

Research Article

Serialization of Telecom Provisioning Transactions in Distributed Systems

Henry P Cyril*

Independent Researcher, Anna University, Chennai, India

Received 01 Dec 2025, Accepted 20 Dec 2025, Available online 22 Dec 2025, Vol.15, No.6 (Nov/Dec 2025)

Abstract

Telecom provisioning transactions play a central role in the provisioning and termination as well as the alteration of communication services within the modern telecommunications networks. The transactions span a number of dispersed systems, including network management systems (NMS), business support systems (BSS), operations support systems (OSS), different network components, and external third-party platforms. The growth in the use of cloud-native software, network virtualization, and software-defined networking has significantly increased the complexity, scale, and frequency of provisioning operations. The paper is a systemic review of the transaction of telecom provisioning, its life cycle, the general operation conditions, and the major entities involved. It also looks at the issue of handling transactions in distributed telecom systems like dependency ordering and the conflicts of concurrency, race conditions and deadlocks. Different provisioning transaction models such as synchronous transactions, asynchronous transactions and long-running transactions are studied to reveal their appropriateness to the different operational needs. Moreover, the paper reviews and compares methods of serialization and concurrency control, such as pessimistic and optimistic control, lock-based, timestamp ordering, queue-based, and token-based models, and event-driven, message-based, serialization. The trade-offs between consistency, scalability, and performance are discussed and the necessity of sound transaction management structures to the realization of service reliability, proper billing, and high service-level agreements in next-generation telecom networks highlighted.

Keywords: Transaction, Banking, 5G-network, Provisioning, Telecommunication, Distributed system.

Introduction

Introduction

The telecom industry is also interested in adopting this data delivery technique. However, fragmentation is a crucial and crucial step that must be completed before distribution. The database model is taken into consideration while selecting the fragmentation approach. After being created, the fragments are dispersed to the locations where they are required. The intensity of data in telecommunication services has risen due to recent advancements in network technology [1]. At this time, the telecom industry should adopt distributed data management methods that are fault-tolerant, more reliable, and less costly. However, there are certain extremely important concerns that need to be resolved before this new field may be used in the telecom sector. Concurrency control, availability, transparency, recovery, and throughput are the main concerns here.

In contrast to other conventional businesses that employ distributed databases, telecommunications databases must meet stricter standards for availability, recovery, throughput, overall dependability, and response time.

Distributed systems are characterized by the occurrence of multiple transactions on provisioning to meet high service demand and rigid service-level agreements (SLAs)[2]. Concurrency introduces race conditions, resource contention, and partial failures, which can lead to unpredictable system states unless handled appropriately. Serialization is also found to be a primary transaction management mechanism that provides a logical ordering for simultaneous telecom provisioning transactions, enabling their consistent execution as sequential transactions. Serializability can be used to ensure that distributed telecom systems remain correct without compromising scalability or parallelism.

Telecom provisioning needs serialization because of the existence of shared, and vital, resources, including subscriber databases, network configurations, virtual network functions (VNFs), and spectrum assignments[3]. The lack of control over

*Corresponding author's ORCID ID: 0009-0000-7606-9862
DOI: <https://doi.org/10.14741/ijcet/v.15.6.6>

simultaneous access to these resources may lead to service disruptions, misconfigurations, or incorrect billing. As a result, pessimistic and optimistic concurrency control, lock-based, timestamp ordering, and event-driven processing are all popular serialization techniques used to manage transactions in distributed telecom platforms. Moreover, the telecom provisioning transactions are usually long-lasting and cut across administrative and technological domains, and thus the traditional centralized transaction control is not feasible[4]. Saga-based and compensation-driven models of distributed transaction management are becoming increasingly popular to ensure system fault tolerance and resilience. These models are based on serialization strategies that ensure consistency but allows network latency, node failures and atypical communication patterns characteristic of distributed systems. Operationally agile and scalable, efficient serialization of provisioning transactions is necessary in the era of 5G and beyond with network slicing, edge computing, and ultra-low-latency services as some of the enablers[5]. A major research and engineering challenge remains balancing strict consistency requirements with performance and availability constraints. This paper discusses telecom provisioning transaction serialization in distributed system, its underlying models and techniques, issues and implications of such systems on next generation telecom networks.

