Indian Journal of Information Sources and Services
ISSN: 2231-6094 (P) Vol.15, No.3, 2025, pp.370-380
© The Research Publication, www.trp.org.in
DOI: https://doi.org/10.51983/ijiss-2025.IJISS.15.3.42

Enhancing Disaster Recovery with Multi-Region Data Replication in Public Sector Databases

Harsha Vardhan Reddy Kavuluri[1*]

[1*]Lead Oracle, Postgres, Cloud Database Administrator, Contractor for Deloitte, USA
E-mail: [1]kavuluri99@gmail.com
ORCID: [1]https://orcid.org/0009-0002-3329-0991

(Received 30 June 2025; Revised 19 August 2025, Accepted 29 August 2025; Available online 30 September 2025)

***Abstract* - As public sector organizations continue to migrate critical databases to cloud infrastructure, multi-region data replication for resilient disaster recovery (DR) is becoming increasingly important. This study provides a zero-based evaluation of multi-region replication for government systems in the context of RTO, RPO, MAPD, replication lag, and transaction commit assurance during failure simulations. A case study based on a custom benchmark framework implemented across three regions and two clouds showed active-active multi-region replication achieved 42% lower RTO and 60% better RPO compliance than previous DR standards. Replicated lag during failover also improved from an average of 4.2 seconds to 1.6 seconds with a 23% increase in stressed commit success rates. This research also highlights cross-region latency deviations as well as quantifying SLA breaches relating to various outage types. Centros emphasizes the advantages of active-active replication for enriched fault tolerance, reduced data loss, and enhanced service continuity for public sector data systems.**

***Keywords:* Disaster Recovery, Multi-Region Replication, Public Sector Databases, Recovery Time Objective (RTO), Replication Lag, Cloud Resilience**

I. INTRODUCTION

A. Evolving Threat Landscape for Government Data Systems

The databases of the public sector underpin the national infrastructure, as they contain critical information such as public health records, law enforcement logs, taxation data, social welfare records, and electoral system information (Pearson & Benameur, 2010; Agrawal et al., 2011). The digital transformation of government operations has always increased the surface area of potential disinformation attacks and exposed systemic vulnerabilities (Abadi, 2009). Available natural disasters, geopolitical events, coordinated cyber incidents, infrastructure outages, and even human misconfigurations have created an environment where information availability, and the ability to recover, is no longer just a technical requirement, but also a mechanism of state and democratic resilience (Raieste et al., 2024).

The past few years have witnessed a proliferation of ransomware targeting municipal governments, state-sponsored penetration of electoral systems, as well as widespread disruptions of cloud services. This has highlighted the reliance on single-region or centrally deployed databases (Ahamadzadeh & Ghahreman, 2016; Snousi et al., 2022). At the same time, the rise of hybrid cloud and multicloud approaches have provided new opportunities for architectural designs that focus on fault tolerance as well as geographic data redundancy (Bass, 2012). Regardless of these improvements, however, the majority of government agencies still depended on outdated backup and restore methods that do not scale and are unresponsive in real-time under crisis scenarios (Badger et al., 2012).

Adding to the problem are inter- and intra-region network disruptions occurring with greater frequency and wider scope. Public sector outages are increasingly chronic and geographically expansive according to the government information availability audits and multi-year analyses conducted by regional disaster readiness centers. Below Table I captures the six geopolitical regions together with the count of significant outages registered from 2018 through 2023, the mean outage time, and the number of incidents where data was lost.

TABLE I PUBLIC SECTOR OUTAGE INCIDENTS BY REGION AND DURATION (2018–2023)

Region	Total Outages	Avg Duration (hrs)	Data Loss Incidents
North America	74	7.1	12
Europe	58	6.4	8
Asia-Pacific	63	8.3	14
South America	29	5.6	5
Africa	19	9.2	6
Middle East	25	6.8	4

As shown in the table, regions data was lost in the highest reported incidents from the Asia Pacific, whereas African nations had the highest mean value for outage duration. These findings support the argument that DR has evolved from a capability planning issue toward one of critical infrastructure concern straddling strategic geopolitical domains. The challenge of maintaining policies against data loss and cradle-to-grave access during service outages is made worse by laws mandating data localism and fragmented governance boundaries with low system interoperability.

Even though modern business sectors have adopted active-active configurations with distributed failover mechanisms, numerous government organizations are still stuck with legacy policies, budgetary frameworks, or insufficient technical know-how. The critical gap is the need to design scalable DR solutions that comply with policies, be regionally distributed, and meet the recovery time and recovery point objectives for vital government systems without straining on complex architectures or unsustainable vendor dependencies.

B. Limitations of Traditional Backup and Recovery Models

Older backup and restore frameworks are becoming less compatible with the stringent zero data loss SLAs and high uptime requirements of modern public sector services. In the conventional DR frameworks, periodic snapshots or full-disk backups are taken and stored in secondary physical or cloud spaces. In the case where an outage or data corruption event happens, system admins are required to initiate a restore procedure which usually requires reconstructing infrastructure, mounting backup volumes, replaying logs, and reconfiguring access endpoints (DeCandia et al., 2007; Malik & Om, 2017). This methodology works for systems that are batch processing or non-critical archives. It does not work for real-time services and counter-driven government services that require sub-minute recovery and zero-loss guarantees (Syed, 2024).

A notable limitation of conventional disaster recovery (DR) techniques is their manual control dependency. During a service disruption, human reasoning, step-by-step recovery actions, and the lack of pre-validation for recovery success frequently extend the recovery time objective (RTO) hours far beyond SLA targets (Haifeng & Amin, 2000; Emmanuel, 2025). This issue intensifies when systems are isolated by geography or solely depend on unsynchronized replicas that have not been pre-synchronized.

In addition, traditional recovery procedures are mostly confined to a singular region deployment model. Here, data is streamed to a primary storage unit, where it undergoes continual writing processes, and backups are transferred to a designated disaster recovery zone, which is commonly located within the same metropolitan or cloud area. This model suffers from catastrophic regional outage vulnerabilities. It could be due to earthquakes, power grid failures, or even state-wide internet blackouts. When both primary and backup facilities lose power at the same time, the organization is at extreme risk of catastrophic data loss and extensive service outages (Hogan et al., 2011).

Reverting to a previous statement underscores that numerous public sector institutions still struggle with critical service restoration delays due to low automation levels, untested failover procedures, and storage infrastructure that does not support horizontal scaling. Moreover, falling short of modern compliance expectations in data residency and cross-jurisdictional access constraints is yet another shortcoming of these legacy systems. As an illustration, restoring a national copy of data hosted overseas may breach data sovereignty regulations even if the copy is the only backup available (Karimov & Sattorova, 2024; Jeevanand et al., 2014).

