affects its price and installation efforts. However, this regulation improvement proposal should be adopted only if the national EDS capabilities allow acceptable traceability and measurement accuracy for 1 kWh of electricity.

10.3 Marketplace Requirements

The actors and assets must be defined precisely for the P2P renewable energy marketplace to operate correctly. The primary definition of actors and assets was done in [8]. However, considering the amendments towards D2018/2001, requirements have been adapted to the new marketplace structure.

10.3.1 Marketplace Actors

The changes in D2018/2001 lead to the removal of the regulator as a separate entity run by the governmental authority. Instead, EP can now be certified by the NRA and act as a regulator. Further, each role within an energy marketplace is described.

10.3.1.1 Prosumer

The prosumer utilizes a certified RES and metering device and acts as a DER within the marketplace. The prosumer's incentive in becoming a part of the marketplace is to determine the conditions of energy trade settlement with another prosumer or EP. Further, the prosumer wants to get GO for the generated renewable energy utilizing the marketplace's automated services.

10.3.1.2 Energy Provider

In order to sell electricity, prosumers are connected to an EP. An EP's main interest is to profit from energy trade transactions by charging prosumers a fee for utilizing its computing infrastructure, *i. e.,* marketplace services. Further, the EP charges prosumers for the utilization of the energy grid, *i. e.,* in case it is owned by an EP, to transport sold electricity. Further, the EP now has the ability to become a regulator, *i. e.,* by receiving a certification

from NRA, and issue GOs for its prosumers. In addition, EP acts as a certifier of RESs and metering devices for its respective prosumers.

10.3.2 Functional Requirements (FR)

Within the marketplace, there are several functional requirements in terms of services that have to be available for prosumers. Prosumers should have the ability to conduct *operations with the electricity*, which acts as a *fungible token* (FT) [24] within a marketplace. The prosumer should be able to register the generated electricity and sell it via marketplace automated P2P energy trade settlement. In addition, prosumers should be able to work with the electricity market through a *marketplace ordering system* where they create offers to sell and buy electricity of a given quantity at a given price.

Further, certified EP should be able to *issue the GO* on the electricity generated by a specific RES. As GOs act as a *non-fungible tokens* (NFTs) [25], the marketplace must preserve their uniqueness. Further, when the electricity is sold, the GO should be *consumed*, preventing its double-selling.

10.3.3 Non-Functional Requirements (NR)

The marketplace must fulfill the non-functional requirements that concern the correctness of system services execution, data privacy, and marketplace governance. The traded electricity within the marketplace represents the virtual kWh which on a digital level is a number in a database. Thus, *virtual kWh must only be issued following the actual generation of electricity* and certified EP must ensure that there is a perfect match between the virtual kWh in the prosumer's marketplace account and the actual electricity produced in the grid. Further, when an order is being executed, the marketplace system has to *ensure that a needed amount of resources is available* for both sides of the trading operation, *i. e.,* the electricity on the seller's side and currency on the buyer's side. Next, a reliable checking mechanism must be enforced for the *GO to be only issued following the actual generation of electricity.*

Further, considering the sensitivity of the trade transactions, data privacy [26] has to be preserved within the marketplace. *All transactions from a prosumer, including generation, consumption, and purchase, should not be disclosed to other prosumers.* Further, *P2P energy trade details should be disclosed only to those prosumers and their respective EPs who participate in the transaction.*

10.4 Blockchain-based Energy Marketplace

For the energy marketplace, the defined requirements have to be met in order to ensure correct and compliant operation. The prosumers and EPs must have a necessary degree of trust in the marketplace services, *i. e.,* guarantees that trade is executed according to agreed-upon rules, maintaining data provenance, and preventing tampering. Blockchain technology [27] can be used to meet the decentralized marketplace requirements, providing marketplace participants with distributed storage, *i. e.,* the ledger, and bringing such benefits as provenance and accountability to all data processed in a system. It also acts as a consensus-reaching and trust-enabling platform, allowing EPs to establish a trusted relationship in a decentralized

marketplace [28]. When choosing the marketplace blockchain, private permissioned platforms HF and HB are considered. Further, HF and HB capabilities are discussed in detail, and the rationale for a chosen platform is discussed.

10.4.1 Blockchain Platform

HF is a private permissioned blockchain platform which is developed by the Hyperledger Foundation. All transactions in HF are executed within a *blockchain channel*, which establishes a connection between the ledger participants. In HF, there are two types of nodes: *peers* and *orderers*. Further, each node performs a specific task: *endorsement* (peers), *ordering* (orderers), or *validation* (peers). All HF functions are expressed in a *smart contract* (SC), which automates HF-based services execution. Transactions in HF are initiated by the user accounts represented by a public and private key pair that is generated by the dedicated membership service provider.

HB is representative of a private permissioned blockchain platform based on an open-source Ethereum [29] client. HB implements the *Enterprise Ethereum Alliance Protocol* to enable such functionality as private transactions, IAM, and permissioning. In the HB network, the *validator* nodes order, execute, and verify transactions in the blockchain network. All transactions in the HB network are initiated by *user accounts*, representing a public and private key pair that can be generated off-chain. A SC defines functions a user account can call to operate on the data in the ledger.

The trust that all transactions in the marketplace are following the predefined rules is provided by the *consensus mechanism*, *tamper resistance*, and *ledger immutability* of HF and HB. All marketplace functions are expressed as a SC that is audited by all the blockchain organizations, *i. e.,* EPs and regulators, and is stored in the ledger. As a result, there is a clear consensus regarding the rules, expressed as programming code, that the transactions need to follow. Every action that the participants can take in the marketplace is implemented solely through the execution of this SC. Hence, relying on the guarantees provided by the blockchain platform, *i. e.,* HF or HB, that the execution of the SC can be trusted, every marketplace transaction can be trusted to follow the rules.

10.4.1.1 Consensus Mechanisms

In HF, the consensus mechanism is divided into two layers. The first layer denotes the *endorse-order-validate* transaction life-cycle. This life cycle provides the guarantees discussed above, as well as trusted SC execution. The second layer of the consensus mechanism is concentrated on the transaction ordering process. Hyperledger Fabric currently supports only *RAFT* protocol [30] to order transactions. RAFT is *crash fault tolerant* (CFT) consensus which is protected only from orderer node failures, *i. e.,* if less than 50% of the nodes fail, the network can operate successfully. Thus, RAFT consensus needs to be executed by one or several trusted organizations within the blockchain network.

Further, private permissioned HB supports the Proof of Authority (PoA) consensus mechanism QBFT [31] that is identified as byzantine fault tolerant (BFT). BFT consensus provides the same level of protection as CFT and, in addition, can operate in the presence

of adversaries, *e. g. ,* nodes that manipulate transactions and try to disrupt the blockchain network operation.

10.4.1.2 Private Data Handling

A feature of some private blockchain platforms is the ability to store *private data*, *i. e. ,* data that is disclosed only to a subset of the organizations in the blockchain. In HF, there are two approaches to storing private data on HF's ledger: *separate blockchain channel* and *private data collections* (PDCs) [32]. The separate blockchain channel requires deployments of separate ledgers for each private dataset. This approach isolates ledgers from one another and limits private data verification flexibility, *e. g. ,* data integrity cannot be verified by the parties that didn't have access to it initially. Further, the PDC approach enables saving private data in the context of a single blockchain channel. The PDC participants can access and modify the data, while other blockchain channel members collect and store private data hash value, *i. e. ,* to verify the data integrity in case it is disclosed.

