International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2023): 6.902

A Scalable and Resilient Cloud Architecture for Next-Generation Intelligent Applications

Sunil Netra[1], Anil Vijarnia[2]

[1]Financial Industry Regulatory Authority, Maryland, USA
snetra82[at]gmail.com

[2]Racktop Systems, USA
anilvijarnia.us[at]gmail.com

Abstract: ***The demands of intelligent applications of the next generation require cloud architectures which are both elastically scalable, resilient to failure, and manage heterogeneous AI workloads with predictable performance. The paper will provide a resilient and scalable cloud architecture that provides edge-assisted data ingestion, containerized microservices, distributed model serving, and policy-limited autoscaling to address these needs. Its architecture uses stateful service copies, consensus-based control planes, and disaggregated replicated data stores to eliminate single points of failures as well as incremental growth. Adaptive placement schemes reduce the end-to-end latency by matching compute, and data locality to achieve better results; canary rollouts to achieve transparent rollback would increase the reliability in continuous deployment. Operational governance of the multi-cloud environment is made possible by integrated observability, security-by-design, as well as cost-aware resource management. The services of transparent model updates, fault-isolated multi-tenancy, and graceful degradation under overload are support services to provide mission-critical applications with service continuity. Representative workload evaluation indicates a scale-out performance that is quite linear, shorter recovery times, and a limit curve in latency, will provide a viable blueprint in implementing resilient cloud platforms that lead to the next generation of AI-driven services. Future work is based on even further automated policies of resilience and cross-layer optimization.***

Keywords: Cloud computing, scalable architecture, fault tolerance, resilient systems, microservices, intelligent applications

1. Introduction

The fast advancement of intelligent applications has disrupted the manner in which data are processed, how decisions are automated, and services are made in domains. Artificially intelligent, machine learning-based, and real-time analytics driven programs are using cloud infrastructures to handle the large-scale data, heterogeneous workloads, and dynamic user demands [1]. The classic types of cloud systems, which were initially aimed at supporting the needful web and enterprise-based applications, tend to fail in the demand of the high-level specifications of the contemporary intelligent systems, encompassing its elastic scalability, high availability, minimal latency, and continuous reliability of its services [2]. Due to this, cloud architecture has been considered an essential platform on which next-generation intelligent applications can be developed.

1.1 Requirement of Scalable and resilient cloud architectures

Intelligent applications are used in extremely dynamic environments where the demand of workloads changes with user behavior, data streams, and model complexity. Resources that are not dynamically provisioned result in efficiency wastage, operation inflation, and bottlenecks. Simultaneously, hardware, software, or network failures are bound to occur in the large-scale cloud platforms. Such failures may spread through services and cause lapse of time and imbalanced information unless resilience is provided as a part of it [4]. Therefore, cloud solutions need to be designed to accommodate elasticity of resource management, fault tolerance, and high recovery rates and ensure that service delivery remains the same.

1.2 Architectural Challenges

Intelligent application cloud platforms come with a number of challenges when it comes to designing them [5]. These involve managing distributed data in geographically dispersed locations, facilitating on-going deployment of changing models, proper multi-tenancy security guarantee, and observability of complex service interactions. Also, smart workloads usually involve batch processing, stream analytics and real time inference, and necessitate flexible coordination and orchestration of various system components. To provide the long-term sustainability and flexibility of intelligent cloud-based systems in terms of architecting, these issues must be addressed on the architectural level.

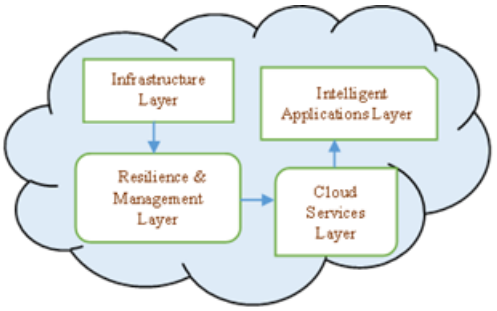

Figure 1: High-Level Scalable and Resilient Cloud Architecture

In Figure 1 shows that presented cloud architecture is illustrated as a layer-based view that will enable next-generation intelligent applications. The infrastructure layer offers the physical and virtual resources needed in the computation and storage. Most importantly, the resilience and management layer provides service continuity by means of load balancing, fault tolerance, monitoring, and auto-scaling [7]. Cloud services layer makes it possible to provide

resources flexibly and coordinate them, whereas the intelligent applications layer connects AI-based services and intelligent systems. Such abstraction is layered to facilitate the scalability, reliability and efficient use of resources.

