International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

A Scalable Cloud Architecture for Intelligent Automation in Distributed IoT Sensor Networks for Semiconductor Test Environments

Anumandla Mukesh

Independent Researcher, Malaysia

mukeshanumand@gmail.com

Matteo Ricci

Academic Research Assistant, Verona Institute of Digital Sciences, Italy

matteorricci@outlook.com

Publication History: Received: 01.07.2026; Revised: 27.07.2026; Accepted: 01.08.2026; Published: 11.08.2026.

ABSTRACT: Cloud-based smart automation architecture is proposed for managing a distributed network of low-cost, heterogeneous, redundancy-provisioned Internet-of-Things (IoT) sensors deployed in semiconductor chip testing environments. Technology discovery research is guided by the need for continuous physical experimentation of automated testing procedures in dedicated engineering laboratories that require expensive chip-testing equipment, not always available at use time. Chip test automation without on-demand access to the dedicated equipment is also pursued during University Business and Industry collaboration, being made available when dedicated equipment for traditional tests is either busy being operated, maintained, or under repair. However, such dedicated equipment has not been made available for node testing.

The architecture comprises edge-cloud components, and these interact with a sensor network that includes a large number of devices for monitoring a variety of physical variables. Process control and monitoring devices connected to the equipment operate under an edge-cloud architecture; processing piles of data, and supervision and control signals are exchanged with the cloud. At an appropriate processing stage, data is available for incorporation into an IoT-sensor constellation and is distributed to support Smart City and other applications. The design decisions and technical choices assist several cloud management and automation functions and address requirements and constraints such as service level agreement compliance, data quality and governance, device onboarding and admission control, action triggering and execution, plug-in/play capability, process and playback recording, and budget-friendly maintenance of heterogeneous multi-device clouds.

KEYWORDS: Cloud Computing; Edge Computing; Internet of Things; Automation; Multi-Tenancy; Data Management; Data Analytics; Device and Resource Orchestration; Cloud-native automation; Distributed IoT sensor networks; Smart manufacturing systems; Semiconductor chip testing; Edge–cloud orchestration; Real-time data analytics; Automated test equipment (ATE) integration; Scalable microservices architecture; Predictive fault detection; Industrial IoT (IIoT) security and reliability.

I. INTRODUCTION

Semiconductors are crucial to modern society, yet chip testing represents a major bottleneck. This work addresses the following research question: How can cloud-based automation enhance the efficiency of chip-testing operations in real life? Although the intersection of IoT-sensor networks, edge-cloud computing, and automation has been the subject of attention in recent years, generating considerable interest in literature, a comprehensive architectural blueprint remains elusive. To address this gap, a cloud-based smart automation architecture for distributed IoT-sensor networks in chip-testing environments is proposed.

The architecture is designed to ensure seamless interaction across four distinct layers: management and orchestration, data management, automation mechanisms, and security and compliance. An application scenario from semiconductor

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

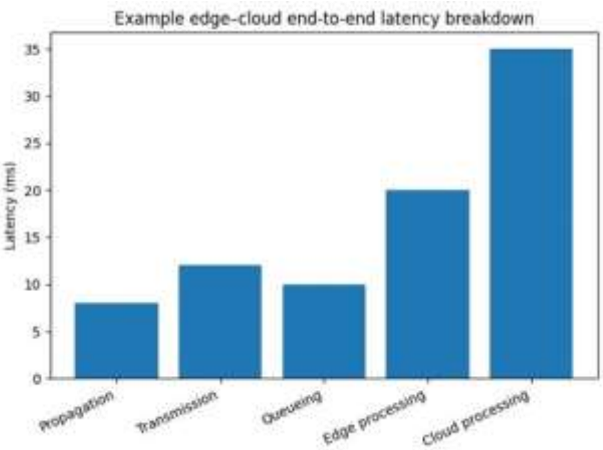
testing serves as a practical use case. The goal of cloud-based smart automation is to enable IoT-sensor networks in chip-testing environments to carry out complex and smart tasks without human intervention, thus reducing operational costs and maximizing resources. Automation capabilities are designed to allow the architecture to complete tasks in a repeatable, faster, and error-free manner with minimal supervision. In addition, a workflow-engine solution occupies a prominent role in architecture, allowing the chip-testing process to be orchestrated and automated in a flexible manner.

Fig 1: An IoT-Based Smart Home Automation System

II. BACKGROUND AND MOTIVATION

Research on cloud-based smart automation of distributed IoT-sensor networks has evolved rapidly, especially in application domains with strict operating and reporting constraints, such as semiconductor fabrication plants. However, the nature of the edge-cloud paradigm and the business logic—to what data should the edge layer send data, and when—has not been systematically studied. Therefore, a cloud-based smart automation architecture enabling such thematic analysis and service across these sensor nodes is proposed. Such automation requires a framework-first approach, focusing on data management and analytics, automation mechanisms, and associated face and cross-cut policies. Emerging application-specific workflows are presented, along with architecture-centric enabling conditions.

In semiconductor fab environments, a critical operation is the testing of fabricated chips (i.e., testing chips after disabling the test logic). Such testing operations have very stringent latency requirements. The implementation of a cloud-based smart automation architecture with an associated IoT-centric thematic analysis of the chip-testing environment can simplify this complex testing operation and help speed up the decision-making process, including prevention. Furthermore, these two operations are becoming more and more complex as operations such as predictive maintenance are getting added on top. A suitable natural fit for such a setting is cloud-based smart automation.



| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

III. SYSTEM ARCHITECTURE OVERVIEW

Overall Architectural Framework
The proposed architecture operates at three complementary levels. The first layer comprises distributed sensor networks that monitor the physical environment surrounding chip testing equipment. The second layer encompasses an edge-cloud computing system that collects sensor data, issues control commands, and orchestrates overall operations. The third layer provides cloud-based data management and analytics services that support chip testing activities across different time scales. Each layer is equipped with distinct elements and performs different roles, yet all contribute to delivering smart cloud-based automation.

The design is driven by specific goals, constraints, and principles. The architecture must enable real-time, policy-driven, and workflow-driven automation of chip testing as well as provide data management and analytics services for topics such as predictive maintenance. The network of communicating sensors must support a variety of data acquisition tasks while remaining resilient to short-term disruptions. Furthermore, the architecture must ensure that cloud-based functions for data storage, processing, and analysis are cost-effective, secure, and compliant with laws and regulations. These demands guide the design and implementation of the architecture while offering a basis for future refinements and extensions.

Interactions between edge and cloud computing functions also shape the architecture. Sensor data are constantly transmitted to the cloud and used for data storage and analytics. Periodically, the cloud sends control signals to the edge—typically low-bandwidth commands with relatively tight time limits—that trigger actions in the environment being monitored. Only those signals with longer-than-usual latency requirements are offloaded for direct cloud management. Edge computing resources offer a natural interface for preprocessing data streams, executing local decision loops, and implementing low-latency feedback control.