In HB, private data is stored in transactions disclosed only to a subset of network participants, *i. e. ,* privacy group (PG), while the rest of the network does not have access to the contents. The private transactions in HB are handled by the *Tessera* private transaction manager. Each organization in HB must have a Tessera node to participate in private transactions. When a new private transaction is generated, it is passed from the HB's validator node to the associated Tessera node. Further, the Tessera node distributes the transaction to the PG members. The rest of the nodes outside the PG receive the record confirming that the private transaction was executed. Such a record consists of a hash of the encrypted transaction data and an indicator that the transaction is private. Further, this record is written into the global ledger.

Nowadays, the HF and HB demonstrate similar capabilities regarding private data processing. Both blockchain platforms provide firm guarantees regarding private data handling, *i. e. ,* blockchain participants can verify data integrity in case of disclosure. However, HF allows modifying public and private data within a single transaction. In HB, public and private data modifications must be done in separate SC functions, *i. e. ,* generating separate transactions. This results in HB requiring more transactions to execute complex automated services which require to modify both private and public blockchain data.

10.4.2 Blockchain Platform Choice Rationale

We choose HF as a blockchain platform for implementing the P2P energy marketplace with updated regulations described in Section 10.2. The choice of HF has several factors derived based on the studies conducted in [8] and [15]. The rationale of HF choice is described next:

- HF has a modular transaction life cycle, *i. e. ,* endorse-order-validate, which enables a pluggable consensus mechanism and flexibility of ordering process configuration.
- HF PDCs enable public and private data modification within a single transaction which is paramount for the P2P trade settlement execution process.
- As was demonstrated in our previous studies [8] and [15] HF demonstrates double the throughput of HB for a P2P energy marketplace SC.

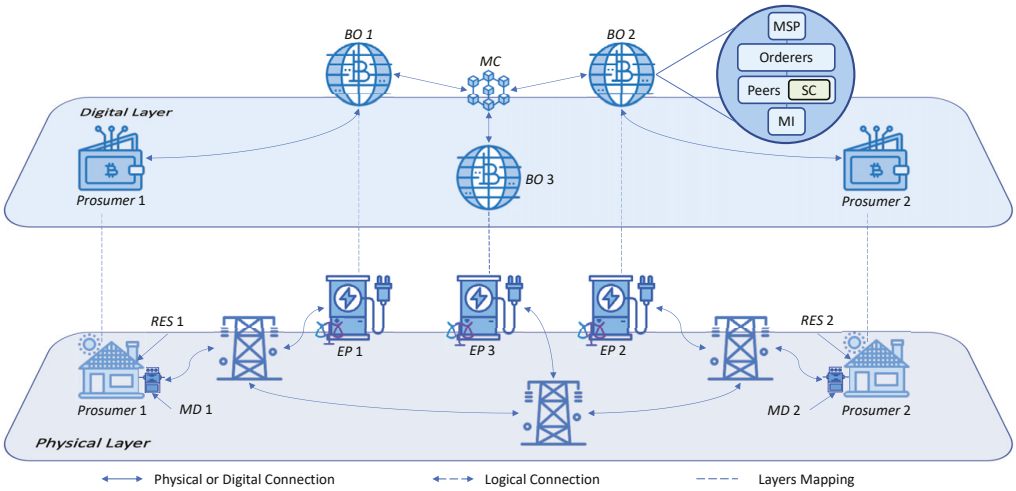


Figure 10.1: *Energy Marketplace.* (Physical layer, i.e., energy grid, is mapped to a digital blockchain-based layer, where the electricity trade operations are executed).

The disadvantage of HF is that it uses CFT consensus for the block ordering process. Thus, the ordering process must be executed by one or several trusted organizations within the blockchain network. This results in HF blockchain network's partial centralization around the trusted entity. In [8], the regulator acted as a trusted party, assuring the validity of all transaction ordering processes within the blockchain channel. However, since EPs can now become regulators, the ordering process cannot be executed by a single EP. Thus, every EP operating within a marketplace must run at least one orderer node to ensure the integrity and auditability of the ordering process.

To address the limitation of CFT consensus, we propose an improvement towards HF ordering process by utilizing the BFT-SMART (BFTS) [16] consensus mechanism. BFTS implements the Practical BFT (PBFT), following the three-phase commit process, *i. e.*, pre-prepare, prepare, and commit. A new block is disseminated to all orderers with a *pre-prepare* message. Then, orderers broadcast *prepare* message. When receiving *prepared* replies from *2/3+1* of orderers, the leader orderer broadcasts *commit message*. When *commit* message is received by *2/3+1* of orderers, the new block is written to the ledger.

10.4.3 Marketplace Architecture

The architecture of the energy marketplace is depicted in Figure 10.1. It consists of two layers: *physical* and *digital*. The physical layer is the energy grid, where generated electricity is distributed. The digital layer is the communication network between the energy providers and prosumers where trading of the electricity, *i. e.*, virtual kWh, takes place. In order to correctly map both marketplace layers, each physical layer actor has to have representation in the digital layer, *c. f.*, Figure 10.1. EPs act as a separate *blockchain organization* (BO). The BO operates a number of peer nodes, which are the primary guarantors of valid transaction execution and require the most computational power. Further, acting as regulators, EPs

operate a number of orderer nodes responsible for block ordering. All peers and orderers are interconnected, forming the *marketplace channel* (MC). Further, each EP BO has a *marketplace interface* (MI), which is utilized by the prosumers to execute market functions. In addition, all BOs have a dedicated *membership service provider* (MSP), which generates cryptographic identity information for the prosumers who join the marketplace. Prosumers do not act as BO, *i. e. ,* have no peers, relying on the EPs to endorse and validate trade settlement transactions on their behalf. Further, prosumers are represented within a marketplace as records in the ledger, *i. e. ,* blockchain wallets. Finally, prosumers also have a *metering device* (MD) installed in their households, which monitors the RES-generated electricity and sends the data to the EP.

The marketplace system supports a number of functions to enable the energy trade process. The *electricity generation registration* function registers data from the prosumers metering device in the prosumer's account in the blockchain. Further, the *GO management* function enables GOs' issuing and consumption. Next, the marketplace enables an energy *ordering system*, where prosumers can create *buy* and *sell* electricity orders. Finally, the marketplace enables a *P2P energy trading* process, where prosumers can trade with each other utilizing the computing infrastructure of their respective EP.

10.5 Marketplace Implementation

Both system design and HF's technological characteristics are implemented to address all defined marketplace functional and non-functional requirements from Section 10.3. Further, we discuss HF's data structure and SC developed for the defined P2P energy marketplace design.

10.5.1 Blockchain Data Structure

All data records saved on the blockchain ledger or PDC are generated by the respective SC that is deployed in the network. The prosumer is represented within a marketplace as logically connected data records that define entities associated with electricity trade, *c. f. ,* Tables 10.1 and 10.2.

