Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Self-Healing Cloud Databases: Automatically Resolving Outages for Non-Stop Business

Veeravenkata Maruthi Lakshmi Ganesh Nerella
Sr. Database Administrator
Summerfield, NC, USA
ORCID: https://orcid.org/0009-0008-6149-846X

ARTICLE INFO
Received: 05 Apr 2025
Revised: 15 May 2025
Accepted: 25 May 2025

ABSTRACT
In the modern digital era, cloud databases serve as the foundational layer for business-critical applications, demanding uninterrupted availability, dynamic scalability, and fault tolerance. The concept and architecture of self-healing cloud databases emphasize their role in minimizing downtime and ensuring business continuity. Traditional reactive fault recovery approaches are increasingly insufficient in handling today's complex, distributed, and real-time cloud systems. Self-healing databases leverage automation, real-time monitoring, and artificial intelligence (AI) to proactively detect, diagnose, and autonomously recover from faults. The study details enabling technologies such as predictive analytics, observability frameworks, container orchestration (e.g., Kubernetes), and DevOps-SRE integrations that make automated remediation possible. Additionally, techniques like load balancing, elastic resource provisioning, and auto-scaling are examined to demonstrate resilience in dynamic workloads. such as anomaly classification complexity and cross-architecture generalization. Evaluation metrics including MTTR, MTTD, fault recovery rate, and uptime percentage are presented to quantify effectiveness. The paper concludes that the convergence of AI, cloud-native infrastructure, and proactive automation is essential for the next generation of resilient database systems capable of responding to operational anomalies autonomously, thus supporting continuous service availability for non-stop business operations.

Keyword: Self-Healing Cloud Databases, Fault Tolerance, Automated Recovery, Outage Recovery, Resource Elasticity, business database SQL, NoSQL.

Introduction

In today's highly digitized and interconnected landscape, uninterrupted access to data has become a critical backbone for business success. Enterprises across industries now operate in real time, rely on dynamic data exchanges, and demand seamless performance from their digital services. As cloud computing continues to dominate IT infrastructure, cloud databases have become indispensable in enabling scalable [1][2][3], SQL databases are relational and use a fixed schema distributed, NoSQL databases are non-relational and are optimized for unstructured or semi-structured data, offering flexible schema design, horizontal scalability, and high availability and on-demand data processing capabilities. and minimizing unplanned downtime has emerged as one of the foremost [4][5][6]. These effects are clear and dire including lost financial gains, declining productivity, customer churn, and damaged reputations due to outages in database [7].

The concept of self-healing cloud databases has become more popular in response to this rising worry. These systems are constructed to pinpoint, diagnose and repair defects automatically, without the intervention of people [8]. Relevant to traditional fault recovery processes, characterized by manual operations, reactive error notification, and slow administrative procedures [9], self-healing systems incorporate proactive automation and automated recovery procedures to react to failure in near-real-time [10]. This does not only decrease mean time to recovery (MTTR) but also enables organizations to ensure business continuity even in the face of infrastructure-level incidents or high-impact service

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

anomalies [11][12]. The need to support such autonomous recovery capabilities is rapidly gaining importance at the heart of self-healing cloud databases is a combination between monitoring, automation, and artificial intelligence (AI) [13]. Real-time observability systems offer real-time visibility into a system so that faults can be identified and anomalies categorized instantly. The responses are either scripted or policy-based, so automated tools can correct them quickly and immediately. In the meantime, intelligent recovery operations are improved through AI and machine learning to predict and reduce the possibility of failure using historical trends and contextual analytics. This layered model vents the cloud database management into a proactive, intelligent, and adaptive system that is capable of changing with the needs of operations [14][15]. These systems improve on the past occurrences and monitor to enable organizations to protect their services and resources in advance.

Self-healing extends into other critical aspects of cloud databases such as load balancing, resource elasticity, and data consistency [16][17][18]. As an example, when a node in a distributed database environment becomes saturated or unresponsive, traffic can be balanced between nodes dynamically using load balancers, and auto-scaling can be used to add resources to the system to restore performance levels. Essentially, methods involving data replication, pooling in standby nodes, backup recovery guarantee a high degree of accessibility and integrity of data, even with the presence of severe outages [19]. These self-managing operations are often orchestrated through microservices and containerized environments that enhance modularity, fault isolation, and independent recovery at service-level granularity [20][21]. Cloud-native tools like Kubernetes and service meshes further automate this orchestration, creating a resilient and continuously healing ecosystem.