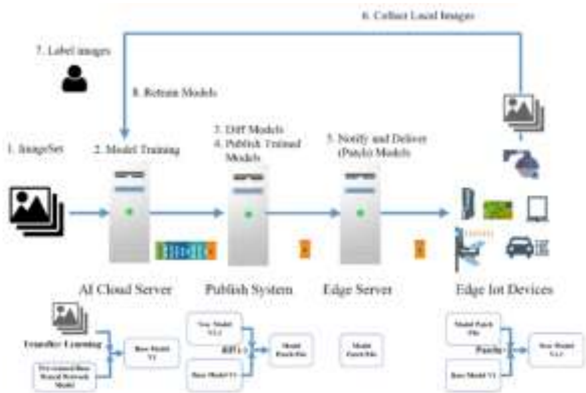


Fig 2: A Cloud-Edge-Smart IoT Architecture

3.1. Edge-Cloud Interaction: User keywords: 'response times', 'network delay', 'data processing times', 'edge-cloud interaction', 'travel distance', 'temporary data storage', 'preprocessing', 'control signals', 'data offloading', 'edge processing'

Three types of latency affect cloud-edge interaction: response times between the edge and the cloud, overall network delay (summation of transmission delays and data processing times), and travel distances covered by control signals issued from the cloud and by the associated actions. Ideally, sensitive control signals are carried by the cloud-to-edge paths, while less delicate ones may travel through the edge-to-cloud connections. Data offloading is governed by latency requirements from sending devices to the processing endpoints. If the cloud cannot provide timely decisions for local control loops requesting edge-related services, not strictly bound to the cloud, a portion of the computation may be moved beside the service edge for fast response and/or network optimization. Preprocessing gear deployed at the edge, such as noise filtering and data cleansing, is regularly performed to reduce data load toward the cloud. Temporary storage buffers can also be used to accumulate data until signal patterns triggering early detection scenarios are detected.

The responsibility of the edge computing element located on the edge-cloud boundary thus encompasses border management and edge support for safety-critical automation scenarios. By consolidating support for these two aspects in

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

a single equipment, costs can be minimized while preserving the typical advantages of distributed architectures, such as quick responses to safety-critical control loops, minimization of control signal delays, and reduction of local volume synthesis and local area network traffic for high-frequency, low-volume signals.

Equation 1: Edge–cloud latency equations (derived from the paper's definitions)
The paper explicitly describes overall network delay as a **summation** of transmission + processing times.
Let:

- D_{net} = overall network delay
- D_{tx} = transmission delay
- D_{proc} = processing delay

Step by step:

1. Start from the paper's definition ("summation of transmission delays and data processing times"):

$$D_{\text{net}} = D_{\text{tx}} + D_{\text{proc}}$$

2. Expand transmission delay into a basic link model (standard comms modeling):

- message size S (bits)
- link bandwidth B (bits/sec)

$$D_{\text{tx}} = \frac{S}{B}$$

3. Expand processing delay into typical stages (queue + compute):

- D_q queue/waiting
- D_c compute time

$$D_{\text{proc}} = D_q + D_c$$

4. Substitute back:

$$D_{\text{net}} = \frac{S}{B} + (D_q + D_c)$$

That's the "complete" operational equation consistent with the paper's statement.
The paper references "response times between the edge and the cloud". For a request–reply to interaction:
Let:

- RT = response time seen by the caller
- RTT = round-trip network time (there + back)
- D_{edge} = processing at edge
- D_{cloud} = processing at cloud

Step by step:

1. Response time is network round trip + processing:

$$RT = RTT + D_{\text{edge}} + D_{\text{cloud}}$$

2. If you want RTT in terms of components (one direction propagation + transmission + queue), you can write:

$$RTT \approx 2\,(D_{\text{prop}} + D_{\text{tx}} + D_q)$$

So:

$$RT \approx 2\left(D_{\text{prop}} + \frac{S}{B} + D_q\right) + D_{\text{edge}} + D_{\text{cloud}}$$

The paper mentions "travel distances covered by control signals". In networking terms, distance mainly affects **propagation delay**:
Let:

- physical distance d (meters)
- effective signal speed v (m/s) (fiber is ~2×10^8 m/s)

$$D_{\text{prop}} = \frac{d}{v}$$

This links the paper's "travel distance" notion to measurable latency.
Related graph + table (already generated above)

- **Bar diagram:** example edge–cloud latency breakdown
- **Table:** latency components + percentage contribution

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

(These are illustrative, not measured from your lab—use your real message sizes and bandwidths to compute actual numbers.)

3.2. IoT-Sensor Network Layer: The described sensor networks constitute the IoT-sensor network layer. Most chip assembly solutions follow the same or similar steps, enabling the implementation of multi-tenant architecture. In this case, a simple topology can be modelled as a star or a mesh using the MQTT protocol with the JSON data format. Device management is implemented using the Calimero library, allowing easy deployment and enabling communication with 868, 915, and 433 MHz Lora WAN networks. Network-device availability is a source of concern since the connection may change; this is mitigated by constantly sensing network availability and services.

Due to the nature of Internet connectivity between the sensor networks and the cloud, with poor Quality of Service (QoS) and not guaranteed reliability in data transmission, the use of buffers is essential. These must not only act as temporary storage for the information being sent but also guarantee that storage maintains the expected sensor refresh rates. An additional analysis of the recent past can help eliminate errors due to faulty nodes that can remain unnoticed for long periods. For data acquisition, suitable data structures are required to qualitatively analyze the data received from the buffer. The data structures mitigate the variability of the number of data vectors sent per time.

3.3. Cloud Management and Orchestration: Cloud services support multiple tenants across diverse devices, assuring high availability, scalability, and multitenancy. Lifecycle management encompasses service instantiation, scaling, updates, and terminations. Leveraging containerization simplifies these operations and enables orchestration workflow automation. Furthermore, cloud management oversees the entire edge-cloud ecosystem, including service provisioning, monitoring, logging, and governance.

Cloud architecture encompasses compute, storage, and network management functions, employing cloud constituents' strategy data. Internal QoS configurations balance service performance with operational costs, while outbound signals establish telemetry retention policies. Events trigger input data processing pipelines, shape automated test workflows, and invoke resource allocation managers. Monitoring signals facilitate fault detection, trigger remediation procedures, and verify SLA compliance.

Cloud services collectively comprise a multi-tenant ecosystem for distributed devices across advanced technological domains. The interaction model coordinates synchronous activities and orchestrates complex event processing and decision-making workflows. Telemetry storage normalizes entries from heterogeneous sources, ensuring quality and supporting timely query execution. Data repository characteristics focus on online analytic processing, enabling real-time alert generation for rapid user remediation.

IV. DATA MANAGEMENT AND ANALYTICS

Well-established principles from database systems, data engineering, and data science inform data architecture design decisions. Ecosystem participation entails leveraging resources available to the data-driven layer. Data products include pipelines for time-series telemetry data, marshalled by analytic readiness. Telemetry data correctness, completeness, and granularity are priorities, guided by accepted principles of quality, provenance, and governance.

The data acquisition pipeline combines acquisition rate capping, sliding-window buffering, quality control measures, and auxiliary data. Updating co-occurring data sources align feature streams radiating from the data product repository. Data storage schemes, indexing for fast retrieval, retention policies governed by data-product lifetimes, and support for time-slice-oriented query primitives enable efficient access to telemetry data.

Real-time analytics encompass streaming-analytic methods using statistical and model-based techniques to extract features from incoming telemetry streams. The machine-learning support data cube includes predictive-maintenance-model definition, training-data collection, and online/ offline servicing of inference requests.