Table 10.1: *Prosumer Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Prosumer's unique identifier
Electricity	*Double*	Amount of prosumer generated electricity *(kWh)*
WalletID	*String*	Prosumer's Wallet identifier
EnergyProviderID	*String*	Prosumer's Energy Provider identifier

10.5.1.1 Prosumer Blockchain Data Record

The prosumer record is described in Table 10.1. It is private for the prosumer, the respective EP, *c. f. ,* PDC2 or PDC3 in Figure 10.2. This record contains prosumer unique *ID*, which is generated by the marketplace SC when he/she joins a marketplace. Further, the prosumer

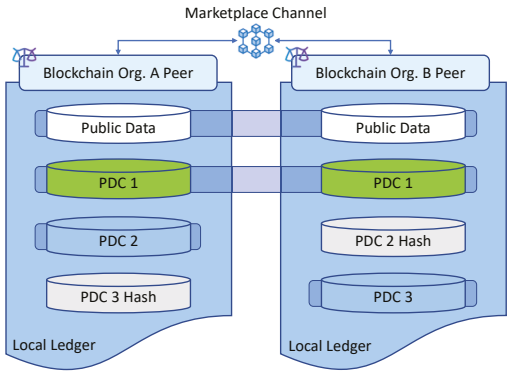


Figure 10.2: *Marketplace PDC.*

record contains generated *Electricity*, updated by the EP based on the data from the prosumer's metering device. In addition, the *WalletID* field connects the prosumer record to a particular wallet within the marketplace. The prosumer record also contains the *EnergyProviderID*, which establishes the link with a specific EP that issues GOs on generated electricity. The prosumer record intentionally does not contain any Personally Identifiable Information (PII) to comply with data regulation documents, *e. g.*, GDPR [33]. All PII needed for legal purposes can be saved in the conventional DB. Further, the record in conventional DB can contain an ID from the prosumer blockchain record, establishing a link between PII and non-PII saved in the ledger. If a prosumer leaves the marketplace, the PII record can be deleted from the conventional database. Further, the records in the blockchain can also be marked as deleted.

Table 10.2: *Wallet Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Wallet's unique identifier
Currency	*Double*	Amount of fiat currency, *e. g.*, USD, EUR
Electricity	*Double*	Amount of prosumer bought electricity *(kWh)*

10.5.1.2 Wallet Blockchain Data Record

The wallet record is described in Table 10.2. It contains the *Currency* field, *i. e.*, the amount of fiat currency a prosumer has at disposal. In addition, the wallet record contains *Electricity* field, which shows the amount of bought electricity. The wallet record *Electricity* and the prosumer record *Electricity* fields are separated to ensure that the bought electricity is not resold twice. Every prosumer must have only one wallet record. The wallet record is private to the prosumer and the respective EP *c. f.*, PDC2 or PDC3 in Figure 10.2. However, due to HF's PDC design, the buyer and seller wallets are visible to EPs' peers during a P2P energy trade. This is a necessary measure for the selling prosumer to ensure that the buyer has enough currency in his/her wallet.

Table 10.3: *GO Blockchain Data Record*

Field Name	Type	Description
ID	*String*	GO unique identifier
OwnerID	*String*	GO owner ID
RegulatorID	*String*	Issuer of GO
RESID	*String*	Certified RES generator of electricity
ElectricityAmount	*Double*	Amount of GO certified electricity *(kWh)*
IsConsumed	*Boolean*	Set *True* when electricity is sold

10.5.1.3 GO Blockchain Data Record

The GO blockchain data record is described in Table 10.3. The *OwnerID* field contains an identifier of the prosumer who owns GO. Further, the *RegulatorID* field contains an identifier of the EP, who is the issuer of GO and guarantor of its validity. Next, the *RESID* field contains an identifier of the RES used to generate the electricity. Such a RES must be certified by the EP. Further, *ElectricityAmount* field contains the amount of electricity certified by the EP for trading. Finally, the *isConsumed* field is set *True* when the energy GO was issued for is sold.

Table 10.4: *Order Blockchain Data Record*

Field Name	Type	Description
ID	*String*	Order unique identifier
Type	*String*	Order Type *(Sell or Buy)*
Price	*Double*	Electricity Price *(For the entire amount sold)*
ElectricityAmount	*Double*	Amount of electricity *(kWh)*
GOID	*String*	GO unique identifier
SellerWalletID	*String*	Seller wallet identifier
BuyerWalletID	*String*	Buyer wallet identifier

10.5.1.4 Order Blockchain Data Record

The order record is described in Table 10.4. The *Type* field shows what kind of order it is, *i. e. ,* sell or buy. Further, the *Price* field contains the entire sum of currency that has to be paid for this order. In addition, the *ElectricityAmount* field contains the number of kWh traded. Next, the *GOID* links the GO data record for the electricity order. Finally, the *SellerWalletID* and *BuyerWalletID* fields contain identifiers of prosumer wallets. Depending on the type of the order, when it is created, one of the wallet identifiers is left empty, *i. e. , SellerWalletID* is empty for a buy order, and *BuyerWalletID* is empty for a sell order. When the order is fulfilled, it is private for prosumers and EPs participating in trade settlement, *c. f. ,* PDC1 in Figure 10.2.

10.5.2 Marketplace Smart Contract (SC)

The HF enables broad capabilities in working with the data saved on the ledger or in PDC. For marketplace purposes, the SC was developed where all services defined in Section 10.4.3, *i. e. ,* defined as SC functions, were implemented. Next, we outline the base functions used

Algorithm 14 Fulfill Buy Electricity Order

Require: *Order* (*Type* = *Buy*), *GO*, *Seller Wallet*, *Buyer Wallet*

```
1: function SellElectricity(orderId string, goId string, sellerId string, sellerWalletID string, sellerPDC string, buyerPDC string, tradePDC string)
2:     order ← GetState(orderId)                                   ▷ Retrieving the buying order record.
3:     go ← GetState(goId)                                         ▷ Retrieving the record of GO proposed by seller.
4:     sellerWallet ← GetPrivateData(sellerPDC, sellerWalletId)    ▷ Retrieving the seller wallet record.
5:     buyerWallet ← GetPrivateData(buyerPDC, order.BuyerWalletID) ▷ Retrieving the buyer wallet record.
6:     if go.OwnerID == sellerId & go.IsConsumed == False then
7:         if go.ElectricityAmount == order.ElectricityAmount then
8:             if buyerWallet.Currency ≥ order.Price then
9:                 order.SellerWalletID ← sellerWalletID
10:                order.GOID ← goId
11:                sellerWallet.Currency ← sellerWallet.Currency + order.Price
12:                buyerWallet.Currency ← buyerWallet.Currency − order.Price
13:                buyerWallet.Electricity ← buyerWallet.Electricity + order.ElectricityAmount
14:                PutPrivateData(tradePDC, orderId, order)    ▷ Saving order in PDC1, c. f. , Figure 10.2.
15:                PutPrivateData(sellerPDC, sellerWalletId, sellerWallet)
16:                PutPrivateData(buyerPDC, order.BuyerWalletID, buyerWallet) ▷ Save wallets in the EPs PDCs, c. f. , PDC2 and PDC3 in Figure 10.2.
17:                Call FinalizeOrder(orderId, goId)           ▷ c. f. , Algorithm 15.
18:            else
19:                return "Insufficient buyer currency"
20:        else
21:            return "Insufficient electricity amount"
22:    else
23:        return "Invalid GO"
```

Algorithm 15 Finalize Order

Require: *GO*, *Order*

```
1: function FinalizeOrder(orderId string, goId string)
2:     order ← GetState(orderId)                    ▷ Retrieving the order record to delete.
3:     go ← GetState(goId)                          ▷ Retrieving the GO record to consume.
4:     go.IsConsumed ← True                         ▷ Mark GO as consumed.
5:     PutState(goId, go)                           ▷ Saving consumed GO in the Public Data ledger, c. f. , Figure 10.2.
6:     DeleteState(orderId)                         ▷ Deleting fulfilled order.
```

in HF SCs to read and write public and private data. Further, we describe in detail a *sell electricity* function used for the performance evaluation.