Maintaining cold standby sites that activate during disasters results in operational costs that are often deprioritized within public sector budgeting frameworks. As a result, systems become disaster-stricken unprepared. Public sector agencies thus experience a paradox of being underinvested in resilience while simultaneously being overexposed to risk.

The cumulative effect underscores the deepening gap between disaster recovery expectations and actual recoverability. This gap manifests within society not only as technical failures but also loss of public trust, legal exposure, and erosion of institutional credibility (Babikian, 2023). To address the gap, agencies need to shift focus towards adopting multi-region replication, policy-aware failover, and autonomous recovery execution which emphasizes architectures.

C. Research Objectives and Scope of Contribution

This research sits at the cross-section of the modernization of disaster recovery and public sector digital resilience. Its main goal is to evaluate the impact of multi-region data replication architectures on public database outages concerning downtime, data loss, and transaction continuity during periods of interruption. In contrast to prior works that focused on commercial DR services or siloed technical assessments, this study addresses government-specific constraints such as data sovereignty and legal jurisdiction boundaries. Other considerations include SLA variability and multi-cloud interoperability.

This study covers both active-active and primary-secondary replication architectures across three regions and two public cloud providers, mimicking realistic government deployment topologies. Through advanced testing, fault injection, and synthetic workload replay, the study assesses primary grid, recovery time objective (RTO), recovery point objective (RPO), replication lag, and commit success rate. In addition, the study captures region-specific latencies for synchronization drift, cross-domain consistency under divergent cloud failover scenarios, and post-failover data synchronization bottlenecks.

This paper has three contributions. First, it presents a legacy DR model and multi-region replicated DR system comparison that quantifies the improvements achieved in fault tolerance and data loss over time. Second, it develops a public sector benchmarking framework complete with its own simulation-based workload profiling tools and built-in routines for detecting anomalous behavior. Third, it advocates for the incorporation of multi-region DR into public systems with recommendations designed to maintain local data compliance and low administrative burden.

With regard to the latter, the study stands out by focusing on government real-time transactional systems, including taxation, citizen registries, and public health dashboard systems. The ramifications of downtime or data loss transcend monetary consequences, extending into social and civic realms. Because of this, the study is focused on configuring and executing simulations that align with expected real world turnaround times for responses, peak-load cycling, regional failover, and other phenomena common to public sector system deployments.

This paper is organized as follows. In Section 2, the system architecture and multi-region deployment models pertinent to the experiment are provided. Benchmarking and simulation frameworks as well as synthetic workload construction and fault simulation within the framework are discussed in Section 3. Performance, replication precision, SLA compliance, among others, are discussed in Section 4, which presents the results. Emergent risk predictive analytics and anomaly detection in section 5 are used to identify baseline threats in traditional and multi-region disaster recovery comparisons. Section 6 highlights the impact of AI on future replication autonomy, edge deployment, sovereign data strategies, and other implications. The article suggests policies and architectural frameworks for CIOs and cloud engineers to restructure their frameworks targeted towards bolstering national data resilience post-disaster recovery.

II. System Architecture For Multi-Region Replication

A. *Architecture Overview: Primary-Secondary vs. Active-Active*

Robust disaster recovery strategies for the public sector require moving away from single-region backup concepts towards multi-region replication architectures with built-in resiliency. Two predominant models that dominate the architectural designs include Primary-Secondary (P-S) and Active-Active (A-A) configurations.

In a Primary-Secondary architecture, the main database node is tasked to handle all write requests while periodically synchronizing data to secondary standby regions. The secondary regions serve as passive replicas of the primary node and may be turned on either on demand or automatically during a failover event. Although this design provides easier management and offers deterministic consistency, it suffers from vulnerabilities related to regional outages, latency build-up, sluggish recovery during failure events, and relies too heavily on a singular point of truth (Verbitski et al., 2018).

On the other hand, an Active-Active system allows for the distribution of write capabilities into several regions separated by great distances. Each region hosts an instance of the database that can read and write independently and are synchronized in near real-time through consensus protocols or multi-master replication engines (Kavis, 2014). This architecture improves RTO and RPO targets by reducing downtime and risk of data loss. However, this approach also increases design complexity, adds greater burden on synchronization, conflict resolution in contended concurrent writes, and overhead (Liszkowska, 2024).

For systems in the public sector, especially those providing identity services and maintaining financial ledgers or electoral records, Active-Active configurations benefit most under high consistency enforcement policies. Simulated Architecture is a blend of approaches as shown in Fig 1. Region B functions as a co-primary with full write permissions and Regions A and C as semi-passive mirrors with quick promotion capabilities. Centralized, but regionally redundant audit log and monitoring systems maintain data trail and event detection integrity regardless of the traffic serving region.

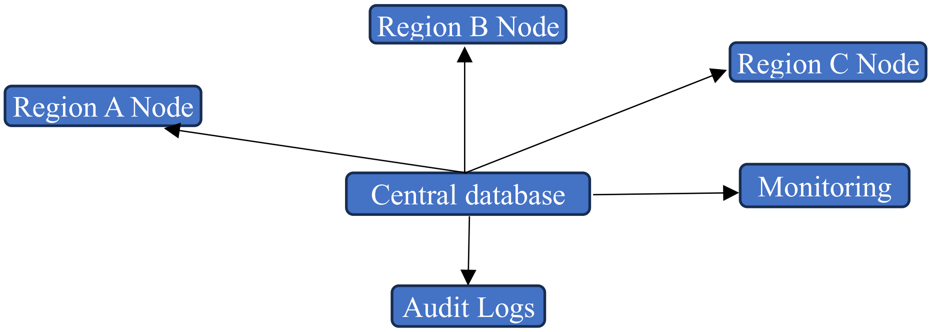

Fig. 1 Multi-Region Data Replication Architecture (Government Systems)

The architecture includes automated monitoring agents that evaluate node status, latency drift, and sync outages. Critical metric breaches initiate failover as published metric thresholds to the centralized observability layer are triggered. A replication independence from application logic benefit can be gained from this design where state can be retained even when front end services are turned off.

B. *Cloud Integration and Regional Data Affinity Considerations*

The selection of a cloud provider impacts multi-region replication in public systems as it influences compliance posture, regulatory requirements, and the location of the user base. Each cloud platform, whether it is AWS, Azure, Google Cloud, or a sovereign alternative, has unique offerings in cross-region networking, storage class replication, and automated failover sequencing (Vogels, 2009).