4.1. Data Acquisition and Preprocessing: Telemetric data generally emerges in high-volume and high-velocity streams and therefore require careful tuning of acquisition rates. Since fast and bursty signals might manifest for only certain periods, buffering and cleansing are required. Subsequently, the data need padding, normalization, optional calibration, and automatic metadata capture for better usability during subsequent data analysis. Time synchronization is another

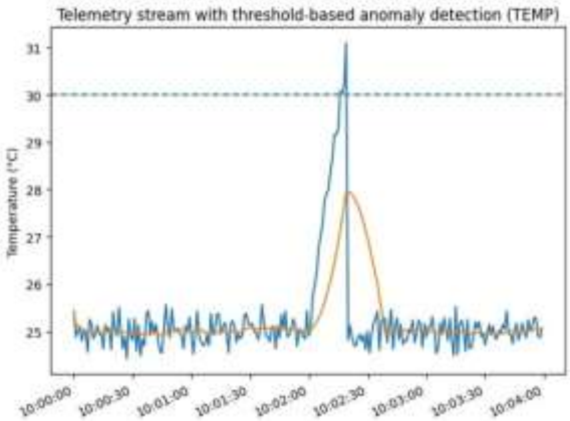
| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

important stage, and sensor nodes should provide suitably synchronized signals with adequately reduced timing discrepancies, with respect to one another, for meaningful analytic functions.

Time-series data acquired from the distributed IoT-sensor network can be viewed as a Quality Control (QC) database since it records the testing procedures, ambient conditions, and environment with respect to the semiconductor chip being tested. Semiconductor Chip testing is done in an enclosed chamber with controlled ambient using humidity, temperature control system and it is that chamber which is smarter with multiple IoT-sensors, edge computing and cloud computing. Hence the storage part for data should include telemetry pertaining to TEMP, HUMIDITY, EFI-values which if cross correlated should give the proper state of Quality of Semiconductor Chip Testing.



4.2. Time-Series Storage and Querying: Supporting time-series storage of telemetry data in extensive IoT-sensor networks requires efficient indexing, compression, and retention. An appropriate data retention policy governs the time-to-live of stored data and balances storage costs with granular historical analytics of long-term data trends. Thus, primitive query capabilities allow for efficient access to stored telemetry in time-series format.

Successive telemetry messages along the same time axis represent the same data stream; indexed storage can eliminate redundant data for storage efficiency. Time-series datastore solutions, such as Influx DB, can provide differencing capabilities along the common source-analyzer-destination time axes, leveraging difference-message ordering under the assumption of slowly varying characteristics across data streams. Fast compound queries using similar keys can also exploit the same principle. Customized implementation of custom-time-series storage using native-time-series capabilities of cloud object storage can provide cost-effective long-term storage with time-series indexing support using the Cloud Data Warehouse service in the cloud provider ecosystem.

Retention policy also governs time-series granularity for long-term access; advanced analytics supporting monthly and yearly reports can appropriately sample and downsize the data before auto archiving into cheap cloud-object storage using the snowball retention concept. Primitive query capabilities permit efficient access to stored telemetry in time-series format. The monitoring team can initiate scheduled retrievals or ad-hoc queries, enabling control over the number and types of devices processed simultaneously to manage query execution load.

4.3. Real-Time Analytics and Anomaly Detection: Streaming analytics for real-time detection of abnormal conditions in chip testing environments have several distinct characteristics. Streamed time-series data from different sensors—such as temperature, humidity, and supply voltage—typically arrive from different devices, and often with distinct scales, units, and noise characteristics. Moreover, the series are usually monitored for anomalous behavior by performing separate checks on each series. For example, an ambient temperature sensor may periodically flag readings above a certain threshold, while a humidity sensor produces alerts when the moisture level drops below a different threshold. In such cases, it is often expedient to express and apply the alerts as threshold functions, as opposed to using more sophisticated anomaly-detection models, even if such models were available.

Provided that the threshold conditions are explicitly defined, they can be converted into streaming computations, supported either natively or through user extensions in stream-processing systems. Thresholds for specific events or operating conditions can be captured through research collaboration with domain experts, provided that the conditions

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

are meaningful and informative. Beyond simple threshold checks, complex event-processing approaches can also be employed, through user-defined functions and subscriptions for monitoring unexpected combinations of temporal patterns across different sensor series.

4.4. Machine Learning for Predictive Maintenance Machine learning has the potential to deliver significant value to system operators by reducing downtime through predictive maintenance. Predictive maintenance is concerned with predicting imminent equipment failure so that necessary maintenance actions can be performed before failure occurs, thus avoiding costly disruptions. The type of model—classification, regression, or sequence-to-sequence—depends on the target variable that indicates impending failure.

The input features used for training the predictive models can be easily generated by cloud-based automation architecture. The ML pipeline transforms data in the training set so that the resultant feature set contains adequate information for the model to learn. Features that indicate failing conditions can be extracted from the telemetry data stream and processed online as part of the machine learning deployment strategy. Features may be computed from sliding training windows, where a new sample is evaluated based on the last N seconds of data. Such a continuous online feature extraction process is useful to monitor the condition of redundant/backup devices too.

Equation 2: Buffering and refresh-rate equations (IoT sensor network layer)
Let:

- sensors refresh rate f (samples/sec)
- sampling period $T_s = \frac{1}{f}$ (sec/sample)
- buffer time horizon T_b (seconds)
- required buffer capacity N (samples)

Step by step:

2. Samples produced in T_b seconds at rate f:

$$N = f\,T_b$$

3. Since $f = 1/T_s$, substitute:

$$N = \frac{T_b}{T_s}$$

4. If each sample payload is p bytes, memory requirement M is:

$$M = N\,p = f\,T_b\,p$$

This is the simplest sizing rule that matches the paper's buffering requirement.

V. AUTOMATION AND ORCHESTRATION MECHANISMS

The architecture supports cloud-based automation of the integrated-chip-testing environment, leveraging an edge-cloud paradigm and an IoT-sensor network for data acquisition and monitoring. Time-series telemetry data streams through dedicated data management and analytics pipelines, enabling real-time alerting and analytics-driven predictive-maintenance capabilities. Additional data created by the automation mechanisms is stored in the cloud.

Automation capabilities are mainly expressed as control loops—continuous telemetry data acquisition drives the automation of day-to-day activities. Mechanisms that are policy-based or workflow-based are emphasized, as both approaches work with the same underlying automation engine.

5.1. Workflow Automation for Chip Testing
Test orchestration automates the execution of predefined test plans—each plan specifies the sequence in which tests are performed, the resources assigned to the test, and the dependencies among tests that must be respected during execution. Orchestration promotes repeatability by controlling the entire execution path, enabling easy rollback in case of failures.

A workflow service accepts incoming requests to activate or deactivate execution plans. Requests can monitor the execution of running plans, receive notifications when plans are complete (even if they fail), or repeat the execution of completed plans. In this deployment, requests originate from an external entity that oversees the automated chip-test facility.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

Executing testing workflows is protected from transient network problems. In the event of a service outage, pending requests are queued for future execution. For such workflows, completion notifications are sent after execution, but failures are logged only locally and not delivered. Dependency-checking conditions are verified when the service resumes operation.

5.2. Policy-Based Resource Allocation

Resource-allocation policies govern the assignment of compute, storage, and network resources in order to meet service-level requirements—for latency, throughput, quality of storage media, and maximum operating costs.

Policy evaluation is performed periodically. Each policy is tested, and its action is executed if the triggering condition is satisfied. For QoS-related policies, the action currently affects only the network bandwidth reserved for edge-cloud connections.