Contract Level Base Functions: The HF defines several base functions used to read and write ledger or PDC data. The *GetState(key)* function reads an asset with a particular id from the public data ledger, *c. f. ,* Figure 10.2. Further, the *PutState(key, value)* function creates or updates the record on the public data ledger. To read private data saved in the PDC, the *GetPrivateData(collection, key)* function is used. A *collection* name has to be provided to read the data from the correct PDC. Further, a peer who tries to read private data from a particular PDC must be authorized according to the access list. Next, in order to write to PDC, the *PutPrivateData(collection, key, value)* function is used.

Private data introduces additional constraints for the execution of the SC. Different peers may need to endorse the transaction when public or private data are added or updated. Thus, to correctly endorse the transaction where public and private records are modified, some

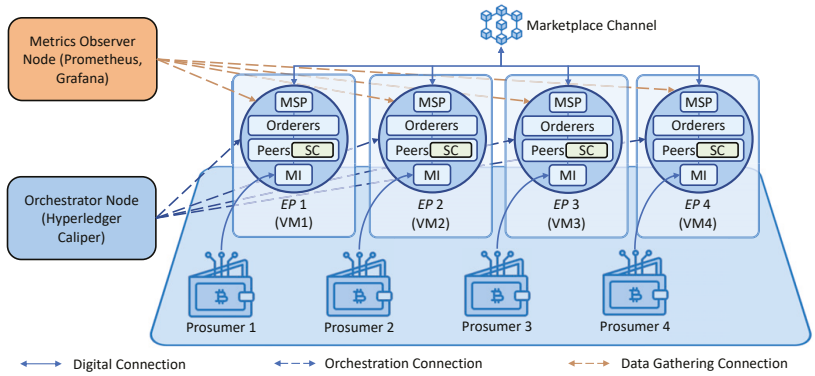


Figure 10.3: *Implemented Energy Marketplace.*.

values from the private data may need to be communicated as part of the transaction input, even if they are not members of the PDC. By design, the input parameters of the transaction proposal are visible to peers who endorse the transaction. Thus, all input parameters of the transaction where private data is manipulated are stored in the *Transient* storage within our marketplace SC functions. Transient storage is not included in the transaction proposal and is used to hide the input parameters from peers who are not members of the PDC [32].

Finally, the data asset on the ledger or PDC can also be marked as deleted. It remains in the ledger, but the record itself cannot be used in transactions. The *DeleteState(key)* function is used to delete public data, and the *DeletePrivateData(collection, key)* is used for private data deletion.

Electricity Trade Smart Contract Function: This study describes in detail the trade settlement SC function that fulfills the buy customer electricity order used for performance evaluation. Prosumers may fulfill the buy order by utilizing *SellElectricity* SC function, *c. f.*, Algorithms 14. The trade settlement operation execution has two stages executed within a single transaction. The *SellElectricity* function takes a buy order posted by a prosumer-buyer. Since the seller brings a new GO into the order, its validity is checked, *i. e.*, ownership, consumption, and electricity amount fields. Further, the buyer's currency is checked to be greater or equal to the order price. If the buyer currency check is successful, the seller currency is increased, and both wallets are saved in the respective PDC, *c. f.*, PDC2 or PDC3 in Figure 10.2. Finally, the order record is saved in the EPs PDC, *c. f.*, PDC1 in Figure 10.2. In the second stage, the *FinalizeOrder* function is executed, *c. f.*, Algorithm 15. First, this function takes the GO, sets its *IsConsumed* value to *True*, and saves it in the public ledger. Further, it marks the fulfilled order as deleted, *i. e.*, it does not appear in the pool of orders for prosumers.

10.6 Performance Evaluation

The aim of the performance evaluation is to investigate the impact of BFTS consensus on the performance of energy marketplace private transaction execution in comparison to CFT RAFT. The test implementation is deployed on four virtual machines (VMs), *c. f.*,

Figure 10.3, where each VM size is 16 vCPUs, 64 GB RAM, and 256 GB high throughput (150MBit/s) disk space. Each EPs runs its own VM. All VMs are connected with a 10Gbit/s network interface. All peer and orderer nodes within the infrastructure are deployed as docker containers. To collect reliable and correct performance evaluation data, *Prometheus*[4], *Grafana*[5], and *Hyperledger Caliper*[6] (HC) tools are utilized. The *Prometheus* is used as the main blockchain operation data collector. The *Grafana* is used as a data visualization tool. The *HC* performance evaluation tool is used as a transaction load generator.

Several performance metrics are considered in this study. The *throughput* is the number of successful transactions (TPS) or reads (RPS) executed per second in the blockchain network. The *latency* is the time it takes to finalize transaction execution and write it to the ledger. The *scalability* is the behavior of the network with an increasing number of peer and orderer nodes. The summary of performance evaluation parameters is shown in Table 10.5.

Table 10.5: *Performance Evaluation Parameters*

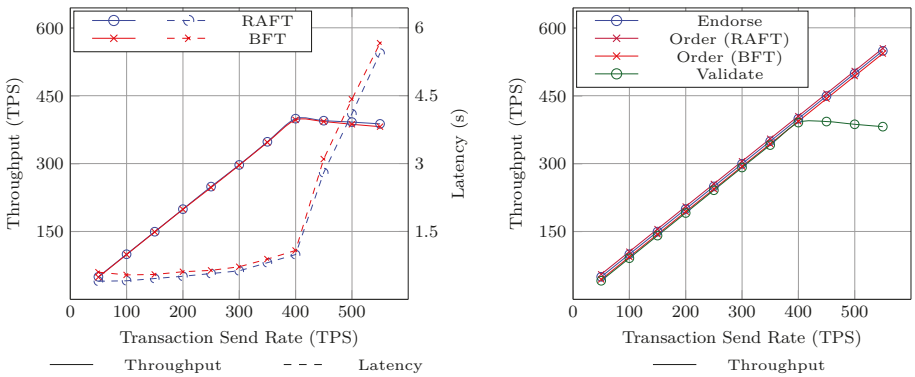
Parameter	Value
Transaction Send Rate (Write)	50 → 550 with step of 50 *(fixed-rate in duration of 5 minutes)
Transaction Send Rate (Read)	100, 300 → 3000 with step of 300 *(fixed-rate in duration of 5 minutes)
BOs	4 → 16 with step of 4
Orderer Nodes	4 → 16 with step of 4
Consensus Mechanism	BFTS, RAFT

10.6.1 Write - Sell Electricity Function Execution

In this study, an Algorithm 14 was executed as an SC function to test maximum write TPS. An Algorithm 14 was chosen due to requiring the highest amount of computations out of all defined SC functions. To write a transaction to the ledger, the endorse-order-validate transaction life-cycle must be executed. We test the baseline HF configuration with BFTS, which includes the minimum necessary infrastructure setup to operate, *i. e.,* 4 VMs with four peers and four orderers. Further, we execute the same experiments with RAFT consensus. The optimal block configuration, *i. e.,* block size and batch timeout, was previously investigated and established in [8]. Block Size corresponds to the maximum number of transactions one block can fit and is set to 500 Tx. Batch Timeout specifies the timeout to receive incoming transactions that go into one block and is set to 1 second. The sell electricity transaction is private for two EPs that trade electricity. The throughput measurement results are shown in Figure 10.4a. The BFTS consensus demonstrates a similar maximum throughput as CFT RAFT, *i. e.,* approximately 400 TPS. Further, BFTS and RAFT consensuses demonstrate similar latency, *i. e.,* up to 1s. Any send rate higher than 400 TPS significantly increases latency for both BFTS and RAFT.