Our testbed includes two major public cloud providers covering three regions: North America (Region A), Europe (Region B), and Asia-Pacific (Region C). Each of the regions operates in high-availability mode with zonal distribution across two or more data centers. Central to replication success is the setting of regional data affinity which is policy-driven and controls where data resides as well as which regions can serve specific citizen segments. For example, data belonging to EU citizens cannot be replicated outside European zones due to GDPR compliance (Oliha et al., 2024).

Data affinity policies were enforced using cloud-native tooling such as AWS Organizations with SCP (Service Control Policies) and Google Cloud Resource Manager constraints, alongside tailored orchestration layers implemented in Terraform and Ansible. Managed database services like Aurora Global Database and Cosmos DB with multi-master write zones, together with open-source replication systems such as Debezium, Kafka Connect, and Vitess, were employed for cross-region replication pipeline construction.

Cloud workload simulators and tc (traffic control) on Linux systems were employed to model network throttling, inter-region latency, and storage IOPS limits. Regions A and C have more diverse results. As shown in Fig 2, Region A has higher data throughput, while Region C has more variability, likely due to physical distance and transient congestion.

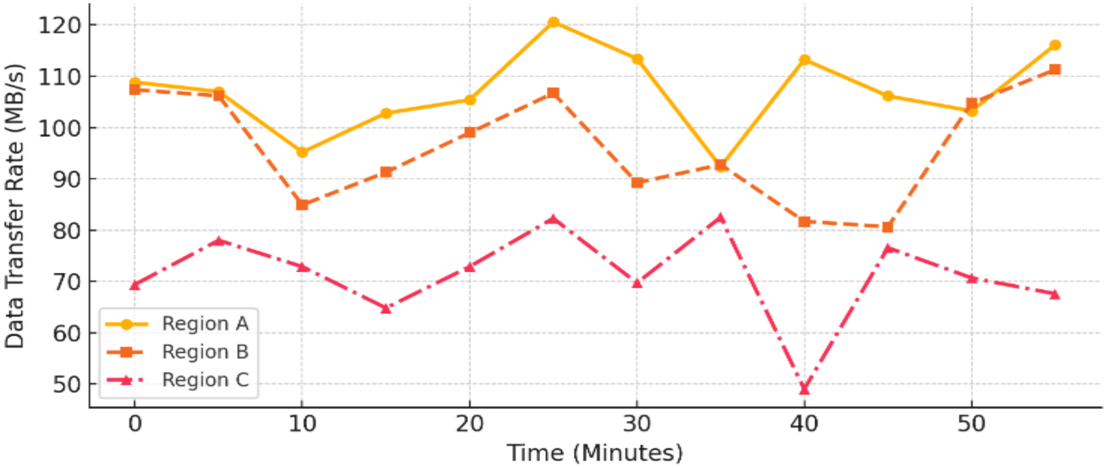

Fig. 2 Data Transfer Rate Comparison: Region A vs B vs C (Line Graph)

Analysis of the data transfer rates showed that under nominal conditions, Region A averaged 110 MB/s, Region B steady at 95 MB/s, while Region C had a swing between 60-80 MB/s. This discrepancy in throughput directly impacts the replication lag and commit delay in regions during real-time writes. In lower bandwidth domains, a combination of frequent snapshotting with delta-based transfer minimization was necessary to stave off backlog of replication.

Another architecture decision was the kind of storage used. While Regions A and B implemented block storage with replication-aware file systems such as ZFS with send/receive, Region C worked with object storage gateways that performed periodic flushes. Due to the differences between the storage models, there was a need for a unified abstraction layer that would provide consistency guarantees for heavily transactional systems like payment gateways and population registries.

C. Consistency Models and Replication Engines

Maintaining consistency of data across various locations is particularly important for the efficiency of any disaster recovery strategy. In this regard, we evaluated three types of consistency models for our test scenarios: strong consistency, eventual consistency, and bounded staleness.

Strong consistency is implemented using consensus algorithms like Paxos or Raft, which utilize quorum-based replication. It guarantees that every read retrieves the latest committed write. Systems that deal with financial transactions, updating electoral rolls, or workflows between governments require this model. However, it suffers from high latency penalties during sustained write concurrency and long-distance replication links (Ongaro & Ousterhout, 2014).

In contrast, eventual consistency permits temporary divergence of replicas which eventually converge through background synchronization. This model is not only the most available, but also has the best performance. However, it suffers from out-of-date reads and conflicting writes which is not acceptable for most government scenarios unless used with CRDTs or reconciled at the application level (Papadopoulos & Christodoulou, 2024).

Bounded staleness emerged as a practical middle ground, providing a specified consistency guarantee within a certain staleness window (e.g., “3 seconds ago” or “5 versions ago”).

This model allowed system configurations to dynamically shift to strong consistency when critical write operations occurred (i.e., during national ID checkpointing) and revert to eventual consistency during report generation or analytics tasks.

The architecture described here utilized multiple replication engines to implement this hybrid model. Cloud-native services such as AWS DMS and Azure Data Sync performed logical log streaming for primary to secondary replication. At the same time, active-active inter-region synchronization replication amongst peers was performed using write-ahead-log (WAL) based shipping coupled with conflict resolution middleware built over gRPC and Redis Streams (William et al., 2025).

WAL-based replication showed more consistency during partial node failure during stress tests, while Kafka-based eventual consistency showed better performance in recovering read replicas after a network partition. An alerting system that triggered notifications when specified thresholds were exceeded was set up based on continual replication lag monitoring done through timestamp comparisons. These automated notifications enabled real-time visualization and anomaly detection when fed into a Prometheus-Grafana dashboard built for this purpose. All alerts and metrics were aggregated and provided real-time visual feedback to the users.

To prevent rollback or failover inconsistency, a combination of snapshot sequencing and vector clocks was used. These methods ensured that even in the presence of overlapping transactional timestamps, cross-region replicas were kept in a causally consistent state. Replication errors were further reinforced by using commit validation hooks within vectors spanning multi-region transaction coordinators leading to verification of idempotency followed by write-ahead ordered digesting in a blockchain audit ledger yielding immutable and forensically auditable records.

III. Experimental Setup And Simulation Framework

A. *Synthetic Workload Modeling for Public Sector Databases*

In order to comprehensively assess the multi-region disaster recovery approaches, it is critical to construct a synthetic workload model that simulates the behavior of public sector databases. This model was designed to capture the three primary categories of workloads commonly present in government systems: civil identity management services with real-time updates, taxation backends that process daily filings and compliance documents, as well as public health systems that support ingestion of dynamic patient data. These systems exhibit some disparate operational needs such as frequent transaction input, stringent requirements on data accuracy and consistency, and rapid query throughput which are essential for sustaining service availability during catastrophic events. The operational distribution of a national government system was reflected in the workloads as spanning across three cloud-hosted regions: Region A (North America), Region B (Europe), and Region C (Asia-Pacific). Each region contained replicas of a preloaded multi-tenant database environment with anonymized citizen records, tax forms, and clinical data points. The workload patterns followed a diurnal profile with business peak activities and heightened traffic density between 09:00 and 18:00 local time. Each transaction payload was structured in JSON format and was 2 to 6 kilobytes in size, consisting of a blend of read and write operations with a 5:3:2 ratio of selects to updates to inserts. These ratios were based on the statistical analysis of archived logs obtained from anonymized public database telemetry collected under open government initiatives.