Structure of the paper

This paper is organized as follows: Section II overview of telecom provisioning transactions. Section III discusses transaction management in distributed telecom systems, while Section IV explains serialization techniques and models. The literature and case studies are reviewed in Section V. Included are findings and suggestions for more research in Section VI, Conclusions.

Telecom Provisioning Transactions

A provisioning management system plays a vital role in the telecommunications industry, especially for service providers, by automating the tasks involved in telecom provisioning. This powerful system helps streamline order processing, resource allocation, configuration management, and activation procedures for data services. It also improves customer service by enhancing the efficiency of the provisioning process.

Conceptually, a provisioning transaction starts with a commercial request in BSS (e.g., “activate 5G plan”), flows through order management, is translated into technical tasks in OSS, and finally reaches network elements for configuration and activation[6]. The lifecycle typically includes order capture, validation, resource reservation, configuration, activation, testing, and completion/closure, with status updates and logs

at each step to ensure traceability and rollback if anything fails.

Typical operations of telecom provisioning

Telecom provisioning operations define the core processes through which communication services are enabled, adjusted, and withdrawn during their lifecycle is shown in table I. These operations ensure seamless coordination between business systems and network elements, maintaining service continuity, billing accuracy, and efficient resource utilization.

Service Activation

Service activation is the step in which a prepared service is turned on and put into customer use. Once all the necessary resources (numbers, SIM/eSIM, bandwidth, IP addresses, QoS profiles, etc.) are configured in the network, activation commands are issued to the corresponding network elements (such as HLR/HSS/UDM, PCRF/PCF, MSC, gNB, or broadband access nodes). This is the stage where customer machines are granted access, policies are implemented and test calls/pings can be made to ensure that the service is operational. The go-live of the provisioning lifecycle is referred to as activation.

Service Modification

Transactions in service modification modify an existing service but without fully deactivating it. Examples include upgrading a data plan, activating roaming packs, activating VoLTE or 5G on an existing SIM, or changing the bandwidth of an existing fixed broadband line. In BSS, a change order is generated and verified as eligible and affecting the charges, OSS then calculates the delta configuration and only applies the required changes to network entities usually whilst maintaining the service running[7]. Correct serialization is used to make sure that conflicting changes (e.g. two upgrades being submitted within close time intervals) do not overlap and that billing, policy, and network configuration remain unchanged.

Service Termination

The de-provisioning of a service (service termination) destroys the service permanently and frees the resources. When the customer cancels or churns, BSS terminates the contract, completes the billing (including any fines), and sends out a termination order. OSS then releases instances of service off of network elements, deconfigures them, returns numbers, IPs, and other resources to inventory and adjusts assurance systems to properly respond to alarming of decommissioned elements[8]. The correct serialization is paramount here in preventing the orphaned resources, phantom bills, or the long access to the resources even after the formal end date.

Table 1 Typical Operations of Telecom Provisioning

Operation	Purpose	Key Provisioning Activities	Systems & Network Elements Involved	Role of Serialization
Service Activation	To enable a newly ordered service for customer usage	Allocation and configuration of resources such as numbers, SIM/eSIM profiles, bandwidth, IP addresses, QoS policies; activation commands issued to network elements; test calls and connectivity validation	BSS (order capture & billing), OSS (service orchestration), HLR/HSS/UDM, PCRF/PCF, MSC, gNB, broadband access nodes	Ensures activation steps occur in correct order and prevents partial or duplicate service enablement
Service Modification	To change attributes of an already active service without interruption	Delta configuration computation; plan upgrades, roaming add-ons, feature enablement (VoLTE/5G), bandwidth adjustment; policy and billing updates while maintaining service continuity	BSS (change order & charging), OSS (delta provisioning), policy servers, access and core network elements	Prevents overlapping or conflicting change requests and maintains consistency across billing, policy, and network states
Service Termination	To permanently decommission a service and reclaim resources	Contract closure; final billing and penalties; deconfiguration of services; release of numbers, IP addresses, and network capacity; inventory update	BSS (contract termination & billing), OSS (de-provisioning), inventory systems, assurance platforms, network elements	Prevents orphaned resources, phantom billing, and unauthorized access after service end