To investigate the impact of each stage of the endorse-order-validate transaction life-cycle, their throughput is measured individually, *c. f.,* Figure 10.4b. The results demonstrate that the main bottleneck is the *validate* stage, *i. e.,* executed on peer, limiting the resulting

[4] https://prometheus.io/docs/introduction/overview/
[5] https://grafana.com/
[6] https://hyperledger.github.io/caliper/



(a) *Transaction throughput and latency with varying send rate (*Block Size = 500Tx, Batch Timeout = 1s, Peers = 4, Orderers = 4, BOs = 4, PDC = 2*).*

(b) *Throughput of all HF transaction life-cycle stages with varying send rate (*Block Size = 500Tx, Batch Timeout = 1s, Peers = 4, Orderers = 4, BOs = 4, PDC = 2*).*

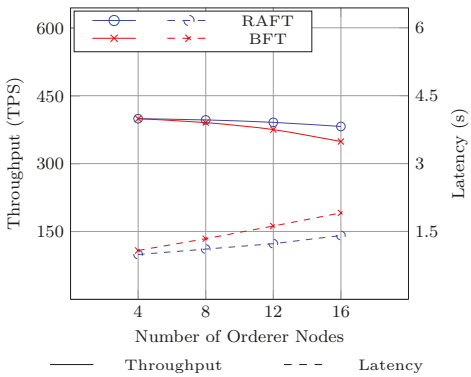
Figure 10.4: *Hyperledger Fabric performance and scalability evaluation results.*

throughput to 400 TPS. The validate stage requires the highest amount of computations and executes each transaction's validation sequentially, resulting in poor process scalability. Both BFTS and RAFT consensus do not show any limitations in terms of order stage throughput scalability for the baseline configuration of 4 orderer nodes.

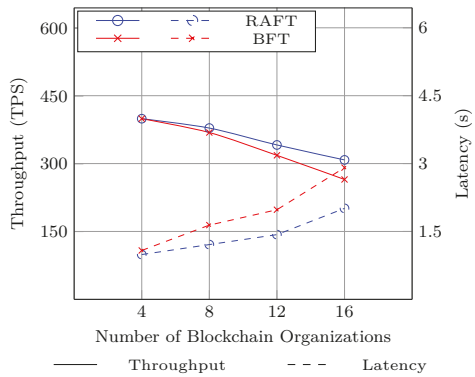
The horizontal scalability was investigated with a varying number of orderer nodes and BOs, *i. e. ,* one BO corresponds to one peer and one orderer node. The results of orderer scalability are shown in Figure 10.5a. Since only the orderer node number is changing, the resulting throughput is affected only by the BFTS or RAFT consensus mechanisms. The RAFT consensus demonstrates better scalability than BFTS, showing minimal-to-no throughput loss at 16 orderer nodes. Further, the decrease in throughput of BFTS consensus is a result of the three-phase commit process, which requires more computations and information exchange between orderers to provide protection against malicious nodes. Finally, RAFT demonstrates slightly lower latency than BFTS. This is a result of increased information exchange between orderer nodes in BFTS consensus.

The results of BO scalability are shown in Figure 10.5b. Since each BO corresponds to one peer and one orderer node, the resulting throughput is affected by the entire endorse-order-validate transaction life-cycle. Here, RAFT demonstrated better throughput than BFTS. However, both consensus mechanisms demonstrate a significant throughput drop due to the increasing number of peer nodes for every BO. RAFT demonstrates approximately 25% throughput drop from scaling the BO number from 4 to 16, while BFTS shows 33% throughput loss under the same conditions, *i. e. ,* a number of peer and order nodes.

The performance evaluation results demonstrate that under the baseline HF setup, both RAFT and BFTS demonstrate similar throughput and latency. Further, BFTS provides protection against malicious orderer nodes, in contrast to RAFT, which is only CFT. In terms of scalability, RAFT demonstrates slightly better results both for the order stage of the



(a) *Throughput and Latency for varying number of* Orderer Nodes *(*Block Size = 500Tx, Batch Timeout = 1s, Peers = 4, BOs = 4, 400 TPS send rate*).*



(b) *Throughput and Latency for a varying number of* BOs *(one peer and one orderer per BO) (*Block Size = 500Tx, Batch Timeout = 1s, 400 TPS send rate*).*

Figure 10.5: *Hyperledger Fabric scalability evaluation results.*

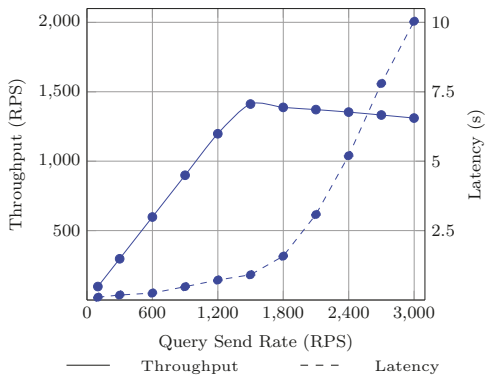


Figure 10.6: *Read Throughput and Latency (*4KB asset*).*

transaction life-cycle and BO increase. However, since BFT consensus is a key mechanism needed to enable amendments towards D2018/2001 it is considered a better candidate for P2P renewable energy marketplace deployment.

10.6.2 Ledger Data Read

An endorse-order-validate transaction life-cycle does not have to be executed in order to query, *i. e.*, read, the blockchain data. Thus, the consensus mechanism, *i. e.*, RAFT or BFTS, does not affect the read throughput. It is the amount of data, *i. e.*, *asset size*, that is read from the HF peer node that affects read throughput. The results of the read throughput investigation are shown in Figure 10.6. The query was constructed to read 4KB of data from the LevelDB world state database. The results demonstrate the maximum throughput of

approximately 1412 RPS. The latency remains under 1s until peak read throughput if the send rate is increased.

10.7 Results and Observations

The usage of the BFT consensus protocol enabled the implementation of the proposed amendments towards D2018/2001. Further, throughout the process of the marketplace architecture design, and system implementation, a number of observations were made. Such observations are discussed next.