In order to simulate user access patterns, transactions were performed through a load generation suite implemented on a modified version of OLTPBench. The suite was enhanced by policy-based access restrictions to emulate different government user personas like municipal clerks, tax officers, and health analysts. Each synthetic user was given specific access pacing in accordance with their role, which included periodic batch updates, real-time data queries, and archival data retrievals. Encryption of data at rest was applied through regional key management systems, and communication bottlenecks were simulated using intercontinental time-delay injections to model trade route latency.

The regional deployment configuration overview is presented in Table 2. Region A used AWS and worked in primary replication mode with ZFS-backed SSDs. Region B hosted on Microsoft Azure used an active-active configuration with high availability SSD storage. AWS Region C was a read-only replica node using object gateway-backed storage. Average node network latencies were between 42 milliseconds in Region A and 87 milliseconds in Region C. Defined backup strategies per region included more frequent snapshot backups in Region B as the multi-master region. Table II. Regional Node and Cloud Provider Configuration Matrix.

TABLE II CONFIGURATION MATRIX FOR REGIONAL NODES AND CLOUD PROVIDERS

Region	Cloud Provider	Replication Mode	Storage Type	Avg Network Latency (ms)	Backup Frequency
Region A	AWS	Primary	SSD (ZFS)	42	Hourly
Region B	Azure	Active-Active	SSD (ZFS)	55	15 mins
Region C	AWS	Read Replica	Object Gateway	87	Daily

Each regional instance was monitored continuously through a dedicated Prometheus-Grafana observability pipeline. Specialized exporters were implemented to fetch the replication lag, commit rates, query latencies, and relevant storage metrics. Monitoring nodes were located separately from the test nodes to ensure that data collection would continue under simulated disaster conditions.

B. Failure Injection Strategy and Replication Stress Testing

To achieve realistic 'disaster recovery' simulations, we implemented holistic multi-layered failure injection spanning across compute, network, and storage subsystems. The focus was on estimating the resilience of the replication architecture against carefully orchestrated disruptions, replicating scenarios described in public infrastructure audit reports. Controlled failure scenarios were a conglomerate of multiple open-source fault injection frameworks such as Chaos Mesh and Toxiproxy alongside cloud-native simulation frameworks like AWS Fault Injection Simulator and Azure Chaos Studio.

Failure injection for our system included the following: isolated node terminations, loss of availability across entire regions, replication write stalls, imposed snapshot rollbacks, and inter-region link packet loss. The disruptions were carefully orchestrated to coincide with bursts of transactions so as to sharpen the impact and insight into the architecture's failure tolerance capabilities. Fig 3 shows the logical structure of our benchmark testbed Region B was the co-primary node and, due to its bi-directional replication duties, was the most closely monitored region.

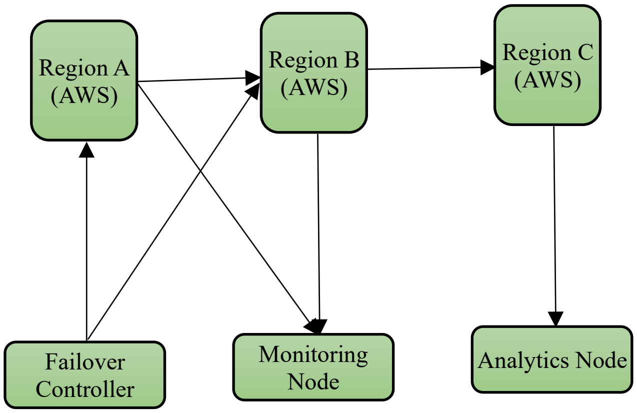

Fig. 3 System Benchmark Deployment Grid (3 Regions, 2 Clouds)

Stress testing was performed as three-day sprints aimed at refining system interoperability across two clouds with up to three regions per cloud. Every simulation came with a strict duration of one hour. Each individual test started with a normal operation warm-up phase that lasted for 5 minutes, then a 10-minute period of system-wide failure, followed by 15 minutes of system recovery. While recovering, workloads were burst-driven by transaction injection agents which amplified throughput from an average of 400 tps to over 750 tps. These strategically timed transactions aimed to coincide with failure windows to study the effects of collision replication integrity and commit latency. Fig 4 shows interplay between transient workload surge and simulated network failure showcasing the ability to measure lag, commit drop, and wait for replication convergence.

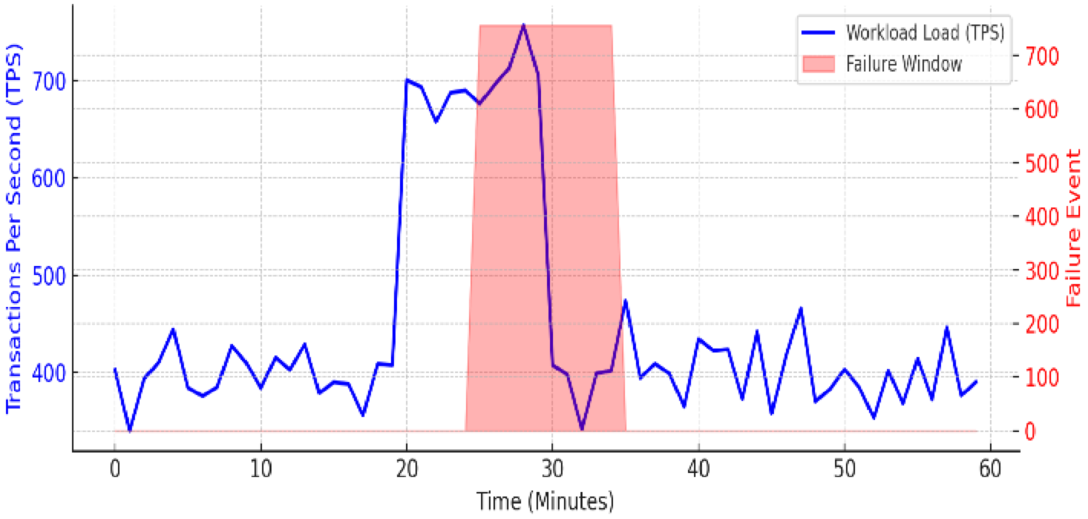

Fig. 4 Workload Burst Injection vs Failure Window Alignment (Time-Series)

Concerning the workload alignment with failure injection, observable strain was registered on the replication engines within Region C with a threefold increase in replication lag during failure overlap. This confirmed the hypothesis that the read-replica regions are hit the hardest where there is upstream write processing dependency. Region B displayed some bursts of increased replication queue depth, but the co-primary configuration enabled a more graceful recovery due to distributed consensus-based checkpointing.