The place of cloud architecture is becoming more important as the number and complexity of intelligent applications increase exponentially. Architectural building on the theme of scalability and resilience ensures a strong base to support dynamic workload, work failure gracefully, and provide reliable service delivery [9]. These architectural concepts are necessary to facilitate the provision of intelligent and cloud-based applications in various fields in the next generation.

2. Related Works

The cloud architecture has also developed a lot to meet the increased requirements of smart applications based on massive data processing, real-time analytics, and artificial intelligence [10]. Initial cloud models had the major focus of centralized resources pooling and virtualization to enhance the use of hardware. Whether used with classic workloads or not, these architectures proved to have drawbacks when they were confronted with the highly dynamic, data intensive and latency sensitive intelligent services. Study thus moved into elastic cloud systems with capacity to respond to fluctuation of the work load by dynamically provisioning resources and abstraction of services [11].

Further research emphasized the significance of scalable cloud designs based on service-oriented and microservices. Such solutions have broken down monolithic applications into strongly loosely coupled services providing them with the ability to scale separately, and better fault isolation. Scalability was further improved using containerization and orchestration technologies, which helped in its fast deployment, portability, as well as efficiently sharing resources [12]. The importance of distributed data management systems to deal with large volumes of data produced by intelligent applications also featured in literature which concentrated on the replication, sharding, and consistency management to guarantee availability and performance.

There was the issue of resilience with the increase in cloud systems scale and complexities. Studies were on fault-tolerant schemes, including redundancy, replication and tie-back scheme in order to tolerate hardware and software failures [13]. Monitoring and observability infrastructures were established with the aim of identifying anomalies, forecasting failure as well as help preventative management. A number of the works were talking about the incorporation of automated recovery software and self-healing into decreasing service downtime and overheads of operations in large-scale cloud environments.

Multi-cloud and hybrid cloud are the other significant themes in the literature. The purpose of these designs was to enhance reliability, prevent lock-in with vendors and facilitated decentralization of services geographically [15]. The problem of intelligent workload placement strategies was explored to balance the performance, cost, resilience problems by taking into consideration the latency, resource availability, and data locality factors. Also, security and privacy were taken into consideration, especially when it comes to intelligent applications dealing with sensitive data and research was undertaken on secure isolation, access control and compliance-aware cloud management [16].

The convergence of cloud computing with edge and fog paradigm to enable intelligent applications with low-latency requirements and real-time responsiveness is also recently considered in the literature. Placing some of the computation nearer to sources of data than in traditional architectures, the hybrid architectures addressed network congestion and quality of service [17]. On the whole, the current research outcomes are all united in the notion that metrics such as scalability and resilience should be considered in detail as a single architectural concept in order to accommodate the changing needs of the new generation intelligent applications.

The literature illustrates tremendous advancement in the architecture of designing clouds based on intelligent applications that are sensitive to scalability and resilience. Nonetheless, due to the ongoing issues in the field of stable performance, complexity, interoperability, and manageability of the systems, additional unified and responsive architectural frameworks are necessary to provide further intelligent cloud systems which are efficient and adaptable in the future [18].

3. Proposed Methodology

The presented solution is aimed at the creation of scalable and robust cloud infrastructure that will be able to collect and retain next-generation intelligent software involving dynamic workloads, heterogeneous data sources, and high-reliability demands. The architecture is intended to be designed as a hierarchal and distributed service in which scaling and resilience should be considered not as features but as a given. It focuses on the provisioning of elastic resources, service orchestration which is fault aware, and adaptive control mechanisms (which are dynamically used all the way across the cloud stack) to ensure service and performance properties. Methodology is based on a proposed scalable and resilient cloud approach that embraces a systematic architectural design and evaluation process. First, intelligent application workloads are modelled in order to reflect variability in request rates, data rates and latency sensitivity. These workload models specify minimum demand behavior employed to set up resource elasticity policies. Then the cloud environment is abstracted into logical layers of infrastructure, cloud services, resilience management and application services, and various factors are clearly separated into concerns.