Participants of Telecom Provisioning Transactions

Telecom provisioning transactions require the involvement of many systems and entities that are tightly integrated, whereby each one of them is responsible for turning customer requests into actions that the network can technically execute[9]. As mentioned in this part, effective communication between the business, operational, network, and external platforms is necessary for service fulfillment to be precise, billing to be synchronized, and end-to-end reliability to be maintained.

OSS (Operations Support Systems)

The systems that support the technical and operational aspects of telecom provisioning include OSS, which provide the network inventory, configuration, service fulfillment, fault and performance management. In provisioning operations, OSS breaks high-level requests into technical operations, coordinates workflows between NMS/EMS and network components, monitors operations, and communicates results back to BSS. Modern OSS is also used to orchestrate virtualized resources and do zero-touch (automated) provisioning with 5G, NFV and SDN.

BSS (Business Support Systems)

BSS deal with business elements of telecom services, product list, price, customer handling, order entry, and billing. A typical provisioning transaction starts at this point: customer order is confirmed (credit checks, eligibility, contract terms), priced, and converted into a service order, which can be fulfilled by OSS[10]. Another similarity between BSS, charging, and general billing is the alignment of charging and billing with the service's technical status, so that billing occurs precisely when activation and termination events occur in the network.

NMS (Network Management Systems)

NMS provides centralized control over specific network domains or technologies (e.g., IP/MPLS core, radio access, transport). During the provisioning process, NMS accepts configuration and activation messages sent by OSS and converts them into network element-specific commands or APIs. They store topology, inventory and performance information about their domain, and can serve both bulk provisioning (e.g. mass CPE configuration) and real-time configuration changes, playing a vital role as an intermediate between high-level orchestration and low-level machines.

Network Elements

The hardware or virtual functions that implement the service in the data and control planes are network elements (NEs), which can include routers, switches, base stations, gateways, softswitches, firewalls, CPEs, and VNFs/CNFs. NEs are updated with configuration during provisioning transactions with routing rules, VLANs, QoS policies, subscriber profiles, and access control lists[11]. The correct arrangement of services is the last thing that can make services work to the end user, hence error handling, rollback, and idempotent NE level are required to provide reliable provisioning in the distributed environment.

Third-Party API and External Systems

The contemporary telecom provisioning frequently goes beyond the operator to third-party platforms and APIs number portability databases, identity and KYC services, cloud service providers, OTT partners, payment gateways and MVNE/MVNA platforms. Examples of such transactions are API calls to provision external resources, to deploy bundled services (such as streaming subscriptions), or to reconcile status with partner systems[12]. Such

interactions should be synchronized with internal OSS/BSS processes to ensure end-to-end consistency, typically through asynchronous message passing, webhooks, and clearly articulated SLAs to address delays and partial failures.

Transaction Management in Distributed Telecom Systems

Transactions in distributed telecom systems are an important operation that provide consistency, reliability, and accuracy when delivering demanding services across multiple network elements, OSS/BSS systems, and cloud-native components. Telecom provisioning is the coordinated change to many heterogeneous systems, and often across administrative domains and vendors[13]. To ensure data integrity, to ensure service continuity, and to achieve high service-level agreements (SLAs) in the face of high transaction volumes, network latency, and the partial failure of systems, effective mechanisms of transaction management are necessary.

Types of provisioning transactions

There are many types of provisioning transactions as discussed below:

Synchronous Provisioning Transactions

Synchronous provisioning transactions are highly coordinated processes, meaning that each process waits to see the other response to the operation before moving on to the next phase of the process. These are typical transactions when activating or modifying a service in real time and need immediate approval, e.g. SIM activation or the allocation of bandwidth[14]. Although synchronous processing provides good consistency and deterministic results, it may cause latency and scalability issues in large, distributed telecom systems.