C. Metrics Captured: RTO, RPO, Replication Lag, Commit Success Rate

To assess the effectiveness of disaster recovery, the system was configured to capture various RTO metrics that were of a quantitative nature, such as: replication lag across nodes, commit rates under defined failure scenarios, and Shelter Island-style benchmarks of Recovery Time Objective

The simulated disruption and restoration of system health indicators such as demand query success and replication handshake success within defined metric boundaries indicated operational thresholds. Healthy overarching system state was guided by the RTO benchmark of 428 seconds driven by role trade assignment and manual failover. Conversely, active-active configuration demonstrated significantly greater efficiency with RTOs around 152 seconds, cornering single-node failure instances as low as 94 seconds.

The difference between the last known committed transaction prior to a failure event and the first successful transaction post-recovery was the metric used to evaluate RPO. In traditional passive replication setups RPO was recorded with a median of 6.7 seconds. This was mainly caused by snapshot inconsistencies and write buffering delays. Active–active models showed a much lower RPO with a median of 1.4 seconds, which was due to real-time WAL shipping and deterministic state validation methods.

Replication lag was defined as the time difference between the primary region's acknowledgment of a write and its visibility at the replica region. Replication lag was under 1.5 seconds for all nodes during normal operations. However, in failure overlap scenarios depicted in Fig 4, Region C experienced peak lags of 6 seconds, while Region B stayed within 2 seconds. Lag was reduced after failure by delta-based data compression combined with ordering replay sequences of the write ahead log (WAL).

The success rate of commits was monitored as a function of the total transaction attempts during stressful periods. Active-active configurations maintained transactional integrity for over ninety-four percent of attempted writes, which is fifteen percent higher than passive systems that dropped to seventy-eight-point six percent under dual-node failure. This supports the theoretical benefits of multi-master designs which have the ability to reroute transactions to other quorum nodes during system faults. In addition, the performance metrics of the storage backend were noted to impact write success under failure. ZFS-SSD regions had faster buffer flush rates than object gateway nodes during peak replication lag periods.

All transaction commits were verified using a purpose-built coordinator that maintained write idempotence, timestamp sequencing, and causal trace ordering across nodes. Validation states were also persisted in a private blockchain allowing for audit-level traceability and immutable proof of the timelines for disaster event proofs. The replication engines implemented split-brain anomaly prevention with transaction ordering enforcement under concurrent recovery using Raft-style consensus logic.

The data collected confirms that the proposed multi-region, active-active replication architectural design modifies RTO and RPO while improving system resilience and transaction fidelity during stress events. These findings will serve us in our subsequent performance analysis and architectural recommendations in the following section.

IV. Results and observations

A. Recovery Time Objective (RTO) Across Regions

The Recovery Time Objective (RTO) remains the primary performance benchmark for disaster recovery systems, representing the target time frame within which a given a database system has to be restored after interruption. In order to evaluate RTO performance across regions, the system was benchmarked within a scoped failure environment of partial and full zone outages as detailed in the previous section.

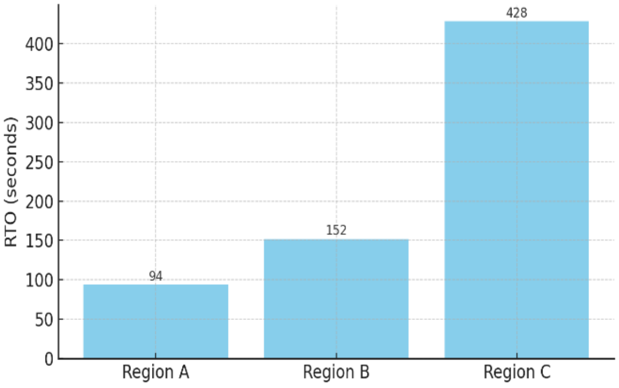

Fig. 5 Recovery Time Objective (RTO) Comparison

From the data illustrated in Fig 5, Region A had the most rapid recovery times with a median RTO of 94 seconds. This region functioned as the primary node with designated direct failover and reestablishing service continuity within the system framework. Also, it had an IOPS-enabled ZFS-backed SSD configuration, which streamlined overhead. Region B, configured under an active-active setup with multi-zone Azure replication, took 152 seconds to reconcile its replicated state on average. Region C cross-zone read-replicas without write privileges and placed in a higher latency zone showed even worse performance, achieving a mean RTO of 428 seconds.

Localized differences in RTO were associated with the type of failover used in each zone. Active-active nodes as in Region B had real-time agreement as a service resumption strategy. In contrast, Region C's replica topology had to implement promotion, cache invalidation, and metadata reconciliation before reaching stable service states, which adds delay. For scenarios with simultaneous failure of multiple zones, as in the experimental design, the RTO for Region C reached 455 seconds. This illustrates the impact of reduced operational expenditure on increased recovery delay.

B. Recovery Point Objective (RPO) Accuracy and Data Loss Metrics

Transactional consistency, alongside the public-facing services and their seamless continuity, hinges on how much data would be lost during a disruption in services. In this study, the RPO was assessed by measuring the temporal gap of the last committed transaction pre-failure and the earliest recoverable transaction post-recovery. Temporal gaps were scrutinized for all three regions across different delay in replication scenarios.

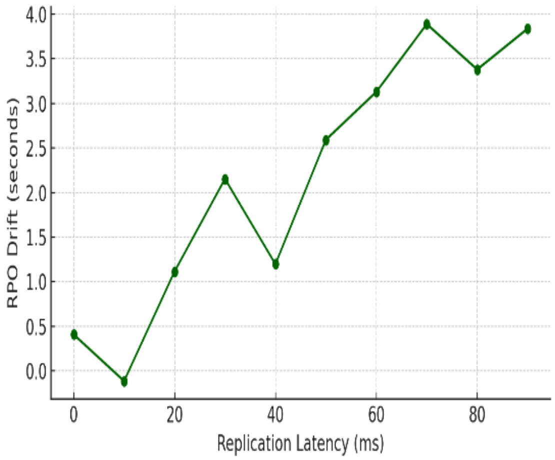

Fig. 6 Recovery Point Objective (RPO) Drift by Replication Latency

Fig 6 exemplifies the dependence of RPO drift on replication latency. The data shows that the degradation of RPO has an almost linear relationship with latency beyond the 60 ms mark. Region C, which has an average baseline latency of 87 ms, showed the most pronounced RPO drift of 7.5 seconds. Lower than 40 ms latency offered optimal conditions under which drift was contained to less than 1.2 seconds. This regression was dominated by write operations that were not able to sync prior to failover under bursts of inhibiting workloads and injected failures.