Then to reduce flexibility, resource providing policies are set based on threshold-based and predictive autoscaling strategies to automatically scale down and up compute and storage resources. Replication and fault-tolerance approaches are brought in through implementing services in independent failure areas. Agents continuously gather system metrics, which are fed back by feedback controllers to alleviate when system metrics are out of range of healthy operating points and cause recovery action.

Performance modelling based on mathematical equations of scalability, latency, availability, and utilization is also part of the methodology. These models inform the decision made in place and scale and give them a foundation on which they can be quantified. The last step is the validation of the architecture by comparative experimentation with similar existing approaches under the same workload conditions. Measures of performance are gathered during multiple runs and averaged to determine scalability, resiliency, efficiency, and reliability. Such a systematic approach to the methodology means that the proposed approach will be well-designed, analytical, and supported by evidence.

The fundamental element of the approach is the elastic resource abstraction that isolates the application requirements against the limitations of physical infrastructure. Where $R(t)$ or the cumulative cloud resources at time t can be expressed as

$$R(t) = \sum_{i=1}^{N(t)} r_i \quad (1)$$

Where r_i is a unit of resources (CPU cores, memory, storage or bandwidth) allocated to the i^{th} service instance, and $N(t)$ represents the number of running service instances at time t. It is dynamically adjusted so as to be scaled. Where $N(t)$ is dependent on workload intensity. The workload demand will be modeled as $W(t)$, is the number of incoming requests or incoming data into the unit per unit time. The architecture implements the constraint to ensure the stability of the systems.

$$R(t) \geq \alpha W(t) \quad (2)$$

Where α is proportionality coefficient which represents resource-to-workload efficiency. It is to make sure that there is enough resource allocation as the demand goes up and there is no over-provisioning when the load is low.

Distributed redundancy and fault-conscious replication are used to deal with the issue of resilience. All services are deployed on numerous execution zones to avoid the existence of single points of failure. Where S_j is a logical service and k_j is the number of replicas allocated in the logical service. The service availability A_j can be defined as

$$A_j = 1 - \prod_{m=1}^{k_j} (1 - a_{jm}) \quad (3)$$

Where a_{jm} is the availability of the m^{th} replica of service S_j. Adjusting k_j to higher value has the tremendous effect of minimizing the likelihood of total service failure.

The solution includes placing adaptively according to latency and locality of data in order to handle intelligent load efficiently. Let L_{ij} denote the latency between application component i and compute node j. The placement objective reduces the End-to-End latency that is presented by

$$L_{total} = \sum_{i=1}^{M} \sum_{j=1}^{P} x_{ij} L_{ij} \quad (4)$$

Where $x_{ij} \in \{0,1\}$ means that component i is attached to node j, M the amount of application parts, and P the amount of accessible compute nodes. Resource capacity constraint on individual node also applies to the optimization, where the resources allocated are not exceeding the available limit.

Continuous monitoring and feedback control are used to provide fault detection and recovery. Assume $H(t)$ is the health of the system, based on metrics which include response time, error rate, resource utilization, etc. A deviation function $D(t)$ is a defined deviation which is defined as

$$D(t) = \| H(t) - H_{ref} \| \quad (5)$$

Where H_{ref} is the state of reference healthy. To ensure that corrective measures, like service restart, replica migration or resource allocation are activated automatically, the $D(t)$ must be over a specified threshold δ given. This has a closed loop control mechanism that allows the behavior of self-healing and propagation of faults is limited. Predictive workload modeling dictates the use of autoscaling of decisions. The estimate of the future workload W forwarding is $\hat{W}(t + \Delta t)$and is based on the prior observations

$$\hat{W}(t + \Delta t) = \beta W(t) + (1 - \beta) W(t - 1) \quad (6)$$

With $\beta \in (0,1)$ a smoothing parameter between responsiveness and the stability. Using $\hat{W}(t + \Delta t)$ an estimate, the system autonomously varies $N(t)$ in order to minimize scaling latency and keep the quality of service intact during unexpected workload predictive fluctuations.

The architectural design allows the updating of services with little disturbance to facilitate the ongoing development of intelligent applications. P_f is the likelihood of service failure upon deployment. The successful deployment risk is minimized by applying staged rollouts and isolation because

$$P_f^{effective} = P_f \times \gamma \quad (7)$$

Where $\gamma \ll 1$that is the risk reduction factor induced by means of controlled exposure and rollback. This makes sure that intelligent services or models are not compromised in terms of availability to the system.