Asynchronous Provisioning Transactions

Asynchronous provisioning transactions prevent the immediate response to provisioning tasks that can be executed, and instead permit systems to process request independently through message queues or event architectures[15]. This model enhances scalability, fault tolerance, and responsiveness of system, and is applicable to a non-real-time service or bulk provisioning operations. Asynchronous execution, however, presents problems in maintaining transaction state and achieving eventual consistency across distributed components.

Provisioning Long-Running Provisioning Transactions

Long-running provisioning operations consume lengthy periods of time and consist of series of dependent operations, including end-to-end service

coordination of core, access, and edge networks[16]. These transactions cannot be based on the conventional locking schemes, as they take a long time. They generally use saga-based or compensation-based strategies to handle partial failures and ensure system stability.

Transaction dependencies and ordering constraints

Telecom systems are such that provisioning transactions are characterized by strict dependencies and ordering constraints in the same way[17]. For example, network resources must be assigned before services can be activated, and authentication settings must be established before user access can be granted. Transaction ordering must be properly enforced to avoid inconsistent states, service failures, or resource leaks in distributed environments.

Concurrency issues

Concurrency issues arise in telecom provisioning systems when multiple transactions are processed concurrently over shared network and service resources. These challenges, summarized in Table II, highlight how uncontrolled parallelism can lead to inconsistencies, service disruptions, and system stalls in distributed OSS/BSS environments.

Concurrency Conflicts

Concurrency conflicts occur when provisioning transactions attempt to use or update network or service resources simultaneously. Such conflicts may result in inconsistencies in data, including duplication or overwriting of configuration parameters[18]. Such conflicts require effective concurrency control mechanisms, such as locking, versioning, and access control.

Race Conditions

Race conditions occur when the timing or sequence of parallel provisioning transactions depends on the timing or sequence of other transactions. Race conditions can cause randomly dependent service behavior, e.g., premature activation of a service or erroneous entry in the billing system, in telecom systems[19]. These timing-related inconsistencies need to be prevented through careful synchronization, atomic operations, and event sequencing.

Deadlocks

Deadlocks occur when there are two or more provisioning transactions that wait indefinitely in each other, leading to stalls in the system and delays of service. Deadlocks can be influencing in a major way to provisioning in a distributed telecom system with multiple resources dependencies. The common techniques to deal with this issue include deadlock detection, order of resources and recovery using timeout.

Table 2. Concurrency Issues in Telecom Provisioning Systems

Issue Type	Description	Potential Impact on Provisioning	Common Mitigation Techniques
Concurrency Conflicts	Occur when multiple provisioning transactions simultaneously access or update the same network or service resources, leading to data inconsistencies	Duplicate allocations, overwritten configuration parameters, inconsistent service states across OSS/BSS	Locking mechanisms, version control, role-based access control, optimistic and pessimistic concurrency control
Race Conditions	Arise due to timing and execution order of parallel provisioning operations, where outcomes depend on sequence rather than logic	Premature service activation, incorrect billing entries, unpredictable service behavior	Synchronization mechanisms, atomic transactions, event sequencing, ordered message processing
Deadlocks	Occur when two or more transactions wait indefinitely for each other's locked resources, preventing progress	Service provisioning delays, system stalls, failed activation or termination processes	Deadlock detection and recovery, resource ordering strategies, timeout-based lock release

Serialization Techniques and Models

Distributed telecom provisioning systems techniques such as serialization guarantee transactional consistency and provide parallel processing. These models identify the sequence of concurrent transactions and the means to control and prevent conflicts, inconsistencies, and service disruption. System scale, performance requirements, and the criticality of provisioning operations all lead to different approaches to serialization.

Pessimistic and Optimistic Concurrency Control

Pessimistic concurrency control is the belief that conflicts between provisioning transactions are common and as such, avoids such conflicts by blocking access to shared resources beforehand. Transactions obtain exclusive or shared resource control prior to execution with a high level of consistency and usually at the expense of lower parallelism and increased latency[20]. On the contrary, optimistic concurrency control presupposes that conflicts are infrequent and that transactions can be executed simultaneously without any restrictions. Only during commit time are conflicts detected, and transactions can be rolled back if inconsistencies are found. More optimistic methods tend to be more scalable and better suited to high-throughput telecom environments with low contention.