A. *Structure of the Paper*

The structure of the paper is as follows: The availability and design of cloud databases are covered in Section II. Section III the concept of self-healing systems Section IV. discusses core self-healing mechanisms and enabling technologies in Section V, Literature review, Section VI, Conclusions and Future Work.

Cloud Database: Architecture and Availability

Cloud databases are scalable, on-demand data management systems hosted on cloud infrastructure, designed to support modern applications with high availability, flexibility, and resilience. Their operation is based on distributed environments that use virtualization, containerization, and orchestration technologies like Kubernetes, as opposed to conventional databases [22]. These systems are deployed in various models public, private, hybrid and support different data types through SQL, NoSQL, or NewSQL architectures. High availability in cloud databases is achieved through techniques like data replication, automated failover, clustering, and distributed storage [23]. However, their distributed nature also introduces challenges such as system complexity, fault propagation, and configuration errors, which can lead to performance issues or outages.

B. *Architecture and Deployment Model*

1) *Cloud Architecture*

The four tiers of functionality that comprise the cloud architecture are shown in Figure 1.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Fig. 1. Cloud Architecture Model

2) Infrastructure as a Service (IaaS):

IaaS offers the greatest feature set and adaptability among cloud services [24]. It essentially provides and controls a fully virtualized computer system via the Internet.

Examples of IaaS: Microsoft Azure, AWS, Cisco Meta-cloud.

3) Platform as a Service (PaaS):

The creation, testing, deployment, maintenance, and updating of software products may be facilitated by PaaS. It incorporates DBMS, middleware, operating systems, and development tools into its infrastructure, which is identical to that of IaaS [25].

Examples of PaaS: AWS Elastic Beanstalk, Apache Stratos, Google App Engine, Microsoft Azure.

4) Function-as-a-Service (FaaS)

FaaS allows customers to reply code reactively and does away with the need for them to pre-allocate processing resources. The infrastructure, which is overseen by the cloud service provider, is accessible to the customer.

Examples of FaaS: AWS Lambdas, Azure Functions.

5) Software as a Service (SaaS):

SaaS refers to an online subscription model for pre-built software applications [26]. The infrastructure, operating systems, middleware, and data needed to provide the application are managed by the SaaS provider. SaaS apps enable companies to easily launch and grow their operations.

Examples of SaaS: Microsoft Office 365, Salesforce, Cisco WebEx, Google Apps.

a) Cloud Computing Deployment Model

There are 4 main categories for cloud computing deployment models: 1) Public Cloud, 2) Private Cloud, 3) Hybrid Cloud and 4) Community Cloud [27] in Figure 2. The following provides a quick overview of every cloud deployment model.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

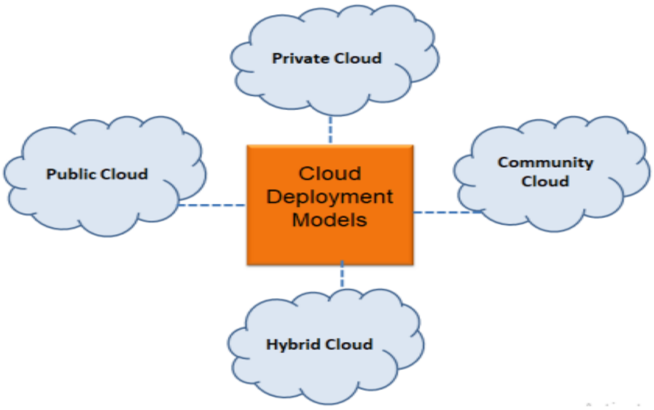

Fig. 2. Cloud Deployment Model

- **Public Cloud Model:** It makes things easier so that the general public may easily utilize the services and systems. A third party that provides cloud services is legally considered the owner and is responsible for accounting for all physical resources.
- **Private Cloud Model:** An individual company is the operator of the Private Cloud [28]. Compared to other cloud models, this one provides better security since only authorized users may access the organization's system.
- **Hybrid Cloud Model:** The architecture that combines public and private clouds is known as a hybrid cloud paradigm. It allows for the simultaneous use of several deployment models.
- **Community Cloud Model:** A select group of organizations shares the community cloud model, which addresses the issues raised by that community. It is feasible for organizations or third parties to handle the system internally.