10.7.1 The New Energy Platform Has a Higher Degree of Decentralization

The change in D2018/2001 legislation led to the removal of the need for a trusted organization in the marketplace scheme, *i. e.,* eliminating system partial centralization points. Instead, all EPs operating within the marketplace also act as regulators, certifying prosumers' RES and issuing GOs on the generated electricity. Further, HF's ordering process is supported by the BFTS consensus mechanism, which protects from a certain amount of malicious nodes. Considering the decentralization of the system and the equal status of all BOs in terms of processes execution, *i. e.,* GO issuing and consumption, and private data handling, the defined HF-based P2P energy platform can be considered as one with a higher degree of decentralization compared to [8]. A higher degree of decentralization makes the marketplace more democratic and robust, due to fair value distribution and increased fault-tolerance [10].

10.7.2 Collusion Possibility for EP-regulators

Considering the degree of decentralization, there may be a downside to this marketplace design. Assigning the role of the regulator to the EP may affect the level of governmental control over the marketplace and energy trade within the national EDS. Multiple EPs may collude and try to manipulate the electricity market prices, *i. e.,* pump-and-dump strategy. Thus, this marketplace design requires strong regulatory controls over the EPs who become regulators, as there is a probability of their collusion and market manipulation. A set of legal penalties should be established as a deterrence mechanism, making the cost of violation higher than the potential gains from market manipulation.

10.7.3 Multilayered Marketplace for Improved Scalability and Outreach

The increasing number of EPs, *i. e.,* BOs, added to the marketplace deployed in one MC results in the decreased throughput of energy trade settlement operations. In order to address the scalability of the HF-based P2P energy marketplace, a new multilayered model is proposed. In the multilayered model, different energy communities run their own MC and are interconnected through layer *interface blockchain organization* (IBO), *i. e.,* a marketplace actor who can communicate with both underlying and overlying layers. The multilayered model involves the usage of multiple MCs to distribute energy trade settlement execution.

The multilayered marketplace model is depicted in Figure 10.7. It consists of several layers which are situated one on top of another and represent an expanding geographical and

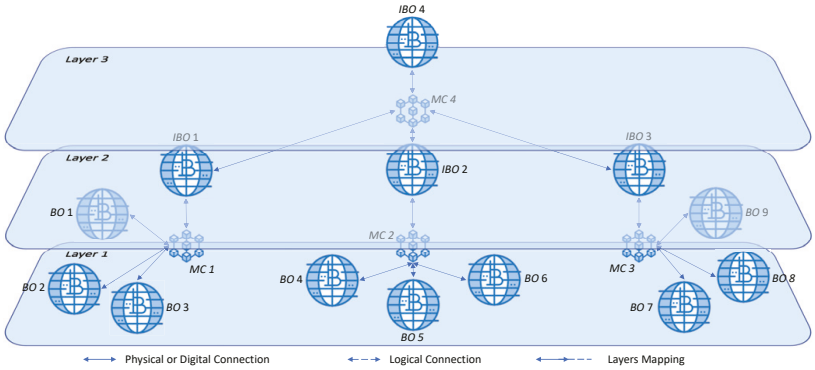


Figure 10.7: *Multilayered Energy Marketplace.* (Each layer corresponds to an expanding geographical and economic unit).

economic unit, *e. g. ,* energy community, city, region, or country. *Layer 1* contains EPs which are represented by BOs distributed over the specific geographical location within concrete economic boundaries, *e. g. ,* a country. Further, BOs are interconnected into local MC, *i. e. , MCs* 1, 2, and 3, to enable prosumers of a certain energy community to conduct P2P energy trade settlement. Next, each local MC is connected to a respective IBO as an entity which provides the ability to communicate with the upper layer. IBOs connected to Layer 1, *i. e. , IBOs* 1, 2, and 3, are interconnected into *Layer 2* MC, *i. e. , MC* 4. Finally, the *IBO* 4 connects Layers 2 and 3, allowing further scalability onto bigger geographic and economic entities.

The IBO performs two actions in the multilayered marketplace model. First, IBO can act as an intermediary in a P2P trade settlement between prosumers that are situated geographically far from each other. From the prosumer's point of view, the P2P trade settlement is still an automated function executed by the marketplace. For BOs and IBOs involved in the transaction, it is a multi-transaction process executed within every MC between the trading prosumers. Second, IBO can act as a national aggregator of energy, *i. e. ,* GOs, which enables cross-country renewable energy trade.

10.8 Related Work

A number of proposals have emerged in the field of decentralized blockchain-based energy marketplaces recently. In [34], the authors propose a platform for blockchain-based energy trading. This platform aims to provide prosumers with the ability to trade electricity after the generation period. According to the authors, blockchain enables P2P energy trading after the generation period allowing prosumers to trade electricity in P2P mode without the need for TTP. Authors of [35] propose a blockchain-based P2P energy marketplace that enables electricity trading without a need for TTP. The marketplace allows for cost reduction and improvement of energy distribution strategies. In [36], the authors propose a blockchain-based energy marketplace that utilizes ML with the aim of enhancing the DER energy distribution strategies. The authors claim that the proposed trading platform enables

the construction of efficient energy distribution strategies while maintaining QoS. The authors of [37] define a decentralized P2P energy marketplace that provides a unified electricity trade model. The marketplace is based on HB's private permissioned blockchain platform. The authors conduct a performance evaluation of an implemented marketplace with Istanbul BFT (IBFT) consensus mechanism and compare results with Ethereum's Proof of Work (PoW) and Clique consensus mechanisms. The authors claim their system has double the throughput compared to existing PoW-based systems. Further, the results demonstrate lower latency and possible optimizations of the energy trading process, *i. e.,* reduction in the number of transactions needed to trade electricity. In [38], the authors propose a blockchain-based P2P energy marketplace for decentralized electricity production and trading. The platform consists of the market and blockchain layers. The market layer consists of an auction allowing prosumers to bid on energy trading proposals. According to the authors, an auction model results in efficient inter-temporal market product trading. The blockchain layer implements SC and automates decentralized marketplace functions. The authors conduct a platform evaluation that demonstrates the increased efficiency of energy trading via blockchain-based settlements. The authors of [39] conduct an in-depth investigation on the role of blockchain technology in microgrids. According to the authors, blockchain technology enables potential solutions for electrification in the transportation, building, and industrial sectors. Further, blockchain-based microgrids enhance the electrification opportunities for remote areas, *e. g.,* islands, towards a green networking ecosystem. The authors claim, that their study may serve as a comprehensive reference for modern microgrids, *i. e.,* their control and communication technology with the integration of blockchain services for the sustainable energy supply chain. In [40], the authors propose a P2P energy marketplace based on novel strategies for bilateral electricity trade, *i. e.,* supply and demand matching based on a distance between producer and consumer. Energy trade strategies evaluation demonstrates more efficient energy distribution while reducing the energy price. For further reading on the developments in blockchain-based energy marketplaces see [2].

10.9 Summary and Outlook

This study proposes a new model of the blockchain-based P2P energy marketplace with increased flexibility and scalability while addressing actors' privacy and trust requirements. The marketplace utilizes HF private permissioned blockchain. Further, to improve marketplace flexibility and data privacy, this study proposes amendments to D2018/2001 legislation and the regulator actor. *1.)* The minimum quantity of renewable energy that GO can be issued for is 1kWh. This should enable prosumers to trade small amounts of electricity and bring flexibility into the marketplace ordering system. *2.)* In order to become a part of the marketplace, the EP has to be authorized by the NRA to become the regulator and issue and consume GOs. In this way, there is no external trusted party participating in the energy trade process, *i. e.,* the governmental regulator. Further, such an improvement preserves prosumer's and EP's data privacy and enables P2P energy trade settlement. Finally, this study proposes a new multilayered marketplace model to address scalability requirements. In the multilayered model, different energy communities run their own blockchains and are interconnected through layer *interface*, *i. e.,* a marketplace actor who can communicate with

underlying and overlying layers. Every layer corresponds to an expanding geographical and economic unit, *e. g.,* energy community, city, region, or country.