5.3. Scheduling and Load Balancing

Deploying a cloud-based architecture to automate an integrated-chip-testing facility raises the need for allocating computation to edge or cloud based on latency and throughput requirements. Furthermore, workload distribution aims to strike a balance between latency and responsive operation, while fault-awareness helps maintain throughput.

Scheduling is based on priority classes. Each active scheduling group can be assigned a percentage of the total allowed compute capacity of the edge and cloud. Workload distribution decides whether each computation will be triggered at the edge or in the cloud, relying on latency considerations for time-sensitive tasks and semantically similar telemetry-data storage.

Balancing the load across edge and cloud enhances QoS—by keeping computational load within agreed thresholds for latency-sensitive tasks—and maintains throughput even during overload situations.

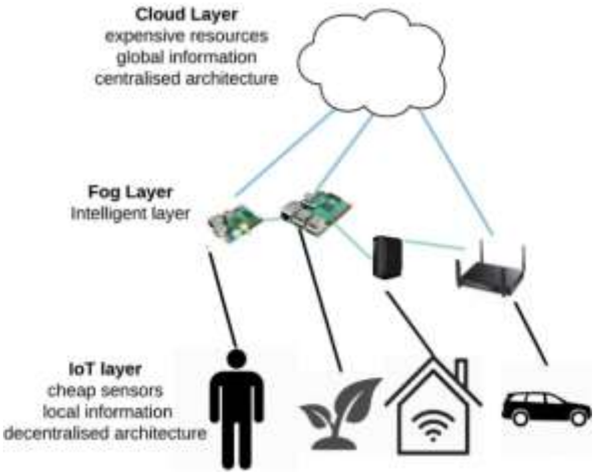


Fig 3: Process Automation in an IoT–Fog–Cloud Ecosystem

5.1. Workflow Automation for Chip Testing: The automation of semiconductor test chips is a real-world application and a potential source of research inspiration. Typically, a semiconductor chip undergoes a lengthy testing process before being released for commercial use. This involves directly injecting various test programs into the chip's memory, with multiple sequential functions assigned to various core modules of the chip. If test failures occur after a function is activated, the test needs to be repeated after some sequential functions are reprogrammed. To alleviate this burden, robust test orchestration, sequencing, dependency management capability, and repeatability guarantee mechanism can be established to greatly facilitate testing. With the appropriate trigger condition, a rollback procedure can be implemented whenever a threshold is exceeded. The flexibility and simplicity of such automation within a distributed IoT-sensor network environment is appealing.

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

The cloud also provides sufficient advantage for automating these scripts by determining the order of execution, managing the orchestration, and guaranteeing the successful completion of all tasks. Proper consideration of the abort mechanism makes this a seamless integration component without introducing significant management overload.

5.2. Policy-Based Resource Allocation: Policies govern allocations of compute, storage, and network resources across the edge-cloud architecture. QoS metrics cater to real-time services, fairness policies prioritize surveillance data, and workload engineering schemes optimize costs during low-activity phases. Policy evaluation mechanisms assess data volumes, processing loads, and retention needs to determine suitable policies. The functionality of the chip-testing orchestration module informs these evaluations.

Cloud-based automation in IoT-sensor networks promises substantial resource-Energy savings compared to current provisioning approaches but also requires careful management to avoid overloading edge nodes. Policy frameworks mediate data-traffic dynamics by specifying resource teams based on latency and throughput targets. For example, QoS policies steer video feeds toward the lowest-latency, lowest-queue-depth team, while checkpoint data gravitate toward the longest-queue team to ensure fair surveillance. Resource-management policies further reallocate excess capacity to edge nodes that transport Heavier data flows. Workload-engineering schemes emphasize economy during low-activity periods by decreasing data-retention within the underlying storage stack and identify windows when prep data may be discarded to minimize operating costs.

5.3. Scheduling and Load Balancing: Dedicated scheduling algorithms assign sensor tasks and allocate resources for execution on the edge or in the cloud. Given the heterogeneous nature of distributed sensor nodes across various chip testing facilities, the time needed to execute user-defined tasks can differ considerably, with some operations consuming a fraction of a second while others may require several hours. The duration of such actions may also fluctuate based on the sensors involved and the environmental conditions. The scheduling framework takes these factors into account by distributing the workload between edge and cloud nodes according to predefined latency or throughput targets. A set of static analysis guidelines further ensures the QoS of the device, shaping the frequency of sensor measurements across both the control and monitoring management functions. Such considerations also enhance data availability, supporting load-balancing policies that guard against downtime due to sensor failure or resamples.

Latency and throughput targets at the network layer are complemented by redundancy requirements at the application layer. For each monitored resource, the system relies on a set of sensors that provide identical measurements. Traffic congestion, packet loss, and device failure all contribute to the variance of real-time telemetry, and the resource monitoring service aims to disperse incoming traffic evenly among the different sensor paths. Internal mechanisms address situations where selected sensors become unresponsive or produce inconsistent measurements, ensuring dependability and guaranteeing a minimum level of monitoring service.

VI. SECURITY, PRIVACY, AND COMPLIANCE

Appropriate technical, administrative, and organizational measures should be in place to mitigate risks throughout the system's lifecycle and demonstrate acceptable levels of risk during auditing activities. Security-by-design principles should be adopted, with risk controls and governance requirements integrated into all architectural layers using the security-by-design methodology. Additional constraints should be validated based on the proposed security architecture and addressed during design and development.

Security controls encompass access management, communication protection, data safeguarding, secure software engineering, cyber resilience, and auditing. Identity and access management covers user and device authentication, authorization mechanisms, role-based credential management, and secure onboarding, while secure communication and data protection define confidentiality, integrity, and privacy measures for data in transit and at rest. Auditing capabilities focus on tamper-evident logging, audit log trail creation, and data provenance to establish the ability to trace data during screening and evaluation processes for compliance purposes.

Access control mechanisms have been designed to limit access at both the cloud and sensor layers, ensuring only authorized personnel can access sensitive components. Authentication and authorization processes are complementary and enable secure interaction, supported by role-based credential management. At the IoT-sensor network layer, device onboarding is supported with industry-proven authentication protocols. Systems must support the full automated onboarding of IoT devices, including authentication credential installation.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

6.1. Identity and Access Management: Policy-driven automation in complex computing environments necessitates comprehensive identity and access management. Distributed IoT-sensor networks operating in multitenant settings require strict control over device onboarding and communication with the cloud management and orchestration components. A role-based access control system is used to filter communication and determine the authorizations of individuals and services. Zero-trust principles are applied to all requests. Dynamic authentication is employed to secure control signals sent from the cloud services to the edge nodes, where all devices are registered, authenticated, and authorized.

The connection procedure begins with the execution of a Secure Remote Password protocol between a client, represented by a device, and the Authentication Service, which stores the login credentials. The Trusted Third Party establishes the necessary keys for session encryption and helps to avoid man-in-the-middle attacks. Following successful authentication, every device receives individual authorization to interact only with specific cloud services, thereby promoting a zero-trust environment across the multitenant system. A dedicated Authorization Service monitors and evaluates the conditions imposed on the devices and rescinds access to the cloud services whenever anomalies are detected.

6.2. Secure Communication and Data Protection: Data protection entails safeguarding confidentiality, integrity, and availability, supported by secure storage, processing, and communication of sensitive information. Confidentiality breaches may stem from stealing data, digital assets, credentials, or intellectual property. Architecture safeguards sensitive information through encryption during transmission and storage, key management, and secure protocols.