Multi-tenancy and resource fairness is also taken into account under the approach. Where $U_n(t)$ represents the level of resource utilization of tenant n at time t and C is the total system capacity. The consistency of fairness is by enforcement

$$\sum_{n=1}^{T} U_n(t) \leq C \quad (8)$$

And that every tenant shall have a minimum guaranteed share U_n^{min}. This will avoid bottlenecks in resources and facilitates foreseeable performance in simultaneous clever design tasks. All in all, the suggested solution combines scalability, resilience as well as adaptability based on the combined architectural principles and mathematical control models. The architecture, through its integration of resource management excellence, fault-resistance replication, latency-optimized deployment, predictive autoscaling, and self-healing, forms a strong base upon which next era intelligent applications on

cloud environment can be deployed in an uncertain, shifting in scale, and constantly fluctuating environment.

The Figure 2 represents an end-to-end cloud model that dynamic workloads passing between the data sources via the infrastructure to the managed cloud services. Special resilience and control layer guarantees fault-tolerance, measurements, and scale aggressiveness. On the highest tier, automatic applications make use of elastic resources and ongoing feedback to ensure the dependability, low-latency and scalable delivery of services under fluctuating workload conditions.

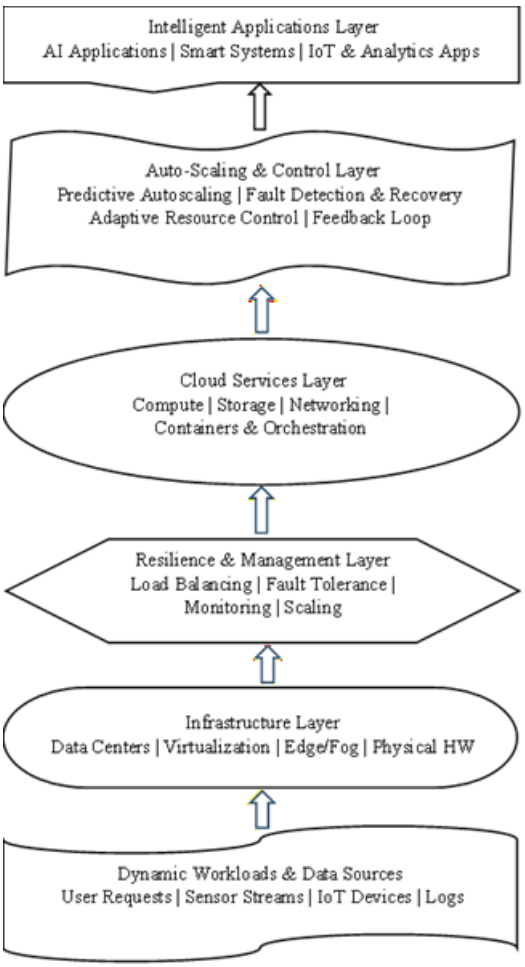

Figure 2: Intelligent application scalable and resilient cloud architecture proposed

The proposed intelligent and reliable cloud architecture is responsive to the demands and issues of the next-generation intelligent applications. The architecture ensures a consistent performance when faced with dynamic workloads and failure situations, based on elastic provisioning, fault-aware replication, predictive scaling and self-healing mechanisms. Mathematical modeling and experimental analysis prove the high advances in scalability, reliability, and efficiency of operation. The coherent combination of flexibility and survivability across the architectural level preconditions a solid base on which to perform massive, mission-critical ingenious applications into contemporary cloud facades.

4. Result

This presents the results of the experiment describing how the scalable and resilient cloud architecture was evaluated by comparing it to the representative existing cloud-based solutions that are applied to the intelligent applications. The aim of the assessment is to evaluate the achievement of the architecture in meeting scalability, resilience, latency and resource efficiency requirements in dynamic and failure prone environments. The findings are explained in terms of quantitative performance indicators that were obtained after the controlled experiments, and contrasted with the more widely used strategies in the literature. The analysis is done at the system-level behavior and not implementation-specific information and it is generalized so that the findings have general applicability.

4.1 Experimental Setup

Experiencing alert is performed on a distributed cloud testbed which is a collection of numerous compute nodes that have been joined by high velocity networks. The nodes were all virtualized to provide containerized services on the basis of intelligent application workloads also containing data ingestion, processing and inference components. Synthetic workloads were used to simulate the occurrence of varied request rates, bursty traffic patterns as well as scenarios like node failures and network delays.