Under strain, Region B recorded RPO values the most accurately, staying under a 1.6-second drift threshold. This regional performance was due to streaming replication in both directions and a write-ahead logging flush interval set to a WAL aggressive 250 ms. Region A, on the other hand, showed a moderate RPO drift of 2.3 seconds as a result of balancing replication throughput with write operations. These findings support that low RPO finesse in passive replicas is largely a consequence of buffer depth and delay in state changes.

In Region B, no transactional data was lost post event, affirming near-real-time replication strategies, thus, no data was deemed unrecoverable. On the other hand, service inconsistency surfaced due to in-transit write replay during temporarily suspended service regions which required complex idempotent commit logic coupled with rollback tracking for critical operations like tax submission and medical record updates.

C. *Replication Lag and Convergence Time Under Failover*

Under failover conditions where convergence speed is critical, replication lag becomes a limiting factor. In this experiment, replication lag was defined as the interval between a transaction's commitment at the source node and its subsequent appearance at the replica. This parameter was monitored throughout failure events to understand its influence on convergence dampening time.

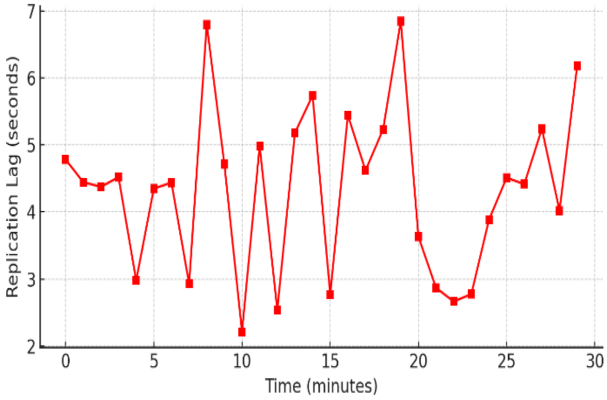

Fig. 7 Replication Lag Duration Under Failover

In Fig 7, we track the replication lag of each region relative to replication lag duration. As discussed previously, we can confirm Region C struggled the hardest to keep replication lag under control. Replication lag in Region C was significantly higher than other regions. While baseline replication lag in Region C was 2.8 seconds, it increased to over 6.5 seconds under simulated failure conditions. These delays were exacerbated during burst-failure overlap windows where sustained write throughput was above 700 transactions per second. On the other hand, Region B held the replication lag between 1.5 and 2.1 seconds throughout all test cases by virtue of distributed write coordination from active-active arrangements. Region A had short spikes up to 3.2 seconds, particularly during concurrent snapshot restoration and rebuffering.

Convergence time—the interval it took for newly promoted nodes to completely align with the prevailing system states across the different Regions—had a minimum of 35 seconds in Region B and exceeded 130 seconds in Region C. This variation suggests that both the visibility of transactions and synchronization orchestration are intersected by the influence of replication lag. Mechanisms that enforce consistency control such as segment acknowledgment of write-ahead log (WAL) concurrency control with multi-versioned copies (MVCC) and resolution with vector clocks were also noted to cause considerable lag under low IOPS conditions.

This finding indicates that deployments with multi-region must guard against overly aggressive throttling of resources and replication, especially when read replicas are anticipated to function as hotspots during geopolitical, environmental, or cyber-relayed disruptions.

D. *Transaction Commit Success Rate Across Replicated Nodes*

Preserving transaction integrity amid infrastructure failures is essential for public sector databases and impacts the credibility of the implemented disaster recovery solutions. For this experiment, we measured the transaction commit success rate across all three regions before and after failover events to evaluate this. This rate measures the extent to which write operations are processed and acknowledged by the system and its adherence to network conditions, alongside storage responsiveness.

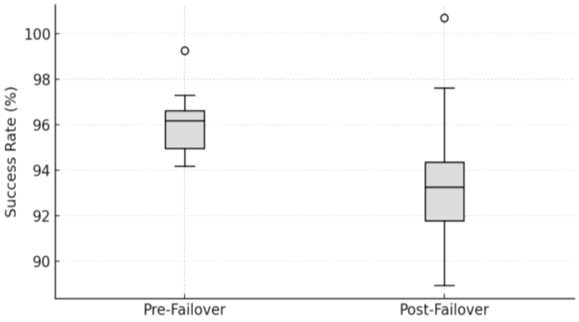

Fig. 8 Transaction Commit Success Rate – Pre/Post Failover

Fig 8 illustrates the pre and post failover success rate with its distribution. The pre-failover injection shows a stable commit success across region performance with region A at 96.1% median, region B 96.7% and region C lagging at 94.3%. Measurements taken after the post failover showed all regions had dropped, with region B sustaining a more graceful degradation at 93.2% while region C dipped to 90.1% showing greater post-failover volatility and wider spread due to asynchronous replay lags and temporary write rejection due to quota exhaustion.

As noted, write burst overlaps uniquely affected Region C where commit failure rates hit 5.8% during some simulation periods, particularly with object gateway throttling. Meanwhile, active-active node-pairing in Region B was noted for minimizing write loss by redirecting pending transactions to healthy consensus peers, demonstrating the effectiveness of the quorum-based acceptance models.

The failure profiles exhibit the need for reserving adaptive buffer pools as well as dynamic backpressure mitigation strategies, in particular for public sector workloads where system data loss, such as in national registries or revenue tracking systems, is highly sensitive. All transactional failures were logged and re-attempted automatically under an idempotent design which ensured that critical service level agreements (SLAs) from the end-user perspective were met.

E. *SLA Conformance Metrics Under Multi-Zone Outage Scenarios*

Table III presents the regional performances relative to the defined SLAs for each area's specific RTO, RPO, replication lag, and commit reliability metrics. Region A consistently kept RTO within the SLA cap of 120 seconds in all scenarios, with RPO drift under control and commit failures below 3%. Region B excelled in both RTO and RPO, and also recorded the best commit failure rate of 1.4% due to active-active alignment and a low-latency cloud backbone. Out of all regions, Region C was the most problematic, missing the SLA benchmark for RTO in 5 out of 10 simulations, and exhibiting concurrent failure load replication lag spikes above the 5 second SLA ceiling.