To ensure encrypted data in transit, all communications, including those to third-party cloud services, utilize TLS or similar protocols. End-to-end encryption is implemented for telemetry collected by the IoT-sensor network and sent to the cloud. Sensitive data temporarily stored in the edge-cloud layer undergoes encryption consistent with the cloud service's security specifications. Sensitive information at rest is stored in encrypted format on cloud storage. The entire architecture leverages a centralized, role-based enterprise key management system for TLS certificates and data encryption/decryption keys, with appropriate protection measures. The encryption mechanism is capable of data masking to uphold data privacy, as required by supervisory authorities and the EU GDPR.

Equation 3: Data acquisition + sliding-window preprocessing equations

If a sensor can burst at f_{raw} but you cap at $f_{\max}$:

$$f_{\mathrm{cap}} = \min(f_{\mathrm{raw}}, f_{\max})$$

The paper explicitly mentions features from “sliding training windows” and “last N seconds.”

If you sample at f, a window of N seconds contains $W = Nf$ samples.

A common window feature (moving average) is:

$$\mu_t = \frac{1}{W}\sum_{i=0}^{W-1} x_{t-i}$$

and moving variance:

$$\sigma_t^2 = \frac{1}{W}\sum_{i=0}^{W-1} (x_{t-i} - \mu_t)^2$$

These window features are exactly what the paper suggests as “features may be computed from sliding windows.”

6.3. Auditing and Provenance: Tamper-evident logging and audit trails provide reliable evidence of activity, which is essential for security incident management and risk mitigation, as well as for demonstrating compliance with regulatory and governance requirements. Capturing the full data lineage of telemetry data enables automated provenance queries that determine the source and sequence of data alterations. Audit trails support retrospective assessments of service usage, resource consumption, and compliance deviations, and they also document policy rule definitions and evaluations.

Telemetry data stewardship poses challenging provenance requirements. Copying data into the cloud for long-term storage disrupts the original data flow, making it difficult to ascertain the source and direction of updates and to distinguish between external noise and operational anomalies. Capturing change events throughout processing reconciles such inconsistencies, and modeling the entire information lifecycle facilitates full chain-of-custody audits. Telemetry monitoring feeds security information and event management services that correlate data from multiple sources to reveal hidden attack patterns. Log entries document internal events and decisions, including processing stages and alterations, the use of predictive analytics models, and the triggering of automated corrective actions.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

VII. RELIABILITY, RESILIENCE, AND FAULT TOLERANCE

To ensure a reliable, resilient, and fault-tolerant cloud-based automation architecture for the distributed IoT-sensor network, dependability, recovery, and continuity are considered throughout the architecture. Risks associated with hardware redundancy, disaster recovery planning, and fault tolerance are evaluated.

Redundancy and Disaster Recovery describe redundancy schemes employed in the infrastructure, with corresponding failover mechanisms and DR strategies for both cached and time-series data. In the case of a presumed loss of physical infrastructure, recovery procedures are specified and the testing procedures of the DR plan are also positioned.

Fault Detection and Self-Healing define the monitoring signals used for the detection of anomalies in the infrastructure and their triggers. When any of these signals signal an anomaly, system behavior designed to correct the anomaly is presented, as well as the associated RTO.

The final subsection focuses on Quality of Service (QoS), presenting high-level definitions for QoS metrics, defining SLAs for each of these metrics, how the system should verify SLA compliance and the consequences of non-compliance. Emerging trends shaping cloud-based automation of IoT-sensor networks in semiconductor chip-testing environments include formal verification of orchestration workflows using model-checking techniques, cloud-level use of reinforcement-learning techniques to manage resource-allocation policies, real-time processing of anomaly-detection signals for self-healing of chip-testing systems, and integration of edge-cloud-sited machine-learning inference to enable higher quality of service in inference-based scenarios.

Use of the architecture enables automation workflows for chip testing and resource-management policies to meet host-emulation requirements. A support infrastructure is established to facilitate monitoring, logging, and management of the underlying IoT-sensor network. Future research efforts will consider machine-learning-based predictive-maintenance scenarios and explore additional enabling capabilities associated with advanced automation requirements.

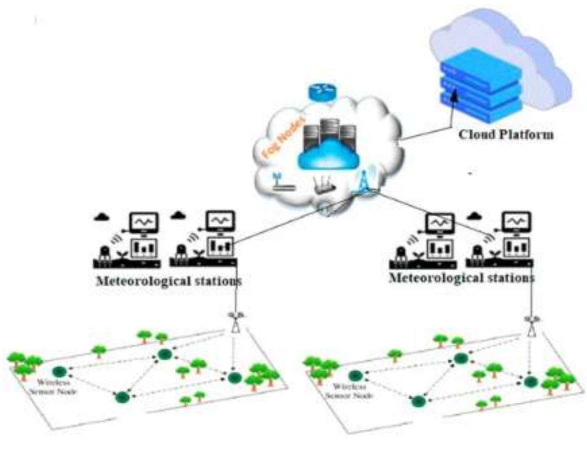


Fig 4: Reliability, Resilience, and Fault Tolerance

7.1. Redundancy and Disaster Recovery: Redundancy schemes mitigate the risk of component failure in edge-instance management, cloud-infrastructure deployment, or within the network links. The architecture can be designed with multiple edge instances, and a conventional load balancing scheme can distribute operation across the set. Within clouds, support for virtual machines or container instances can provide resilience against physical-machine failures. Secure replication of containers across hosts can enable workload migration in case a physical machine fails. Communication failures pose a distinct challenge. In wireline network segments, appropriate redundancy and failover schemes can be used. For wireless segments, techniques for mobile-sensor networks can enable messages to be relayed via intermediate devices. In this setting, message-peering protocols can provide mechanisms for efficient detection and elimination of loops.

Disaster recovery (DR) preparedness becomes important when dealing with business-sensitive operations. Suitable plans, documented as runbooks, can detail recovery steps for major operational failure scenarios. DR planning for cloud-

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

managed services is supported by backup functionality, typically undertaken outside operational-hours whenever possible. Test execution over recovery procedures is likewise planned for when operational demands are lowest.

7.2. Fault Detection and Self-Healing: Monitoring signals, such as temperature, humidity, power supply voltage, and UPS status, can indicate the health of the entire IoT-sensor network, but other signals can monitor the sensors themselves. Anomalies in data transmission can result from failure, ongoing replacement, or temporary disconnection of the sensors and should therefore serve as triggers for remedial actions. In simple cases, the operational procedures may need to be altered or a different traffic route with lower priority may need to be used. More advanced approaches consider faults in the sensor data to provide automation for network recovery. Conditions that can be detected include the sudden unavailability of the data and residual data streams being no longer suitable for fault-recognition purposes due to consistently high error rates, missing normal states, or the appearance of a non-expected normal state. Such situations typically require a more complicated recovery plan involving maintenance activity.

Automated restoration of service can therefore be implemented, along with defined best-practice recovery steps to limit the impact of the fault on the monitoring service. Automated restoration ranges from simple preparation and alerts to more complex requests for dismissal by third-party services. The recovery time objective is highly variable, depending on the chosen remediation mechanism and the problem requirements. Real-time monitoring systems should ideally react faster than the observed phenomena, allowing anomaly appearances to be short, and often invisible in the main system services. Nonetheless, delays are sometimes acceptable and can be factored into action design.