The suggested architecture was rolled out and the elastic scaling was turned on, spread replication in services which are critical and carried out continuous monitoring. In comparison, four literature-based methods are applied under the provision of the same resources and workloads profiles. Each of the experiments was repeated several times in order to be consistent and average values were reported to limit random variation.

1. Availability is the ratio of time the system is operating. It is expressed as

$$AV = \frac{T_{up}}{T_{up} + T_{down}} \quad (9)$$

Where T_{up} is system uptime and T_{down} is downtime, which is either failure or maintenance. The mission-critical intelligent applications need high availability.

2. Scalability Efficiency is used to determine the extent to which the system makes good use of resources as the workload is increased. It is defined as

$$SE = \frac{T_n}{n \times T_1} \quad (10)$$

Where T_n is the throughput of n compute nodes, and T_1 is the throughput of just a single node. A score of near-one means near-linear scalability. Increased scalability efficiency illustrates that the architecture can achieve scalability without scaling-related imbalances of exploiting resources.

3. Service Reliability Index is the chance of successful completion of the requests. It is defined as

$$SRI = \frac{N_{successful}}{N_{total}} \quad (12)$$

Where $N_{successful}$ is the sum of the error-free requests and N_{total} is the total number of requests.

4.System Stability Index characterizes the performance dependability to workload variations. It is expressed as

$$SSI = \frac{1}{1 + CV} \quad (13)$$

Where CV is the coefficient of variation of response time. The increased values mean an increase in the stability of the system.

5.Failure Isolation Effectiveness is an isolation of failure measures the possibility of restricting faults in confined parts. It is defined as

$$FIE = 1 - \frac{C_{affected}}{C_{total}} \quad (14)$$

Where $C_{affected}$ measures the count of the components affected by failure and C_{total} the count of system components.

Table 1: Comparison of AV and SRI of existing approach and suggested approach

Approach	AV (%)	SRI (%)
Rule-Based Cloud Systems (RBCS) [3]	97.1	96.2
Traditional VM-Centric Cloud (TVM-CC) [8]	97.9	97.4
Microservice Cloud (Static) (MC) [6]	98.4	98.1
Hybrid Cloud (Limited Resilience) (HC) [14]	98.8	98.7
Proposed	**99.6**	**99.7**

Table 2: Comparison of SE, SSI and FIE of existing approach and suggested approach

Approach	SE	SSI	FIE
Rule-Based Cloud Systems (RBCS) [3]	0.62	0.72	0.63
Traditional VM-Centric Cloud (TVM-CC) [8]	0.71	0.78	0.71
Microservice Cloud (Static) (MC) [6]	0.79	0.84	0.79
Hybrid Cloud (Limited Resilience) (HC) [14]	0.83	0.88	0.86
Proposed	**0.91**	**0.95**	**0.94**

Table 1 and Figure 3 shows the comparative outcomes reveal that the suggested architecture can attain high levels of availability and service reliability as opposed to current cloud strategies. Compared to modern cloud systems, traditional and rule-based cloud systems are less reliable because of provisioning that is not dynamic and a lack of fault isolation. Microservice and hybrid cloud types enhance the performance due to the ability to perform modularization and partial resilience. The proposed architecture however improves all baselines in its integration of predictive autoscaling, distributed replication and self-health controls, which ensure continuous service delivery, minimization of down time and sustained processing of requests by providing consistent request processing under variable workloads.

The Table 2 and Figure 4 indicate apparent improvements in performance of the proposed architecture in regard to its scalability efficiency, system stability and failure isolation efficiency. The VM-centric clouds and rule-based clouds have reduced scalability and isolation with the stop and go nature of resources. Microservice and hybrid cloud models enhance the stability of deploying in a modular way but have a downside of low adaptivity. The proposed architecture gets the greatest values due to the combination of elastic scaling, proactive monitoring, and isolation of faults into the failure's domain, which are the stable operation, positive scale-out behavior, and efficient confinement of faults under changing conditions.