C. *Type of Cloud Databases*

Database as a service, which is further separated into three main types, is offered by cloud database service providers [29]. Databases may be either rational or non-rational, and there are operating virtual machines that come preloaded with local database applications like SQL. An abundance of distributed, open-source, and horizontally scalable non-relational DBMSs are available, such as Aerospace, MongoDB, Cassandra, and OrientDB. Important to both relational and non-relational database systems is the speed with which massive volumes of data can be stored and retrieved. Figure 3 depicts the two kinds' structures.

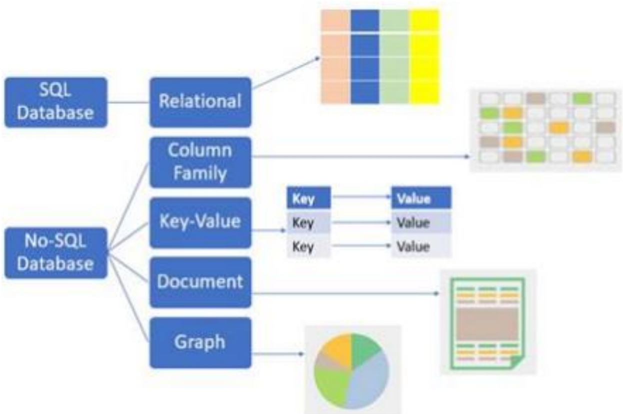

Fig. 3. SQL and NoSQL Structure.

There are two main types of databases: SQL and NoSQL

1) SQL

The database language SQL, which stands for Structured Query Language, was created as a high-level standard interface for the majority of databases and is often utilized for DDL and DML relational system management (RDBMS) [30]. The relational template databases are known to use SQL as its questionnaire language. The MySQL, MS-SQL Server, and Oracle Databases

2) NoSQL

Furthermore, SQL is a DBMS that is not relational. NoSQL databases are widely used and include MongoDB, Cassandra, CouchDB, and HBase [31]. Furthermore, leading internet companies like Amazon, Google, and LinkedIn have been at the forefront of database innovation and expansion, particularly in light of the present trend towards "Big Data."

a) Document Stored

The main idea is the distinction between document-oriented databases. The JSON format is utilized by MongoDB, one of the most popular document-based databases; additional users of XML and YAML storage include BaseX and YAMLDB.

b) Key-Value Stored

Key-value is the easiest of these four databases to grasp. The DBMS stores hash tables, or knowledge. Common central value databases include Redis, Berkeley DB, and Level DB.

c) Column Stored

Instead of storing tables by rows, a column-driven database uses columns. DBMS are mostly responsible for maintaining this differentiation, which often permits column-oriented databases to be interoperable with traditional row-based databases.

D. Key Charateristic of Cloud Database

Cloud databases exhibit several defining characteristics that distinguish them from traditional on-premises databases and enable them to support scalable, reliable, and flexible operations:

- **Managed Services and Automation:** The majority of cloud databases are provided as fully managed services, which lower operational costs by automatically patching, backing up, scaling, and monitoring.
- **Global Accessibility:** there is an internet connection, cloud databases may be accessed, and many of them enable many regions for low latency and data residency compliance.
- **Multitenancy:** A cloud enables the simultaneous use of services by several users [32]. Even though those users are using the same cloud resources (host, network, and applications), there are isolated inside their own virtual application instances.
- **Scalability:** Cloud computing's scalability is one of its defining characteristics. Providers may easily increase the number of nodes and servers in the cloud by making small changes to the underlying software and architecture.
- **Virtualization:** Cloud computing makes it possible for consumers to access services from any location and on any device. Instead of a tangible item, the resources needed are uploaded to the cloud.