7.3. Quality of Service and SLAs: Quality of Service and Service Level Agreements (SLAs) are vital for ensuring that the response time and throughput meet performance expectations. Users expect the executed tests to meet specified time limits, while active workflows require sufficient support from the edge-computing nodes. Therefore, defining appropriate QoS metrics is essential for guiding the automation of the IoT-sensor network layer and enabling fair workload distribution across the edge and cloud.

Service Level Agreements (SLAs) formalize QoS expectations and the corresponding penalties for non-compliance. Three main categories of QoS metrics are considered over two distinct SLAs, based on the need for a child service that is separate from, or shared with, other automated services: (i) the preset limits that a child service must not breach, (ii) the aggregate quality provided by the IoT-sensor network layer to an active workflow, and (iii) the overall system characteristics defined by the ongoing workloads and the remaining capacity of the involved components. Monitoring these QoS properties, measuring their fulfillment, and assessing SLA completions are critical for achieving the targeted service quality and for implementing the proper policy-driven actions.

SLA/QoS category (paper)	Typical variable	Example equation
(i) Preset limits per child service	$RT_s,\ Th_s$	$RT_s \le RT_s^{\max}, Th_s \ge Th_s^{\min}$
(ii) Aggregate quality for workflow	RT_w	$RT_w = \sum_{s \in w} RT_s$ (serial steps)
(iii) Overall system characteristics	utilization U	$U = \frac{\text{used capacity}}{\text{total capacity}}$

Table: QoS / SLA metric categories (structured table)

VIII. CONCLUSION

The proposed cloud-based smart automation architecture for distributed IoT-sensor networks in semiconductor chip-testing environments contributes to high levels of automation by integrating support for monitoring, management, orchestration, control, and data-processing capabilities. Design decisions and supporting processes and mechanisms are driven by an extensive review of related work and a detailed analysis of a motivating scenario, which together highlight the requirements and open challenges associated with realizing cloud-based automation capabilities in such specialized chip-testing environments.

8.1. Emerging Trends: Research interests in cloud-native automation for IoT-sensor networks within semiconductor-fabrication or testing contexts expand beyond the information-capturing layer around data-management and analytics. Emerging trends in the surrounding architectural blueprint span distinct areas of investigation and exploration, though an exhaustive enumeration lies outside the scope of this study. Advanced topics may include foundational security and

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

privacy considerations, the automation of physical tasks supported by robotic systems, and the exploitation of quality attributes in the scheduling of cloud-based resource allocation and workflow execution. Recent developments in AI-powered systems also invite further examination.

The inherent dependence of smart systems on the availability and quality of input data fuels interest in the combination of semiconductor sensing and imaging with AI techniques—rich sources of telemetry pertaining to the internal state of assets and the external environment. Integrating a variety of sensors and data sources into an automated, globally coordinated information-capture solution adds significant value across multiple applications, spanning predictive maintenance and resource-consumption optimization for hardware infrastructure and energy distribution. Real-time capabilities can also support symbiotic interaction with intelligent agents. Recent advancements in the research fields of explainable AI and self-supervised learning may further facilitate the design and deployment of AI models for human-critical applications.

REFERENCES

1. Meda, R. (2025). AI-Driven Demand and Supply Forecasting Models for Enhanced Sales Performance Management: A Case Study of a Four-Zone Structure in the United States. Metallurgical and Materials Engineering, 1480-1500.
2. Kolla, T. (2025). Anomaly Detection Models for Outlier Provider Behavior in Cost and Treatment Patterns. Journal of Computational Analysis & Applications, 34(12), 1204.
3. Vajpayee, A., Khan, S., Gottimukkala, V. R. R., Sharma, D., & Seshasai, S. J. (2025). Digital Financial Literacy 4.0: Consumer Readiness for AI-Driven Fintech and Blockchain Ecosystems. International Insurance Law Review, 33(S5), 963-973.
4. Balamurugan, J., Aitha, A. R., Kishore, D. S. C., Reenaraj, T., Babu, T. S., & Kumar, B. B. (2025, November). Adaptive AI-Driven Optimization in Smart Manufacturing: A Hybrid Neural-Network and Genetic-Algorithm Approach. In 2025 International Conference on Emerging Engineering Technologies and Applications (IC-EETA) (pp. 690-697). IEEE.
5. Kumar, K. M., Parasar, A., Walia, A., Inala, R., & Thulasimani, T. (2025, August). Enhancing Risk Management Strategies in Financial Institutions Using CNN and Support Vector Regression. In 2025 5th Asian Conference on Innovation in Technology (ASIANCON) (pp. 1-6). IEEE.
6. Nigam, N., Sireesha, B., Ediga, P., Segireddy, A. R., & Bokde, S. (2025, December). Comparative Evaluation of Cloud Security Algorithms Using Multiple Classifiers with an Optimized Intrusion Detection System. In 2025 IEEE 5th International Conference on ICT in Business Industry & Government (ICTBIG) (pp. 1-6). IEEE.
7. Yandamuri, U. S. (2024). AI-Driven Decision Support Systems for Operational Optimization in Hospitality Technology. Metallurgical and Materials Engineering.
8. Amistapuram, K. (2025). GENERATIVE AI FOR CLAIMS EXCEPTIONS AND INVESTIGATIONS: ENHANCING RESOLUTION EFFICIENCY IN COMPLEX INSURANCE PROCESSES. Available at SSRN.
9. Mangala, N. (2025). Agentic Data Pipelines: Autonomous ELT Orchestration Using AI Agents on Microsoft Fabric and Databricks. International Journal of Computer Technology and Electronics Communication, 8(6), 11891-11907.
10. Kolla, S. H., & Mangala, N. (2025). DESIGNING AUTONOMOUS LLM AGENT FRAMEWORKS USING GEN AI PIPELINES TO ENHANCE CUSTOMER SERVICE MANAGEMENT AND KNOWLEDGE WORKFLOWS. Lex Localis-Journal of Local Self-Government, 23 (S6), 9719–9733.
11. Kummari, D. N., Burugulla, J. K. R., Malempati, M., Amistapuram, K., Garapati, R. S., & Nagabhyru, K. C. (2025, December). Enhancing Audit Compliance and Operational Efficiency in Manufacturing and Commercial Insurance Through Agentic AI and Data Engineering Frameworks. In 2025 IEEE International Conference on Communication Networks and Computing (CNC) (pp. 714-720). IEEE.
12. Garapati, R. S., Adusupalli, B., Kaulwar, P. K., Gadi, A. L., Annapareddy, V. N., & Challa, K. (2025, December). The Evolution of Digital Payments: A Study on AI-Powered Transaction Monitoring Systems. In 2025 3rd International Conference on IoT, Communication and Automation Technology (ICICAT) (pp. 1-8). IEEE.
13. Priyanka, R. P., Annapareddy, V. N., Yenugu, B. C., Nagabhyru, K. C., & Kapila, D. (2025, September). Optimization of Battery Management Systems Using Machine Learning. In 2025 International Conference on Computing and Communications (COMPUTINGCON) (pp. 1-6). IEEE.
14. Segireddy, A. R. (2025). AN INTELLIGENT, REAL-TIME DIGITAL FABRIC FOR HEALTHCARE AND FINANCIAL ECOSYSTEMS USING AUTONOMOUS LEARNING AND GENERATIVE SYSTEMS. TPM–Testing, Psychometrics, Methodology in Applied Psychology.
15. Mangalampalli, B. M., & Kolla, S. K. (2025). Large Language Models for Automated Healthcare Data Dictionary Generation and Maintenance. Vascular and Endovascular Review, 8(20s), 363-375.