Lock-Based Serialization Mechanisms

Lock-based serialization orders transactions based on locking shared resources such as subscriber profiles, network configurations, or virtualized network functions. These locks may be fine grained (granular) or rough (coarse-grained) with respect to system design. Although the lock-based mechanisms offer high levels of serializability, they may create certain performance problems like blocking, deadlock, and low throughput which are a major concern in the large-scale distributed telecom systems with high degree of concurrency.

Timestamp Ordering and Version Control

Serializations based on a time-stamp are such that each transaction gets a special time-stamp or logical

sequence which ensures that all transactions follow a globally consistent sequence. The transactions are confirmed by their time stamps and operations that break the order are either aborted or postponed. Version control systems build on this idea and store an assortment of data objects in different versions, enabling read operations to use older, consistent versions, and write transactions to generate newer versions. This method is better concurrent and less blocking, thus it is more suitable to distributed and read-intensive telecom provisioning environments.

Queue-Based Serialization and Token-Based Serialization

In token-based serialization a logical token is used, which permits a provisioning transaction to be performed on a shared resource. The transaction that contains the token is the only one that can be executed, and order is guaranteed as well as conflict is avoided[21]. Queue-based serialization arranges the incoming transactions to orderly queues, which are processed in a sequence; this is depending on priority or dependency that is predetermined. These mechanisms are especially useful for providing deterministic execution of critical provisioning operations, but can cause delays during peak transaction loads.

Event-Based and Message-Based Serialization

Serialization through event-driven and message-queued-based communication is based on asynchronous communication and processing of messages in order thus attaining transaction serialization. Provisioning requests are provided as events or messages to the queue and are processed in order by consumers in a predetermined or partitioned fashion[22]. It improves scalability, fault tolerance and decoupling of systems components which makes it very appropriate with the current cloud-native telecom architectures. Nonetheless, the and-only storage of data and the rigorous ordering of distributed queues may be difficult to provide and require careful planning.

Literature of Review

This section presents earlier studies on Telecom Provisioning Transactions. Table III provides a structured comparison of previous research.

Balakrishnan, De, and Wang (2025) propose a computationally efficient cooperative energy-transfer-based distributed energy bank strategy to alleviate green energy outages and to design energy-sustainable networks. It first develop low-complexity Markovian frameworks to estimate green energy outage in a standalone BS without energy cooperation (WEC) and a multi-BS energy-cooperative (EC) setting, respectively. For the WEC system, it's a computationally efficient three-state discrete time Markovian statistical model, while the multi-BS EC framework is characterized by a two-state Markov model[23].

Guan, Zhang, and Zheng (2024) propose a secure, private blockchain-based intelligent e-commerce platform designed to streamline commercial and trade flows while fostering trust and security. By leveraging blockchain's immutability and implementing end-to-end encryption (E2EE), the platform ensures secure transactions and enhances data privacy, addressing common pain points in digital commerce. Furthermore, incorporating large language models (LLMs) enhances the user experience through personalized recommendations and efficient customer service. The synergy between blockchain and AI not only optimizes operational efficiency but also cultivates a more engaging and satisfying customer journey[24].

Yang *et al.* (2023). Real-time changes in energy demand are accounted for, along with two scenarios: energy oversupply and a conservative supply. The equilibrium solution of the Stackelberg game is obtained using the backward induction method and an iterative algorithm. This includes the ESP unit energy price, the telecom operator's total power, the percentage of transmission power allocated to the relay service, and the utility payment plan. The suggested algorithm may rapidly and precisely converge to the optimal solution to the problem,

according to simulation results. The technique can also enhance demand-side control stability, lower utility costs, and boost telecom operator profits[25].

Aeri *et al.* (2023) Customer churn, also known as attrition or customer defection, is the process by which customers stop being affiliated with a business or subscribing to a certain service or business. A crucial component of corporate strategy in today's fiercely competitive market, when gaining new clients can be expensive, is comprehending and reducing customer attrition. The purpose of this abstract is to provide a succinct overview of customer turnover, including its origins, effects, and methods to reduce churn rates. According to experimental findings, the Random Forest predicts customer attrition with a 79% accuracy rate[26].