The results demonstrate that the new marketplace design with the BFTS consensus application enables assigning the role of the regulator to the EP. Further, the new design enables the P2P energy trade automation while providing better privacy-preserving capabilities than the HF-based marketplace design described in [8], *i. e.,* no external trusted party participates in the trade process. Next, the BFTS consensus protects the blockchain network from malicious nodes and demonstrates a similar maximum throughput (400 TPS) as CFT RAFT, *i. e.,* within the endorse-order-validate life cycle. Further, assigning the role of the regulator to the EP may affect the level of governmental control over the marketplace and energy trade within the national EDS. Such a change requires strong regulatory controls over the EPs who become regulators, as there is always a risk of them colluding in order to manipulate the market. Thus, auditing combined with significant legal penalties should be established to efficiently deter market manipulation.

The consensus mechanism is fundamental for the secure, compliant, and trusted collaboration between parties in decentralized systems. Thus, investigation on how to improve consensus performance and scalability is of interest.

References

[1] L. Strezoski, H. Padullaparti, F. Ding, and M. Baggu. "Integration of Utility Distributed Energy Resource Management System and Aggregators for Evolving Distribution System Operators". In: *Journal of Modern Power Systems and Clean Energy* 10 (2 Mar. 2022), pp. 277–285. DOI: 10.35833/MPCE.2021.000667.

[2] S. Gawusu, X. Zhang, A. Ahmed, S. A. Jamatutu, E. D. Miensah, A. A. Amadu, and F. A. J. Osei. "Renewable energy sources from the perspective of blockchain integration: From theory to application". In: *Sustainable Energy Technologies and Assessments* 52 (2022), pp. 1–26. DOI: 10.1016/j.seta.2022.102108.

[3] S. Saxena, H. Farag, A. Brookson, H. Turesson, and H. Kim. "Design and Field Implementation of Blockchain Based Renewable Energy Trading in Residential Communities". In: *2019 2nd International Conference on Smart Grid and Renewable Energy (SGRE)*. IEEE, Nov. 2019, pp. 1–6. DOI: 10.1109/SGRE46976.2019.9020672.

[4] Y. Yang, S. Zhang, and Y. Xiao. "Optimal design of distributed energy resource systems coupled with energy distribution networks". In: *Energy* 85 (2015), pp. 433–448. DOI: 10.1016/j.energy.2015.03.101.

[5] EU Parliament. *Directives Directive (EU) 2018/2001 of the European Parliament*. 2022, pp. 82–209. URL: http://data.europa.eu/eli/dir/2018/2001/2022-06-07 (visited on 06/18/2023).

[6] B. Hertz-Shargel, D. Livingston, and A. C. of the United States. *Assessing Blockchain's future in transactive energy*. 2019. ISBN: 9781619775992. URL: https://www.atlanticcouncil.org/in-depth-research-reports/report/assessing-blockchains-future-in-transactive-energy/ (visited on 06/18/2023).

[7] Á. Hamburger. “Is guarantee of origin really an effective energy policy tool in Europe? A critical approach”. In: *Society and Economy* 41 (2019), pp. 487–507. DOI: 10.1556/204.2019.41.4.6.

[8] R.-V. Tkachuk, D. Ilie, R. Robert, V. Kebande, and K. Tutschku. “Towards Efficient Privacy and Trust in Decentralized Blockchain-Based Peer-to-Peer Renewable Energy Marketplace”. In: *Sustainable Energy, Grids and Networks* (2023), pp. 1–27. DOI: 10.1016/j.segan.2023.101146.

[9] R.-V. Tkachuk, D. Ilie, and K. Tutschku. “Towards a Secure Proxy-based Architecture for Collaborative AI Engineering”. In: *2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW)*. Naha, Japan: IEEE, 2020, pp. 373–379. DOI: 10.1109/CANDARW51189.2020.00077.

[10] R.-V. Tkachuk, D. Ilie, K. Tutschku, and R. Robert. “A Survey on Blockchain-Based Telecommunication Services Marketplaces”. In: *IEEE Transactions on Network and Service Management* 19.1 (2022), pp. 228–255. DOI: 10.1109/TNSM.2021.3123680.

[11] T. Kollmann, S. Hensellek, K. de Cruppe, and A. Sirges. “Toward a renaissance of cooperatives fostered by Blockchain on electronic marketplaces: a theory-driven case study approach”. In: *Electronic Markets* 30.2 (2020), pp. 273–284. DOI: 10.1007/s12525-019-00369-4.

[12] F. Rahimi, S. Nikhil, G. Gourisetti, J. Kempf, E. Alejandro, A. Flores, C. Lima, H. Albright, P. D. Heitmann, and T. Martinez. *IEEE Blockchain Transactive Energy (BCTE)*. 2021. URL: https://blockchain.ieee.org/verticals/transactive-energy (visited on 06/18/2023).

[13] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick. “Hyperledger Fabric”. In: *Proceedings of the Thirteenth EuroSys Conference*. 2018, pp. 1–15. DOI: 10.1145/3190508.3190538.

[14] H. Foundation. *Hyperldger Besu Ethereum client*. 2022. URL: https://besu.hyperledger.org/en/stable/ (visited on 06/20/2023).

[15] R.-V. Tkachuk et al. “On the Performance and Scalability of Consensus Mechanisms in Privacy-Enabled Decentralized Renewable Energy Marketplace”. In: *Annals of Telecommunications* (June 2023), pp. 1–18. DOI: 10.1007/s12243-023-00973-8.

[16] A. Barger, Y. Manevich, H. Meir, and Y. Tock. “A Byzantine Fault-Tolerant Consensus Library for Hyperledger Fabric”. In: *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 2021, pp. 1–9. DOI: 10.1109/ICBC51069.2021.9461099.

[17] T. Ahmad, R. Madonski, D. Zhang, C. Huang, and A. Mujeeb. “Data-driven probabilistic machine learning in sustainable smart energy/smart energy systems: Key developments, challenges, and future research opportunities in the context of smart grid paradigm”. In: *Renewable and Sustainable Energy Reviews* 160 (2022), p. 112128. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2022.112128.

[18] C. Ibrahim, I. Mougharbel, H. Y. Kanaan, N. A. Daher, S. Georges, and M. Saad. "A review on the deployment of demand response programs with multiple aspects coexistence over smart grid platform". In: *Renewable and Sustainable Energy Reviews* 162 (2022), p. 112446. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2022.112446. URL: https://www.sciencedirect.com/science/article/pii/S1364032122003525.

[19] New-York State. *REV Connect*. 2015. URL: https://www3.dps.ny.gov/W/PSCWeb.nsf/All/B2D9D834B0D307C685257F3F006FF1D9 (visited on 05/04/2023).

[20] US Federal Government. *REV Connect*. 2022. URL: https://www.ferc.gov/media/ferc-order-no-2222-fact-sheet (visited on 05/04/2023).