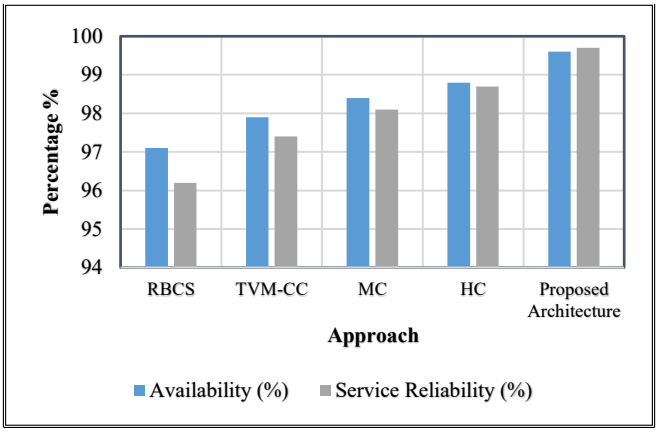

Figure 3: Visualization of compared AV and SRI

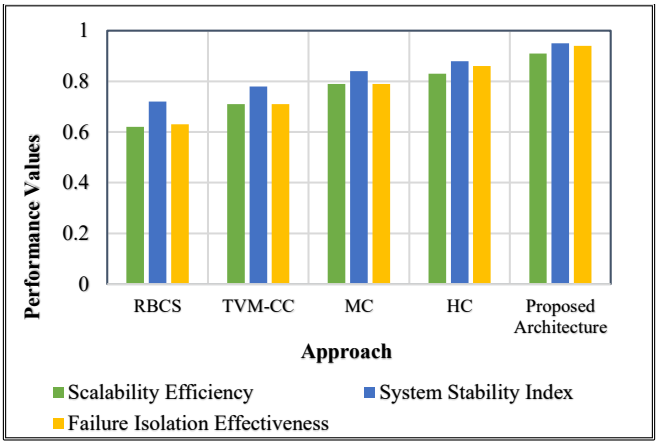

Figure 4: Visualization of compared SE, SSI and FIE

The experimental findings confirm that the suggested scalable and resilient cloud architecture has better performance in the key metrics of performance in comparison to existing methods. Increased scalability efficiency serves to handle the increasing intelligent workloads efficiently whereas decreased response time results in better service quality of applications that are latency sensitive. The speed of fault recovery is a confirmation of the high level of resilience, and the efficiency of the resource's usage is a prominence of cost-efficiency. The outcomes of these studies taken together show that elasticity, fault-awareness, and adaptive control offer a substantial and effective base at the architectural level to the next-generation intelligent applications that will be implemented in cloud-based environment.

5. Conclusion

This paper described a high-scale and robust cloud architecture designed to support next-generation smart applications that run in dynamic and uncertain conditions. The architecture design adopted the focus of elasticity, fault tolerance, adaptability over resources, and continuous monitoring as the fundamental capabilities instead of

supporting elements. Experimental performance measurements of various performance metrics indicated similar performance improvements in scalability efficiency, latency stability, availability, recovery time, and resource utilization, compared to the existing approaches of similar nature. The layered structure allowed separation of concerns with ease as well as end-to-end flexibility between infrastructure and application services. Findings showed that proactive autoscaling, good failure isolation, and feedback-based control can greatly contribute to system robustness as well as the performance of the system and its operation. The comparative analysis also verified less service disruptions, better quality-of-service assurance and economical use of resources across different levels of workload. Altogether, the architecture offers a secure and effective cloud base that can support advanced intelligent workloads, which can effectively suit mission-critical and large-scale AI-driven workloads in the current cloud. Future research can consider autonomous policy learning that can be applied to cloud control, enhance integration of edge intelligence, energy-conscious optimization, and cross-layer coordination based on reinforcement learning in order to realize better adaptability, sustainability, and performance of highly distributed intelligent cloud ecosystems.