Li *et al.* (2022). Around the world, telecom operators are deploying fifth-generation (5G) networks that offer low communication latency, extensive network coverage, high data transfer rates, and convenient access to a large number of devices. However, telecom operators face several obstacles throughout the development and operation of 5G, including inadequate frequency resources, inefficient network management, opacity in data, interoperability issues, and network privacy vulnerabilities. Given that 5G is often implemented in diverse networks with numerous ubiquitous devices, it is imperative to offer decentralized and secure solutions[27].

Pang *et al.* (2022) found that the economic value of telecom big data generated by telecom networks is substantial. To reduce the chance of customer privacy being compromised, telecom carriers are actively investigating big data transaction techniques. The current solutions only allow purchasers to upload data-mining algorithms to the telecom operator's platform for training; they don't allow datasets to leave the database. However, there is a significant chance that this centralized platform tampered with. Furthermore, the information of telecom big data transactions cannot be precisely and swiftly traced using the present technologies[28].

Table 3. Comparative Analysis of Telecom Provisioning Transactions

Author & Year	Focus Area	Technique	Key Contributions	Future Work
Balakrishnan, De & Wang (2025)	Green energy management in telecom networks	Low-complexity Markovian models (three-state for WEC, two-state for EC), cooperative energy transfer	Proposes computationally efficient distributed energy bank strategies to mitigate green energy outage and support energy-sustainable base station networks	Extension to large-scale heterogeneous networks, integration with real-time traffic prediction, and validation using real-world energy datasets
Guan, Zhang & Zheng (2024)	Secure intelligent e-commerce platforms	Private blockchain, End-to-End Encryption (E2EE), Large Language Models (LLMs)	Ensures secure transactions, improved data privacy, and enhanced user experience through AI-driven personalization and intelligent customer service	Scalability analysis for high transaction volumes, energy-efficient blockchain design, and deeper integration of generative AI for autonomous commerce
Yang et al. (2023)	Energy demand management in telecom-utility	Stackelberg game model, backward induction, iterative optimization algorithm	Addresses energy oversupply and conservative supply scenarios; improves demand-side regulation stability, reduces utility costs, and increases telecom operator profit	Consideration of stochastic renewable energy sources, multi-operator competition, and real-time adaptive pricing mechanisms

	cooperation			
Aeri et al. (2023)	Customer churn prediction	Machine learning models, Random Forest classifier	Demonstrates that Random Forest achieves 79% accuracy in predicting telecom customer churn, supporting proactive retention strategies	Incorporation of deep learning and explainable AI (XAI), inclusion of behavioral and temporal features, and deployment in real-time CRM systems
Li et al. (2022)	Secure 5G network deployment	Decentralized and blockchain-based security frameworks	Identifies major challenges in 5G such as privacy, interoperability, and network management inefficiencies, advocating decentralized security solutions	Design of lightweight blockchain protocols for 5G/6G, cross-domain interoperability, and AI-driven automated security management
Pang et al. (2022)	Telecom big data transactions and privacy	Secure data trading frameworks, privacy-preserving analytics	Highlights risks of centralized data platforms and challenges in traceability and tamper resistance in telecom data transactions	Development of decentralized data marketplaces, integration with federated learning, and fine-grained data traceability using smart contracts

Conclusion and Future Work

In present-day communication networks, telecom provisioning transactions are the core of service delivery, and attention must be paid to the critical coordination among BSS, OSS, NMS, network elements, and third-party platforms. The activation, modification, and termination lifecycle has stringent dependencies, ordering constraints, and high concurrency, and thus transaction management is very important to deliver consistency, reliability, and proper billing. Distributed telecom conditions, which are heterogeneous, high-volume transactions and partial failures, require robust serialization schemes like locking, timestamp ordering, token-based control, and event-driven/message queue schemes to eliminate conflicts, race conditions, and deadlock. As networks become cloud-native, virtualized, and software-defined, serialization and concurrency control remain core to achieving deterministic behavior, service continuity, and strict SLAs.