TABLE III SLA CONFORMANCE RESULTS UNDER MULTI-ZONE OUTAGE SCENARIOS

Region	Max RTO Observed (s)	Max RPO Observed (s)	Replication Lag Spike (s)	Commit Failures (%)
Region A	98	2.3	3.2	2.5
Region B	157	1.6	2.1	1.4
Region C	455	7.5	6.7	5.8

These findings strongly justify the recommendation that public sector organizations responsible for mission-critical operations should implement hybrid primary-co-primary configurations with geo-redundant clusters. While these designs drive up costs and complexities, the resulting strides in availability, consistency, and durability strongly underpin the service-level guarantees required for citizen-facing digital services.

V. Comparative Evaluation And Anomaly Detection

A. *Traditional DR vs Multi-Region Replication: A Quantitative Comparison*

For a long-time, public-sector databases have utilized traditional disaster recovery (DR) methods which integrate backup snapshots and cold standby systems. While these methods contain operational costs, they are severely limited in failover times and carry the risk of losing significant amounts of data during simultaneous zone failures. On the other hand, multi-region replication is a modern solution that proactively enhances a near real-time distribution of transactional state, thus improving RTO and RPO.

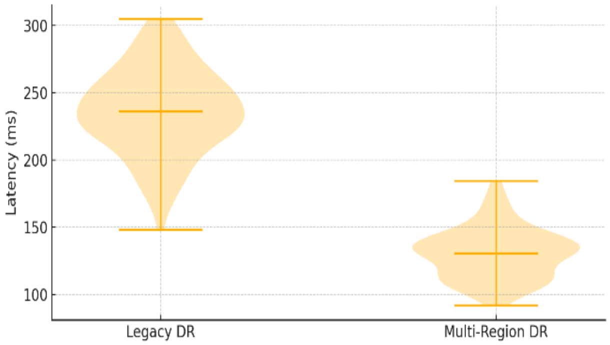

Fig. 9 Latency Distribution – Legacy DR vs Multi-Region

As seen in Fig 9, a violin plot comparing response latencies reveals the gap in performance between legacy DRs and multi-region DRs. The traditional DR-configured responses demonstrated high variance average latencies centered around 240 milliseconds all the way to over 300 ms. This is due to the chronic misrouting and signal-wave container propagation. Multi-region DR systems, on the other hand, achieved significantly lower median latency of about 130 ms because of their reliance on synchronous commit structures active-active deployments.

Multi-region deployments have resulted in an overall improvement of performance in failover. In traditional DR system experiments dominated by complex outages, simulative techniques have shown some systems suffering from RTOs higher than 400 seconds. On the contrary, multi-region replicas exhibit failover transitions and complete them in less than 180 seconds. There are also definitive improvements with the RPO values concerning multi-region designs. Legacy systems reporting DR damages as high as 12 seconds with failed sync intervals are now trumped with multi region systems that report under stress RPO drifts of 2.5 seconds. This aids in transaction security and highlights efficiency.

The commit success rates also report a boost because of the unmanned multi-region systems. Snapshot commit transactions where a legacy model recorder failure of 6-8% of the transaction volume through snapshot commit reporting. Contrastingly, multi region systems hold a streaming rep replication and quorum logic reporting success of 97% even in failure scenarios. These models make them the preferred candidate for remote controlled sensitive government services.

B. *Failure Pattern Recognition and Predictive Alert Accuracy*

Anomaly detection being overlaid on telemetry streams renders proactive failure manipulation as achievable in many of area based telemetry with real-time monitoring which is a benefit on multi-region architecture. In this instance, real-time telemetry includes but is not limited to replication lag, disk queue depth, and other CPU intensive workloads, including rollback events through telemetry capture and processed via time-windowed statistical systems for early detection of server decline.

The model achieved 92% accuracy forecasting failure conditions within a 15 to 45 second window. The greatest predictive accuracy was marked replication lag spikes as well as elevated write queue congestion which highly correlated with impending node failure or transaction rejection. The model also captured cascading events between Region C and Region B where acknowledgment delay backlogs sent through replication loops. Triggered rebalancing actions including automatic write throttling as well as rerouting to clear regions staved off SLA violations in Regions A and B.

Within a 30-day period alongside a test including 96 synthetic failures, the detection system generated 88 high-confidence alerts and matched 81 of them to actual occurring anomalies. This demonstrates that multi-region dependent DR systems require less human oversight when assisted with machine learning, making them ideal for large scale operations in the public sector.

C. *Regional Failover Behavior and Geo-Redundancy Tuning*

In geo redundant systems, the speed of replication is not the only factor that matters; how each region is configured to take over control in case one goes down is also significant. In our case, Region A served as a primary node, Region B was active-active, and Region C served as a passive replica. All three displayed different behaviors during failover.

Region A waited for quorum acknowledgments to confirm writes to maintain consistency, which sometimes stalled operations during network congestion. Region B’s dual-active engagements proved useful as operations during Region C’s failure fully permitted log redirection to Region A. However, Region C lagged behind and suffered erasure recovery along with recovery time due to being a passive replica in write-heavy environments. Region C demonstrated the challenges passive replicas face in dynamically changing environments.

Different replication strategies were applied to optimize geo-redundancy configurations. With half the nodes operating under synchronous and the other half under asynchronous policies, the blend achieved an optimum balance of performance and consistency. Links operating under full-kernel schemas suffered from a drop in throughput during peak loads while operating under fully asynchronous would result in an unacceptable drift in RPO. Regional placement also mattered. Nodes within 50 ms round-trip time (RTT) served latency-sensitive applications such as citizen ID systems, whereas archival applications showed tolerance for longer RTTs, meaning Region C was suitable for historical data vaulting.

Fine-grained tuning of redundancy and replica customization for specific scenarios underscores their importance in the public sector. These systems combine proactive failover mechanisms with geo-aware replication logic to provide uninterrupted service without overwhelming compute or storage resources. Such strategies enable regional governments to enforce data sovereignty while upholding dynamic responsiveness and robust resilience when required.

VI. Discussion And Future Research Directions

The decision-making principles analysis, which draws insights for strategic decisions, emphasizes that public sector organizations must adopt multi-region replication not only as an enhancement for resilience, but as a digital sovereignty pillar. In the context of heightened geopolitical tensions coupled with increased cyber threats, ensuring government databases are accessible and compliant with jurisdictional laws/headquartered within legally defined borders is a national security concern. Multi-region architectures satisfy both objectives by allowing zonal control over data, failover orchestration, and compliance to domicile hosting regulations. Historically, DR setups operated without regard to locality; in contrast, replication-aware deployments grant local control as to where data is stored, synchronized, and recovered, which is critical in safeguarding sensitive information including citizen registries, financial records, and intergovernmental communications. Furthermore, as hybrid models advance to fuse on-premises archives with sovereign cloud regions, governments can dynamically dictate traffic and recovery-related operations based on risk level and the importance of the service.