Concept of Self-Healing in Cloud Environments

The original meaning of the phrase "self-healing" was a system's capacity to spontaneously restore itself to a necessary condition when damaged, with little to no human involvement. Figure 4 shows only a few examples of its numerous current applications and more widespread acceptance: self-healing

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

computer networks, smart materials and structures, peer-to-peer systems in the self-healing sector, and many more. It is possible to see all currently existent systems with self-healing capabilities as interconnected cyber-physical systems.

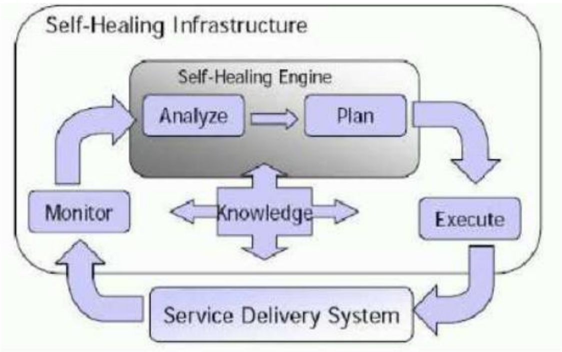

Fig. 4. Self-Healing System

Many different types of redundancy have long been used in self-healing systems, including hardware, software, failure prediction and diagnosis, fault recovery, and many more. In addition, AI is primarily used to aid self-healing systems in detecting and testing the repair of complexities that would normally be beyond the capabilities of a human expert.

E. Key Point of Self-Healing Techniques

There is some key self-healing technique are discussed below:

1) Auto-Scaling

The ability to automatically scale up or down resources (such storage or processing power) in a cloud environment is known as auto-scaling. If the system detects increased load or resource consumption, it scales up by provisioning additional instances.

- **Application:** Auto-scaling is widely used for web applications, containerized workloads Cloud providers like AWS, Google Cloud, and Azure offer native auto- scaling solutions.

2) Automated Failover:

Automated failover is the process by which cloud systems automatically detect a failure in one or more components and reroute traffic or workloads to healthy, redundant resources without human intervention [33]. Failover mechanisms are critical for maintaining uptime and ensuring high availability

- **Tools:** Load balancing and DNS routing mechanisms (e.g., AWS Elastic Load Balancer, Azure Traffic Manager) facilitate automated failover in cloud environments

3) Self-Repairing Systems:

Self-repairing systems are capable of detecting failures or misconfigurations and automatically repairing them, often by restarting services, redeploying containers, or provisioning replacement instances.

- **Tools:** Kubernetes, Docker Swarm, AWS Elastic Beanstalk.

4) Proactive Remediation:

The goal of proactive remediation is to head off problems in their tracks by using predictive algorithms to scan for and eliminate possible causes. These may involve maintenance such as patching vulnerabilities, moving more resources into expecting heavy traffic or setting configurations according to past data and performance history.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

- **Tools:** Cloud-native monitoring tools like AWS CloudWatch, Google Cloud Operations Suite, and Azure Monitor help identify early warning signs and trigger automated actions.

5) Health Checks and Auto-Restart:

Health checks serve to constantly check the state of health of cloud instances, applications or services. When an unhealthy instance is detected (e.g., due to a crash, overload, or performance degradation), the system can automatically restart or replace the instance to restore normal operations.

- **Example:** AWS EC2 instances and Kubernetes pods support health checks to verify the health of running services

F. Role of Automation, Monitoring, and AI

In self-healing cloud databases, automation, monitoring, and artificial intelligence (AI) form the core pillars that enable proactive and efficient system recovery:

1) Automation for Rapid Recovery

Cloud databases with automated workflows enable the system to start remedial activities without human assistance. Restarting failed services, reallocation of resources, or relocation to backup nodes are some examples. Automation guarantees company continuity and drastically cuts down on recovery time.

2) Real-Time System Monitoring

The information about the health of system, latency and use of the resources is collected through the continuous monitoring technologies. The health indicators on CPU usage, memory load, latency, and I/O activity are gathered using continuous monitoring tools. Such metrics aid in determining anomalies or loss in performance. This data provides the basis to detect irregularities, track performance, and initiate the process of automated healing when certain thresholds have already been crossed.