In the future, it is possible to develop intelligent, adaptive serialization frameworks that use machine learning to dynamically switch between pessimistic, optimistic, and hybrid concurrency strategies based on real-time system load and potential conflict. Future studies on the blockchain-based design of immutable distributed ledgers and event logs have the potential to improve traceability and confidence in multi-vendor and cross-operator provisioning environments. As the network is evolved to 5G-Advanced, 6G, network slicing, and edge computing, slice-aware cross-domain transaction models and unified intent-based provisioning APIs are increasingly required. Consideration should also be given to further development of automated rollback operations, compensation processes for long-running transactions, and self-healing orchestration to bring network operations to the point of fully autonomous, zero-touch operations.

References

- [1] S. K. Davuluri, V. Challagulla, V. Mudapaka, and U. Konka, "AI-Driven DevOps in Telecommunications: Bridging Predictive Analytics with Continuous Delivery for Network Agility," in 2025 IEEE International Conference and Expo on Real Time Communications at IIT (RTC), 2025, pp. 1–4.
- [2] B. M. Ferzo and S. R. M. Zeebaree, "Distributed Transactions in Cloud Computing: A Review Reliability and Consistency," *Indones. J. Comput. Sci.*, vol. 13, no. 3, pp. 3520–3542, 2024, doi: <https://doi.org/10.33022/ijcs.v13i3.3830>.
- [3] A. R. Bilipelli, "AI-Driven Intrusion Detection Systems for LargeScale Cybersecurity Networks Data Analysis: A Comparative Study," *TIJER – Int. Res. J.*, vol. 11, no. 12, 2024.
- [4] A. Parupalli, "The Evolution of Financial Decision Support Systems : The Evolution of Financial Decision Support Systems : From BI Dashboards to Predictive Analytics," *KOS J. Bus. Manag.*, vol. 1, no. 1, pp. 1–8, 2025.
- [5] S. B. Karri, S. Gawali, S. Rayankula, and P. Vankadara, "AI Chatbots in Banking: Transforming Customer Service and Operational Efficiency," in *Advancements in Smart Innovations, Intelligent Systems, and Technologies*, 2025, pp. 61–81. doi: 10.3233/FAIA251498.
- [6] P. R. Marapatla, "Brand Transformation in Nonprofits : A Technology and Strategic Approach for Lasting Impact," *Sch. J. Eng. Technol.*, vol. 13, no. 9, pp. 738–745, 2025.
- [7] B. R. Ande, "Federated Learning and Explainable AI for Decentralized Fraud Detection in Financial Systems," *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 35s, pp. 48–56, 2025.
- [8] R. J. S. K. Das and Y. Makin, "Behavioral Risk Tolerance in U.S. Retirement Planning Vs. Property Insurance: A Comparative Analysis," *Int. J. Appl. Math.*, vol. 38, pp. 41–70, 2025.
- [9] S. K. Tiwari, "Quality Assurance Strategies in Developing High-Performance Financial Technology Solutions," *Int. J. data Sci. Mach. Learn.*, vol. 05, no. 01, pp. 323–335, 2025, doi: 10.55640/ijdsml-05-01-26.
- [10] S. R. Kurakula, "Cloud-native microservices in financial services: Architecting for scalability and flexibility," *World J. Adv. Res. Rev.*, vol. 26, no. 02, pp. 1435–1442, 2025.
- [11] M. A. A. F. N. Al-wesabi, A. T. Zahary, and M. N. Ali, "A Survey of Internet of Things Services Provision by Telecom Operators," *EAI Endorsed Trans. Internet Things*, vol. 4, no. 13, 2018.
- [12] D. Patel, "Enhancing Banking Security: A Blockchain and Machine Learning- Based Fraud Prevention Model," *Int. J. Curr. Eng. Technol.*, vol. 13, no. 06, Dec. 2023, doi: 10.14741/ijcet/v.13.6.10.
- [13] D. P. Putri, "Analysis SIM Card Provisioning Process at Telecommunication Operator using Balance Scorecard," *Int. J. Eng. Res. Technol.*, vol. 3, no. 3, pp. 2107–2112, 2014.
- [14] T. Shah, "Leadership in digital transformation: Enhancing customer value through AI-driven innovation in financial services marketing," *Int. J. Sci. Res. Arch.*, vol. 15, no. 03, pp. 618–627, 2025.
- [15] H. Kapadia and K. C. Chittoor, "Quantum Computing Threats to Web Encryption in Banking," *Int. J. Nov. Trends Innov.*, vol. 2, no. 12, pp. a197–a204, 2024.