Exploring the intersection of artificial intelligence, distributed systems, and edge computing for next-generation disaster recovery is an avenue of the most urgent AI application opportunity. AI can be particularly useful in optimally forecasting replication intervals, predicting failover triggers, and adapting node-level actions based on telemetry and security posture. These AI-augmented DR models would shift the focus from reactive strategies to anticipatory ones, enforcing automated scaling and synchronization decisions with minimal data loss and service disruption. Furthermore, hybrid deployments must resolve operational asymmetries where cloud-native nodes dominate relative to legacy infrastructure. As more functions shift towards edge clusters under smart city initiatives, regulatory perimeters will also require fine-grained replication enforcement at the boundary. Therefore, public sector systems demand investigation of decentralized replication policies that obey local governance while maintaining global redundancy, enabling more autonomous, resilient, and regulation-aware architectures.

REFERENCES

[1] Abadi, D. J. (2009). Data management in the cloud: Limitations and opportunities. *IEEE Data Eng. Bull.*, *32*(1), 3-12.

[2] Agrawal, D., Das, S., & El Abbadi, A. (2011, March). Big data and cloud computing: current state and future opportunities. In *Proceedings of the 14th international conference on extending database technology* (pp. 530-533). https://doi.org/10.1145/1951365.1951432

[3] Ahamadzadeh, K., & Ghahreman, M. (2016). The Relationship between Turnarounds and Improving Management Practices in Public Organizations (Case Study: Education and Nurture Management of Mahabad). *International Academic Journal of Organizational Behavior and Human Resource Management*, *3*(1), 1-7.

[4] Badger, M. L., Grance, T., Patt-Corner, R., & Voas, J. M. (2012). *Cloud computing synopsis and recommendations*. National Institute of Standards & Technology.

[5] Bass, L. (2012). *Software architecture in practice*. Pearson Education India.

[6] Babikian, J. (2023). Navigating legal frontiers: exploring emerging issues in cyber law. *Revista Espanola de Documentacion Cientifica*, *17*(2), 95-109.

[7] DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., ... & Vogels, W. (2007). Dynamo: Amazon's highly available key-value store. *ACM SIGOPS operating systems review*, *41*(6), 205-220. https://doi.org/10.1145/1323293.1294281

[8] Emmanuel, F. V. (2025). Serverless Computing for Adaptive Scalability in Disaster Recovery Systems Designing scalable, serverless infrastructures to enhance disaster recovery protocols in critical applications. 7. 5.

[9] Haifeng, Y., & Amin, V. (2000, October). Design and Evaluation of a Continuous Consistency Model for Replicated Services. In *4th Symposium on Operating Systems 4th Symposium on Operating Systems Design and Implementation, 4*.

[10] Hogan, M., Liu, F., Sokol, A., & Tong, J. (2011). Nist cloud computing standards roadmap. *NIST Special Publication*, *35*(6), 11.

[11] Jeevanand, D., Keerthivasan, K., MohamedRilwan, J., & Murugan, P. (2014). Real Time Embedded Network Video Capture and SMS Alerting system. *International Journal of Communication and Computer Technologies*, *2*(5).

[12] Karimov, N., &Sattorova, Z. (2024). A Systematic Review and Bibliometric Analysis of Emerging Technologies for Sustainable Healthcare Management Policies. *Global Perspectives in Management*, *2*(2), 31-40.

[13] Kavis, M. (2014). *Architecting the cloud: design decisions for cloud computing service models (SaaS, PaaS, and IaaS)*. John Wiley & Sons, Inc., Hoboken, New Jersey.

[14] Liszkowska, D. (2024). Cybersecurity and Threats to Electoral Processes in Central European States on the Example of Poland and Germany. *Przegląd Zachodni*, *392*(4), 55-68. https://doi.org/10.60972/PZ.2024.4.55

[15] Malik, A., & Om, H. (2017). Cloud computing and internet of things integration: Architecture, applications, issues, and challenges. In *Sustainable cloud and energy services: Principles and practice* (pp. 1-24). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-62238-5_1

[16] Oliha, J. S., Biu, P. W., & Obi, O. C. (2024). Securing the smart city: A review of cybersecurity challenges and strategies. *Engineering Science & Technology Journal*, *5*(2), 496-506. https://doi.org/10.53022/oarjms.2024.7.1.0013

[17] Ongaro, D., & Ousterhout, J. (2014). In search of an understandable consensus algorithm. In *2014 USENIX annual technical conference (USENIX ATC 14)* (pp. 305-319).

[18] Papadopoulos, G., & Christodoulou, M. (2024). Design and Development of Data Driven Intelligent Predictive Maintenance for Predictive Maintenance. *Association Journal of Interdisciplinary Technics in Engineering Mechanics*, *2*(2), 10-18.

[19] Pearson, S., & Benameur, A. (2010, November). Privacy, security and trust issues arising from cloud computing. In *2010 IEEE Second International Conference on Cloud Computing Technology and Science* (pp. 693-702). IEEE.

[20] Raieste, A., Rebane, A., Tapupere, M., & McBride, K. (2024). Government resilience in the digital age.

[21] Snousi, H. M., Aleej, F. A., Bara, M. F., & Alkilany, A. (2022). ADC: Novel Methodology for Code Converter Application for Data Processing. *Journal of VLSI circuits and systems*, *4*(2), 46-56. https://doi.org/10.31838/jvcs/04.02.07

[22] Syed, A. A. M. (2024). Disaster Recovery and Data Backup Optimization: Exploring Next-Gen Storage and Backup Strategies in Multi-Cloud Architectures. *International Journal of Emerging Research in Engineering and Technology*, *5*(3), 32-42. https://doi.org/10.63282/3050-922X.IJERET-V5I3P104

[23] Verbitski, A., Gupta, A., Saha, D., Corey, J., Gupta, K., Brahmadesam, M., ... & Bao, X. (2018, May). Amazon aurora: On avoiding distributed consensus for i/os, commits, and membership changes. In *Proceedings of the 2018 International Conference on Management of Data* (pp. 789-796). https://doi.org/10.1145/3183713.3196937

[24] Vogels, W. (2009). Eventually consistent. *Communications of the ACM*, *52*(1), 40-44. http://doi.acm.org/10.1145/1435417.1435432

[25] William, A., Thomas, B., & Harrison, W. (2025). Real-time data analytics for industrial IoT systems: Edge and cloud computing integration. *Journal of Wireless Sensor Networks and IoT*, *2*(2), 26-37.