3) AI-Powered Predictive Intelligence

Machine learning and artificial intelligence algorithms analyze the patterns in the activity of machines to predict potential faults allows detecting patterns, anomalies, and the root of the problem. The system learns and grows in both smarter and robust ways because of the capability of AI to prioritize remedial works, make it possible to investigate on root cause and cause changes to future actions based on event data.

G. Evaluation Metrics for Self-Healing Effectiveness

In the evaluation of the effectiveness of self-healing mechanisms in cloud databases certain performance measures are employed. These metrics help quantify how well a system can detect, respond to, and recover from failures without human intervention in shows in Table I.

TABLE I. METRIC OF SELF-HEALING PERIMETER

Metric	Description	Objective
Mean Time to Detect (MTTD)	Average time taken to identify faults or anomalies.	Detect issues early to trigger faster healing
Mean Time to Recovery (MTTR)	Average time to restore system functionality after a failure.	Minimize downtime and disruption

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

Fault Recovery Rate	Percentage of faults resolved without manual intervention.	Maximize automation reliability
System Uptime (%)	Overall availability of the system over time.	Ensure continuous service delivery
Error Recurrence Rate	Frequency of the same fault reoccurring.	Improve long-term healing and root fixes
Performance Degradation During Recovery	Measure of system performance impact during healing.	Maintain service quality while recovering

H. *Benefits of Self-Healing Cloud Systems*

There are some benefits of self-healing cloud system are discussed in below:

1) *Reduction in Downtime Through Proactive Self-Repairing Mechanisms*

Self-healing clouds are deployed on predictive maintenance so that infrastructure can be monitored on a real-time basis, and problems are detected before breakdown. With the help of machine learning algorithms to preempt network issues like an overload of servers, network failures or hardware failures. In contrast, this proactive strategy substantively decreases unplanned downtime hence enabling businesses to operate smoothly even at the event of possible infrastructure outages.

2) *Cost Efficiency by Preventing Costly Repairs and Outages*

Self-healing systems can help keep operational costs down by identifying problems in the early stages of becoming serious failures that could otherwise have incurred costly emergency services or replacement of hardware [34]. Automated reallocation of resources, system optimization, and self-diagnosis will allow preventive work that will automatically resolve minor issues and limit human interventions. Through this, businesses reduce the costs of outages of the service when it is not planned.

3) *Enhanced System Reliability, Improving Customer Satisfaction and SLA Compliance*

Self-healing cloud systems improve the reliability of the services by making cloud infrastructures very highly available and consistent. The cloud providers are in a position to satisfy or even surpass their Service Level Agreements (SLAs) that specify expectations of uptime, performance and availability. This regular reliability can increase customer satisfaction since the clients will have fewer interruptions and have more certainty on the service quality.

Mechanisms for Self-Healing in Cloud Databases

Self-healing in cloud environments involves the means of ensuring continuous service availability, resiliency and reliability, through programmatically detecting, responding, and recovering faults or service failures. These processes reduce the amount of hands necessary to run them, reduce downtime, and improve the overall experience of using them. Self-healing capabilities are essential for cloud environments that demand high availability and fault tolerance, especially in distributed systems with multiple components spread across data centers or geographic regions in Figure 5. Below self-healing techniques and strategies commonly used in cloud environments:

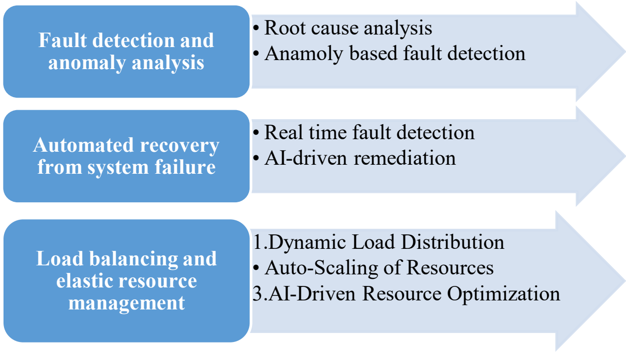

Fig. 5. Self-Healing Mechanisms in Cloud Environments

I. Fault Detection and Anomaly Analysis

Anomaly levels are used to identify occurrences of anomalies that are distinct from the norm. The identified instances of abnormality are then examined by the diagnostic function. The diagnostic function then determines the underlying causes of the anomalous occurrences to determine if corrective actions are necessary or not to reduce the possible dangers, and the remedial process is then activated if needed [35]. The main issue that hinders the smart grid network's ability to operate at its best nowadays is the frequency of system failures brought on by complex fault areas. These failures can have a major negative economic impact and occasionally have an adverse effect on people's quality of life or means of subsistence.