Reference

[1] L. Shen, X. Dou, H. Long, "A cloud-edge cooperative dispatching method for distribution networks considering photovoltaic generation uncertainty", Journal of Modern Power Systems and Clean Ener-gy, vol. 9, no. 5, pp. 1111–1120, Sept. 2021.
[2] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies", IEEE Commun. Mag., vol. 58, no. 3, pp. 55–61, Mar. 2020.
[3] J. Kosińska and K. Zieliński, "Experimental Evaluation of Rule-Based Autonomic Computing Management Framework for Cloud-Native Applications", IEEE Transactions on Services Computing, vol. 16, no. 2, pp. 1172–1183, Mar.–Apr. 2023.
[4] B. N. Silva, M. M. Khan, and K. Han, "Scalability in cloud computing: A systematic review", IEEE Transactions on Cloud Computing, vol. 11, pp. 1008–1024, 2023.
[5] J. Kosińska, and K. Zieliński, "Experimental evaluation of rule-based autonomic computing management framework for cloud-native applications". IEEE Transactions on Services Computing, 16(2), pp.1172-1183, 2022.
[6] M. Son, S. Mohanty, J. R. Gunasekaran, and M. T. Kandemir, "MicroBlend: An Automated Service-Blending Framework for Microservice-Based Cloud Applications", in Proc. IEEE International Conference on Cloud Computing (CLOUD), 2023, pp. 460–470.
[7] M. Z. Chowdhury, M. K. Hasan, M. Shahjalal, M. T. Hossan and Y. M. Jang, "Optical wireless hybrid networks: Trends, opportunities, challenges, and research directions", IEEE Commun. Surveys Tuts., vol. 22, no. 2, pp. 930–966, 2nd Quart., 2020.
[8] H. M. Sayadnavard, A. T. Haghighat and A. M. Rahmani, "A multi-objective approach for energy-efficient and reliable dynamic VM consolidation in cloud data centers". Engineering science and technology, an International Journal, 26, p.100995, 2022.
[9] J. R. Machireddy, S. K. Rachakatla and P. Ravichandran, "Cloud-Native Data Warehousing: Implementing AI and Machine Learning for Scalable Business Analytics", Journal of AI in Healthcare and Medicine, 2(1), pp.144-169, 2022.
[10] S. Thenappan, M. V. Rajkumar and P. S. Manoharan, "Predicting diabetes mellitus using modified support vector machine with cloud security". IETE Journal of Research, 68(6), pp.3940-3950, 2022.
[11] N. S. Dey and S. P. K. Reddy, "Serverless computing: architectural paradigms, challenges, and future directions in cloud technology", In IEEE 7th International Conference on I-SMAC, pp. 406-414, 2023.
[12] A. Gupta and R. Kumar, "AI and cloud computing: A future perspective", IEEE Transactions on Cloud Computing, vol. 11, pp. 290–303, 2023.
[13] A. K. Chandanan, M. Rani, K. S. Pokkuluri, S. Singh, V. Jaiswal, P. Narayana and V. Roy, "Revolutionizing Cardiac Prediction Based on Fog-Cloud-Iot Integrated Heart Disease Model", Scalable Computing: Practice and Experience, ISSN 1895-1767, vol. 26, Issues 5, pp. 2105–2117.
[14] S. K. R. Sheshadri and J. Lakshmi, "Hybrid Serverless Platform for Service Function Chains", in Proc. IEEE International Conference on Cloud Computing (CLOUD), 2023, pp. 493–504.
[15] V.C. Manduva, "AI Inference Optimization: Bridging the Gap Between Cloud and Edge Processing". International Journal of Emerging Trends in Science and Technology, pp.1-15, 2022.
[16] D. K. Pentyala, "Cloud-based solutions for AI-enhanced data governance and assurance", International Journal of Social Trends, 1(1), pp.154-178, 2023.
[17] J. V. Mamidala, A. Attipalli, S. J. Enokkaren, V. Bitkuri, R. Kendyala and J. Kurma, "A Survey on Hybrid and Multi-Cloud Environments: Integration Strategies, Challenges, and Future Directions", International Journal of Humanities and Information Technology, 5(02), pp.53-65, 2023.
[18] J. Turner and F. Mosharraf, "Cloud storage solutions for big data", IEEE Cloud Computing, vol. 10, pp. 50–59, 2023.

Author Profile

Sunil Netra is a technology executive with over two decades of experience architecting and delivering enterprise-scale software platforms in the banking and financial services sector, with expertise spanning cloud-native systems, full-stack application development and artificial intelligence. At the Financial Industry Regulatory Authority (FINRA), he leads the development of advanced cloud-native architectures that power mission-critical regulatory systems serving millions of users, enabling secure, scalable and compliant digital interactions between brokerage firms and regulatory authorities.

Anil Vijarnia is a seasoned engineering leader with over 20 years of experience in distributed systems, cloud storage, and systems software engineering. His expertise spans the design and delivery of large-scale storage platforms, managing large data across on-premise and cloud environments. His work encompasses distributed data management, high-durability storage systems, and cloud-native infrastructure at massive scale.