- [16] G. Mantha, "Transforming the Insurance Industry with Salesforce: Enhancing Customer Engagement and Operational Efficiency," *North Am. J. Eng. Res.*, vol. 5, no. 3, pp. 1–2, 2024.
- [17] V. Varma, "Data Analytics for Predictive Maintenance for Business Intelligence for Operational Efficiency," *Asian J. Comput. Sci. Eng.*, vol. 7, no. 4, pp. 1–7, 2022.
- [18] M. Fischer, J. Garcia-alfaro, and D. Kesdogan, "Secure Communication Networks and Distributed Systems for a Resilient Society (TC 11 Briefing Paper)," *Comput. Secur.*, p. 104791, 2025, doi: 10.1016/j.cose.2025.104791.
- [19] A. Gherghe, "A Novel Framework for Evaluating Application Performance in Distributed Systems," *Appl. Sci.*, vol. 15, no. 23, pp. 1–33, 2025, doi: <https://doi.org/10.3390/app152312837>.
- [20] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 04, pp. 3557–3564, 2025.
- [21] S. B. Shah, "Advancing Financial Security with Scalable AI: Explainable Machine Learning Models for Transaction Fraud Detection," in 2025 4th International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), 2025, pp. 1–7. doi: 10.1109/ICDCECE65353.2025.11034838.
- [22] Q. Zheng et al., "Lion: Minimizing Distributed Transactions through Adaptive Replica Provision (Extended Version)," arXiv, pp. 1–14, 2024.
- [23] A. Balakrishnan, S. De, and L.-C. Wang, "Distributed Energy Bank Optimization Towards Outage Aware Sustainable Cellular Networks," *IEEE Trans. Sustain. Comput.*, vol. 10, no. 3, pp. 503–514, 2025, doi: 10.1109/TSUSC.2024.3486976.
- [24] M. Guan, J. Zhang, and Z. Zheng, "Blockchain-Based Intelligent E-Commerce Platform for Enhancing Customer Experience in Telecom Operators," in 2024 10th International Conference on Computer and Communications (ICCC), 2024, pp. 2007–2011. doi: 10.1109/ICCC62609.2024.10942227.
- [25] J. Yang, Y. Zhang, Y. Yuan, K. Ma, and J. Li, "Energy Trading and Power Allocation Strategies for Relay-Assisted Smart Grid Communications: A Three-Stage Game Approach," *IEEE Trans. Mob. Comput.*, vol. 22, no. 12, pp. 7438–7450, 2023, doi: 10.1109/TMC.2022.3211408.
- [26] M. Aeri, S. A. Dhondiyal, Y. Rana, S. Rawat, P. Kothari, and R. Adhikari, "Customer Churn Prediction in Telecom Services," in 2023 International Conference on Sustainable Emerging Innovations in Engineering and Technology (ICSEIET), 2023, pp. 809–811. doi: 10.1109/ICSEIET58677.2023.10303463.
- [27] H. Li, P. Gao, Y. Zhan, and M. Tan, "Blockchain technology empowers telecom network operation," *China Commun.*, vol. 19, no. 1, pp. 274–283, 2022, doi: 10.23919/JCC.2022.01.020.
- [28] Y. Pang, D. Wang, X. Wang, J. Li, and M. Zhang, "Blockchain-Based Reliable Traceability System for Telecom Big Data Transactions," *IEEE Internet Things J.*, vol. 9, no. 14, pp. 12799–12812, 2022, doi: 10.1109/JIOT.2021.3138462.