J. Automated Recovery from System Failure

The capacity of cloud database systems to automatically identify, diagnose, and fix errors or disturbances without human assistance is known as automated recovery from system failure. Real-time system health monitoring, anomalous behavior detection, and the implementation of pre-planned or AI-driven repair steps are all part of this process [36]. The system initiates recovery techniques, such as restarting instances, rerouting traffic, recovering data from backups, these systems may drastically save downtime and lessen the effect of failures on users and company operations by utilizing automation tools, orchestration frameworks, and intelligent rules.

K. Load Balancing and Elastic Resource Management

1) Dynamic Load Distribution

Intelligent load balancers are employed in modern cloud databases to spread query loads and traffic among several nodes in an equitable manner. This minimizes latency, eliminates potential bottlenecks, and ensures that no single element is a point of failure in enabling seamless self-healing and constant operation

2) Auto-Scaling of Resources

Elastic resource management supports automatic scaling of compute, memory and storage to meet the real-time demand. The system reacts rapidly when faced with spikes or failures to allocate the necessary resources to maintain the performance and availability of the service without human intervention.

3) AI-Driven Resource Optimization

With the use of AI and predictive analytics, cloud systems can predetermine work load variations and implement proactive changes in their resources. This reduces the risk of system overloads or downtime and at the same time enhances efficiency and encourages proactive healing initiatives.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Enabling Technologies and Tools in cloud database in self-healing

Self-healing of cloud databases is based on modern technologies which allows automatic fault detection and recovery. Predictive analytics and ML can aid in forecasting a system collapse whereas observability tools like Grafana and Prometheus monitor the state of a system in real-time. Scaling and automated recovery occur through container orchestration platforms, e.g. Kubernetes. Built-in resilience. Many cloud-native services (e.g. AWS RDS, Azure Monitor) are built with resilience via auto-scaling and Infrastructure-as-Code. All these tools assist in continuous availability and smart self-healing in a dynamic cloud system.

L. *Predictive Analytics and Machine Learning*

Self-healing features in cloud databases are made possible in large part by predictive analytics and machine learning (ML). ML models can identify even minor trends that expect problems in the future based on system logs, historical data, and real-time indicators. These models can help the system to make preventative recovery strategies by predicting potential outages or performance degradations in advance. The usage of predictive analytics ensures a faster response, increases reliability, and enables autonomous decision-making.

M. *Observability and Monitoring Framework*

The observability models provide a detailed view of the internal operation of the cloud. There are more than simple monitoring solutions providing real-time information about metrics, logs, and traces at any point in the stack. Tools like Prometheus, Grafana, and ELK (Elasticsearch, Logstash, Kibana) enable operators and automated systems to understand system states, pinpoint root causes of failures, and detect anomalies early. With the data provided by these frameworks, automatic diagnostics and recovery operations may be triggered with minimum latency, making them crucial for self-healing.

N. *Container Orchestration*

Docker Swarm and Kubernetes are examples of container orchestration systems that automate the deployment, scaling, and management of applications that are containerized. These platforms identify issues (such crashing containers or unresponsive services) and automatically restart, reschedule, or move them to healthy nodes in a self-healing environment. In order to guarantee zero-downtime deployments and steady system availability, it also make rolling upgrades and health checks possible. Cloud-native resilience relies heavily on orchestration, which promotes scalability, flexibility, and ongoing disruption recovery.

O. *Native Cloud Service*

Native databases to the cloud are designed to adapt their capacity on the fly to meet fluctuating demand [37]. Maintaining cost-effective operations requires efficient management of resource allocation to decrease expenses while still achieving performance SLAs [38]. It has managed resources in databases that are native to the cloud.