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

16. Kolla, S. H., & Mattaparthi, R. (2025). Hybrid Gen AI Systems: Integrating Small LMs with Large Language Models for Cost-Efficient Enterprise Automation and Decision Intelligence. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 8(6), 13345-13357.

17. Mangalampalli, B. M., Bandi, V. D. V. K., Kolla, S. K., & Kumar, M. V. K. (2025). Towards Self-Evolving Healthcare Intelligence: Integrating Advanced Learning Systems with Real-Time Clinical Data Pipelines. Cultura: International Journal of Philosophy of Culture and Axiology, 22(12s), 464-486.

18. Paleti, S. (2025). Smart finance: Artificial intelligence, regulatory compliance, and data engineering in the transformation of global banking. Deep Science Publishing.

19. Manikandan, K., Pamisetty, V., Challa, S. R., Komaragiri, V. B., Challa, K., & Chava, K. (2025). Scalability and efficiency in distributed big data architectures: a comparative study. Metallurgical and Materials Engineering, 31(3), 40-49.

20. Reddy, V. A. R. (2025). Journal of Rare Cardiovascular Diseases. Health, 5(3), 402-422.

21. Adusupalli, B., Elakkiya, M., Balakrishnan, N., Negi, A. S., Balaram, A., & Rathnakumari, E. A. (2025, May). Decentralized Blockchain Powered Citizen Participation Frameworks for Next Generation Transparent Elections and Trustworthy Governance. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 367-380). Atlantis Press.

22. Loganathan, R. (2025). AGENTIC AI FRAMEWORKS FOR AUTONOMOUS RISK DETECTION AND COMPLIANCE REMEDIATION IN ENTERPRISE DATA CENTER OPERATIONS. Lex Localis-Journal of Local Self-Government, 23 (S6), 9672–9697.

23. Davuluri, P. S. L. (2023). AI-Augmented Sanctions Screening: Enhancing Accuracy and Latency in Real Time Compliance Systems. AI-Augmented Sanctions Screening: Enhancing Accuracy and Latency in Real Time Compliance Systems (December 15, 2023).

24. Kolla, S. K. (2025). Next-Generation Precision Healthcare: AI-Driven Clinical Intelligence, Predictive Analytics, and Adaptive Decision Support Systems. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 8(4), 12539-12552.

25. Annapareddy, V. N., Challa, K., Komaragiri, V. B., Sriram, H. K., Kalisetty, S., & Goma, T. (2025, April). Explaining AI Techniques such as SHAP LIME and RISE in Limited Sample Size Neuroimaging Studies. In 2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0 (pp. 1-6). IEEE.

26. Mashetty, S. (2025). Technology-driven analytics in mortgage-backed securities for single-family mortgage financing. Available at SSRN 5236573.

27. Garapati, R.S., Paleti, S., Meda, R., Nagabhyru, K.C., Deepa Priya, B.S. (2026). Physical-Unclonable-Function-Based Secure and Anonymous User Authentication for Smart Homes. In: Bhateja, V., Yang, XS., Flores-Fuentes, W., Chowdary, P.S.R. (eds) Signal Processing, Telecommunication & Embedded Systems: Automation and Sustainability Applications. ICMEET 2025. Lecture Notes in Electrical Engineering, vol 1598. Springer, Cham. https://doi.org/10.1007/978-3-032-20235-2_33

28. Seenu, A., Sheelam, G. K., Motamary, S., Meda, R., Koppolu, H. K. R., & Inala, R. (2025, July). AI-Driven Innovations in Infrastructure Management with 6G Technology. In 2025 2nd International Conference on Computing and Data Science (ICCDS) (pp. 1-6). IEEE.

29. Nagabhyru, K. C. (2025). Human In The Loop Generative AI: Redefining Collaborative Data Engineering For High Stakes Industries. Metallurgical and Materials Engineering.

30. Babaiah, C., Dobriyal, N., Shamila, M., Aitha, A. R., Patel, S. P., & Upodhyay, D. (2025, December). Intelligent Fault Detection and Recovery in Wireless Sensor Networks Using AI. In 2025 IEEE 5th International Conference on ICT in Business Industry & Government (ICTBIG) (pp. 1-6). IEEE.

31. Amistapuram, K., Pandiri, L., Raju, V. R., Paleti, S., Singireddy, S., & Sheelam, G. K. (2025). AI-Based Cloud Infrastructure and MLOps Frameworks for Scalable Data Engineering Across Banking and Insurance. In 2025 IEEE International Conference on Communication Networks and Computing (CNC) (pp. 186–192). IEEE. 2025 IEEE International Conference on Communication Networks and Computing (CNC). https://doi.org/10.1109/cnc68716.2025.11484532

32. Ashokkumar, S., & Amistapuram, K. (2025, October). Attention-Guided Spatial Temporal Framework for Deepfake Detection on Social Video Platforms. In 2025 International Conference on Communication, Computer, and Information Technology (IC3IT) (pp. 1-6). IEEE.

33. Selvaraj, F. J., Rani, S., Nagubandi, A. R., Chawla, C., & Sekar, G. (2026). Beyond Traditional Ledgers: A Blockchain-Integrated Accounting Model for Seamless Digital Transformation in Retail Economies. Advances in Consumer Research, 3(1), 934-941.