[21] H. Gibbs. *AEMC Wholesale Demand Response Rule*. 2018. URL: https://www.aemc.gov.au/rule-changes/wholesale-demand-response-mechanism (visited on 05/04/2023).

[22] German Government. *Germany's Renewable Energy Sources Act (EEG)*. 2020. URL: https://www.iea.org/policies/12392-germanys-renewables-energy-act (visited on 05/04/2023).

[23] S. Sierla, M. Pourakbari-Kasmaei, and V. Vyatkin. "A taxonomy of machine learning applications for virtual power plants and home/building energy management systems". In: *Automation in Construction* 136 (2022), p. 104174. ISSN: 0926-5805. DOI: 10.1016/j.autcon.2022.104174.

[24] K. B. Wilson, A. Karg, and H. Ghaderi. "Prospecting non-fungible tokens in the digital economy: Stakeholders and ecosystem, risk and opportunity". In: *Business Horizons* 65 (5 Sept. 2022), pp. 657–670. DOI: 10.1016/j.bushor.2021.10.007.

[25] Q. Wang, R. Li, Q. Wang, and S. Chen. *Non-Fungible Token (NFT): Overview, Evaluation, Opportunities and Challenges*. 2021. URL: http://arxiv.org/abs/2105.07447.

[26] V. Koutsos, D. Papadopoulos, D. Chatzopoulos, S. Tarkoma, and P. Hui. "Agora: A Privacy-Aware Data Marketplace". In: *IEEE Transactions on Dependable and Secure Computing* 19.6 (2022), pp. 3728–3740. DOI: 10.1109/TDSC.2021.3105099.

[27] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. URL: https://bitcoin.org/bitcoin.pdf (visited on 06/18/2023).

[28] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and Y. Liu. "A Survey of Blockchain Technology Applied to Smart Cities: Research Issues and Challenges". In: *IEEE Communications Surveys and Tutorials* 21.3 (2019), pp. 2794–2830. DOI: 10.1109/COMST.2019.2899617.

[29] G. Wood. *Ethereum: a secure decentralised generalised transaction ledger*. Ethereum Project Yellow Paper, 2014, pp. 1–32. URL: https://gavwood.com/paper.pdf (visited on 06/18/2023).

[30] D. Woos, J. R. Wilcox, S. Anton, Z. Tatlock, M. D. Ernst, and T. Anderson. "Planning for change in a formal verification of the raft consensus protocol". In: *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs*. ACM, 2016, pp. 154–165. DOI: 10.1145/2854065.2854081.

[31] H. Moniz. "The Istanbul BFT Consensus Algorithm". In: (2020). URL: http://arxiv.org/abs/2002.03613.

[32] S. Wang, M. Yang, Y. Zhang, Y. Luo, T. Ge, X. Fu, and W. Zhao. "On Private Data Collection of Hyperledger Fabric". In: *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, July 2021, pp. 819–829. DOI: 10.1109/ICDCS51616.2021.00083.

[33] EU Parliament. *Regulation (EU) 2016/679 of the European Parliament (General Data Protection Regulation)*. 2016, pp. 1–99. URL: https://gdpr-info.eu/ (visited on 06/18/2023).

[34] J. Mello, J. Villar, R. J. Bessa, M. Lopes, J. Martins, and M. Pinto. "Power-to-Peer: a blockchain P2P post-delivery bilateral local energy market". In: *2020 17th International Conference on the European Energy Market (EEM)*. 2020, pp. 1–5. DOI: 10.1109/EEM49802.2020.9221901.

[35] R. Khalid, N. Javaid, A. Almogren, M. U. Javed, S. Javaid, and M. Zuair. "A Blockchain-Based Load Balancing in Decentralized Hybrid P2P Energy Trading Market in Smart Grid". In: *IEEE Access* 8 (2020), pp. 47047–47062. DOI: 10.1109/ACCESS.2020.2979051.

[36] F. Jamil, N. Iqbal, Imran, S. Ahmad, and D. Kim. "Peer-to-Peer Energy Trading Mechanism Based on Blockchain and Machine Learning for Sustainable Electrical Power Supply in Smart Grid". In: *IEEE Access* 9 (2021), pp. 39193–39217. DOI: 10.1109/ACCESS.2021.3060457.

[37] J. Abdella, Z. Tari, A. Anwar, A. Mahmood, and F. Han. "An Architecture and Performance Evaluation of Blockchain-Based Peer-to-Peer Energy Trading". In: *IEEE Transactions on Smart Grid* 12 (4 2021), pp. 3364–3378. DOI: 10.1109/TSG.2021.3056147.

[38] A. Esmat, M. de Vos, Y. Ghiassi-Farrokhfal, P. Palensky, and D. Epema. "A novel decentralized platform for peer-to-peer energy trading market with blockchain technology". In: *Applied Energy* 282 (2021), pp. 1–16. DOI: 10.1016/j.apenergy.2020.116123.

[39] Y. Wu, Y. Wu, H. Cimen, J. C. Vasquez, and J. M. Guerrero. "Towards collective energy Community: Potential roles of microgrid and blockchain to go beyond P2P energy trading". In: *Applied Energy* 314 (2022), pp. 1–16. DOI: 10.1016/j.apenergy.2022.119003.

[40] T. AlSkaif, J. L. Crespo-Vazquez, M. Sekuloski, G. van Leeuwen, and J. P. S. Catalao. "Blockchain-Based Fully Peer-to-Peer Energy Trading Strategies for Residential Energy Systems". In: *IEEE Transactions on Industrial Informatics* 18 (2022), pp. 231–241. DOI: 10.1109/TII.2021.3077008.

This thesis aims to advance the knowledge on the efficient design and evaluation of distributed marketplaces with an emphasis on trust and privacy. Although distributed, most of today's marketplaces are centrally governed, limiting its services interoperability and automation. Thus, a decentralized marketplace model can be adopted to distribute governance among multiple organizations, enhancing marketplace scalability as well as service automation and interoperability. However, trust issues are raised if more than one organization governs the marketplace. Thus, a decentralized marketplace requires a robust and secure digital trust-enabling mechanism to automate service execution.

This thesis investigates distributed marketplaces where centralized and decentralized governance models are applied to use cases of Artificial Intelligence (AI) artifacts and renewable energy trading. It begins with a study of a marketplace for AI artifacts where a Secure Virtual Premise is defined to enable AI pipeline execution in a centrally governed system. The thesis continues with a survey of the telecommunication services marketplaces, where centralized and decentralized governance models are discussed. In addition, the survey provides an in-depth investigation of blockchain technology as a main trust-enabling platform. Having mapped the state-of-the-art, the research shifts towards an in-depth investigation of blockchain-based decentralized renewable energy marketplaces. The studies provide an in-depth requirements definition, system architecture, implementation, and performance evaluation of marketplaces based on two major blockchain platforms. The final study of this thesis provides improvements towards the renewable energy marketplace model aiming to enhance digital trust, privacy, and scalability. Ultimately, such a marketplace should incentivize the widespread adoption of renewable energy sources, resulting in the decarbonization of electricity distribution systems.

Printed by Media-Tryck, Lund 2024 NORDIC SWAN ECOLABEL 3041 0903

Blekinge Institute of Technology
Doctoral Dissertation Series No. 2023:13
ISSN: 1653-2090
ISBN: 978-91-7295-465-6