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

34. FinOps Strategies for AI-Enabled Real-Time Compliance Platforms in Cloud Native Environments. (2025). MSW Management Journal, 35(2), 2080-2088.
35. Mattaparthi, R. (2023). Deep Learning-Driven Combustion Anomaly Detection in Diesel Powertrains: A Multi-Sensor Fusion Approach for Real-Time ECM Adaptation. International Journal of Intelligent Systems and Applications in Engineering, 11, 1084.
36. Kolla, T. (2025). The Future of Healthcare Analytics: Leveraging AI and Data Engineering for Personalized Medicine. Journal of Computer Science and Technology Studies, 7(4), 634-640.
37. Peddi, R. K. (2024). AI-Based Workforce Analytics for SLA Governance and Uptime Assurance in Data Centers. Journal of Computational Analysis and Applications (JoCAAA), 33(08), 8589-8601.
38. Mashetty, S. (2025). LEVERAGING DEEP LEARNING, NEURAL NETWORKS, AND DATA ENGINEERING FOR INTELLIGENT MORTGAGE LOAN VALIDATION. INTERNATIONAL JOURNAL OF SOCIAL SCIENCE & INTERDISCIPLINARY RESEARCH ISSN: 2277-3630 Impact factor: 8.036, 14(04), 51-65.
39. Meda, R. (2025). Dynamic Territory Management and Account Segmentation using Machine Learning: Strategies for Maximizing Sales Efficiency in a US Zonal Network. EKSPLORIUM-BULETIN PUSAT TEKNOLOGI BAHAN GALIAN NUKLIR, 46(1), 634-653.
40. Mangalampalli, B. M., Kolla, S. K., Bandi, V. D. V. K., Yandamuri, U. S., & Rani, P. S. (2025). Designing Intelligent Healthcare Ecosystems through Adaptive Data Integration and Autonomous Learning Systems. Vascular and Endovascular Review, 8(20s), 330-347.
41. Mattaparthi, R. (2025). GenAI-Augmented Diagnostic Reasoning for Diesel Engine Fault Triage: A Large Language Model Framework for Technician Decision Support at Scale. Journal of Material Sciences & Manufacturing Research, 6(12), 1.
42. Gadi, A. L., Garapati, R. S., Inala, R., Singireddy, J., & Kapila, D. (2025, October). Robust Mutual Authentication for Distributed IoT Systems: Balancing Security and Efficiency. In International Conference on Microelectronics, Electromagnetics and Telecommunication (pp. 538-548). Cham: Springer Nature Switzerland.
43. Danghi, P. S., Maniraj, K., Jain, P., Adilakshmi, K., Garapati, R. S., & Jain, S. K. (2025, December). Artificial Intelligence Based Energy Optimization Framework for Wireless Sensor Networks. In 2025 IEEE 5th International Conference on ICT in Business Industry & Government (ICTBIG) (pp. 1-6). IEEE.
44. Balamurugan, J., Bhuvaneswari, M., Aitha, A. R., Nagaraju, S., Viswanathan, R., & Dhasarathan, N. (2025, November). Explanatory Transformer-Based Sequential Recommendation: Assessing BERTRec with SASRec for Customer Behaviour Prediction. In 2025 International Conference on Emerging Engineering Technologies and Applications (IC-EETA) (pp. 470-475). IEEE.
45. Radhakrishnan, P., Nagabhyru, K. C., Manonmani, C., Srinu, M., Kaur, H., & Nandhini, N. (2025, October). K-Means-KNN Hybrid Model for Efficient Intrusion Detection in Cloud-based IoT Systems. In 2025 10th International Conference on Communication and Electronics Systems (ICCES) (pp. 1583-1588). IEEE.
46. Radha, S., Gottimukkala, V. R. R., Thottara, S., & Vandhana, K. (2025, October). Adaptive Video Streaming Over 5G Networks Using Deep Reinforcement Learning with Closed-Loop Feedback Mechanism for Bitrate Control. In 2025 International Conference on Communication, Computer, and Information Technology (IC3IT) (pp. 1-6). IEEE.
47. Srikanth, T., Segireddy, A. R., & Elavarasi, S. A. (2025, October). STaSFormer-SGAD: Semantic Triplet-Aware Spatial Flow-Guided Spatio-Temporal Graph for Anomaly Detection in Surveillance Videos. In 2025 International Conference on Communication, Computer, and Information Technology (IC3IT) (pp. 1-7). IEEE.
48. Amistapuram, K. (2025). Agentic AI for Next-Generation Insurance Platforms: Autonomous Decision-Making in Claims and Policy Servicing. Journal of Marketing & Social Research, 2, 88-103.
49. Nagubandi, A. R. (2025). Advanced Predictive Autonomous Agents for Multiportfolio Risk Analytics and Real-Time Enterprise P&L Decisioning: Self-Learning AI Systems for Multi-counterparty Derivatives, Collateral Valuation, and Accounting Reconciliation. Collateral Valuation, and Accounting Reconciliation (December 01, 2025).
50. Mahadevan, S. (2024). Clouding the Future: Innovating Towards Net-Zero Emissions. International Journal of Computing and Engineering, 6(2), 17-23.
51. Yandamuri, U. S. (2023). An Intelligent Analytics Framework Combining Big Data and Machine Learning for Business Forecasting. International Journal Of Finance, 36(6), 682-706.
52. Kolla, S. H. (2025). Autonomous Agentic Frameworks for Enterprise Service Operations and Intelligent Process Automation. International Journal of Science, Research and Technology, 8(4), 14643-14655.
53. Kolla, S. K., & Bandi, V. D. V. K. (2025). Autonomous Clinical Monitoring Platforms Using Reinforcement Learning and Deep Neural Networks. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 8(6), 13300-13313.
54. Bandi, V. D. V. K. (2025). Self-Optimizing Data Pipelines Using Machine Learning for Cloud Workloads. Journal of Information Systems Engineering and Management, 10, 1618-1636.

International Journal of Advanced Research in Computer Science & Technology (IJARCST)

| ISSN: 2347-8446 | www.ijarcst.org | editor@ijarcst.org |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 9, Issue 4, July-August 2026||

DOI:10.15662/IJARCST.2026.0904004

55. Mattaparthi, R. (2024). Transformer-Based Fault Diagnosis for Large-Scale Standby Power Generators: Partial Discharge Pattern Recognition at Hyperscale Data Center Installations. Journal of Computational Analysis and Applications (JoCAAA), 33(08), 8781-8799.
56. Kolla, T. (2025). Generative AI for Intelligent Medical Coding and Healthcare Analytics. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 8(6), 13285-13299.
57. Mahadevan, S. (2024). Empowering Manufacturing: Generative AI Revolutionizes ERP Application. Int. J. Innov. Sci. Res, 9, 593-595.
58. Loganathan, R. (2024). Generative AI-enabled compliance documentation and audit trail automation for global data center governance. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 15(3), 487-504.
59. Ranga Reddy, V. A. (2024). Comparing Batch vs. Streaming Approaches in Healthcare Data Warehousing Environments. Journal of Neonatal Surgery, 13(1), 2287-2309.
60. Challa, K. (2025). Ai driven fraud detection in digital payments using big data and machine learning. American Journal of Analytics and Artificial Intelligence (ajaai) with ISSN 3067-283X, 3(1).
61. Chabokpour, J., Kalisetty, S., Malempati, M., Challa, K., Mandala, V., Kumar, B., & Azamathulla, H. M. (2025). Integrating Hydrological and Hydraulic Approaches for Adaptive Environmental Flow Management: A Multi-Method Approach for Adaptive River Management in Semi-Arid Regions. Water, 17(7), 926.
62. Sadhukhan, S. K., Jaishankar, R., Adusupalli, B., Revathi, V. M., Kaliappan, S., & Ponnusamy, M. (2025, October). Healthcare Insurance Fraud Detection Using a Blockchain–Deep Neural Network–AdaBoost Hybrid Model. In 2025 2nd Asian Conference on Intelligent Technologies (ACOIT) (pp. 1-6). IEEE.
63. Adusupalli, B., Malempati, M., Paleti, S., Mashetty, S., & Singireddy, J. (2025). Integrated financial ecosystems: AI-driven innovations in taxation, insurance, mortgage analytics, and community investment through cloud, big data, and advanced data engineering. Journal of Information Systems Engineering and Management, 10, 1103-1117.
64. Mashetty, S., & Valiki, S. (2025). AI-Powered Data Engineering for Intelligent Retail Stock Management. International Journal of Research and Applied Innovations, 8(6), 13096-13109.
65. Meda, R. (2025). Transforming retail and paint manufacturing supply chains through agentic AI, machine learning, and intelligent data infrastructure. Global Pen Press UK.
66. Mangala, N. (2024). Leveraging Microsoft Fabric lakehouse as an AI-ready data platform for enterprise analytics. Journal of Information Systems Engineering and Management.
67. Lebcir, I., Shah, A., Nagubandi, A. R., Dhoke, S. M., & Mishra, M. K. (2025). FinTech and Financial Inclusion in Emerging Economies: An Empirical Assessment. Advances in Consumer Research, 2(6).