- **Netflix:** Netflix is a prime example of a company that heavily relies on cloud-native databases driven by AI for tasks like personalization, content recommendation, and other ML operations.
- **Uber:** The ride-sharing network is run by Uber using cloud native databases that are powered by AI. Every day, the network handles millions of requests and transactions.

Literature Review

This section presents previous research on AI-driven self-healing mechanisms in cloud database environments. Table II provides a structured comparison of earlier studies, emphasizing techniques the analysis highlights key strategies used to ensure outage resilience and uninterrupted business operations through intelligent self-healing cloud systems, along with their associated challenges and future directions:

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Saxena and Singh (2025) self-healing and fault-tolerant cloud-based digital twin processing management (SF-DTM) model. They employ collaborative DT tasks resource requirement estimation unit that utilizes newly devised federated learning with cosine similarity integration. Furthermore, SF-DTM incorporates a self-healing fault-tolerance strategy employing a frequent sequence fault-prone pattern analytics unit for deciding the most admissible virtual machine (VM) allocation. SF-DTM improved the services availability up to 13.2% over non-SF-DTM-based DT processing [39].

Giannopoulos et al. (2025) AI-powered self-healing solution for cloud-native system dynamic server activation. This suggested architecture integrates three significant frameworks: MOF for policy-based orchestration of network services, CCF for dynamic scaling of resources, and AIMLF for anomaly detection and predictive analytics. To maintain optimum energy efficiency and SLA compliance, it monitors workload fluctuations, forecasts resource consumption, and dynamically scales computer resources. Compared to conventional static resource allocation techniques, their solution dramatically improves power consumption, load balancing, and resource utilisation, according to numerical assessments that include analysis of real-world traffic data [40].

Barclay (2025) self-healing cloud infrastructure enabled by AI-powered observability and monitoring tools the mechanisms through which AI detects, diagnoses, and resolves cloud anomalies in real time, ensuring uninterrupted service availability and optimal performance. Cloud computing has become a foundational technology for enterprises, offering scalability, flexibility, and efficiency. However, as cloud environments grow in complexity, they become prone to failures, performance degradation, and security vulnerabilities. monitoring and incident response mechanisms are increasingly insufficient to handle the dynamic nature of cloud infrastructure, the key methodologies, frameworks, and technologies that enable AI-driven self-healing capabilities in cloud environments [41].

Luo et al. (2024) self-healing approach for distribution networks that use scattered power sources. Firstly, the outputs of photovoltaic power generation systems and wind power generation systems in distributed power distribution networks are analyzed maximizing the self-healing capacity of loads experiencing power loss, a self-healing model for distributed power distribution networks is constructed by introducing load weights, and multiple constraints of the model are set heuristic rules are employed to generate self-healing strategies for distribution networks with distributed power sources existing fault self-healing methods, their method achieves a fault self-healing coverage consistently exceeding 80%, while significantly reducing the recurrence rate of faults [42].

Padamati (2023) delves into more ideas around IaC with the goal of automating scalable clouds and facilitating self-healing via the use of AI technology. One of the most important aspects of modern cloud computing is infrastructure as code, which involves handling infrastructure code, preventing and forecasting problems, and reducing system idle time via self-remediation. Additionally, the most common problems that may arise during the AI IaC integration stage are discussed in this report along with potential fixes and suggestions for resolving them. Several recent developments and advancements in cloud infrastructure management have been made possible by the examination of the interaction between AI and IaC, which shows that this integration may improve cloud organizations' flexibility, stability, and productivity [43].

Zhang and Tan (2022) ML techniques for autonomous system tuning have shown superior configurations than seasoned database administrators (DBAs) for the current systems in production settings, particularly in the cloud. They first tune for a certain workload within a time frame and disregard the data and task's dynamic nature. which, in dynamic cloud settings, fine-tunes online databases securely. both scalar and adaptive database optimization, it uses context as a feature and applies contextual Bayesian Optimization with a split of the context space. based on real-world workloads and benchmarks that are constantly changing [44].

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

TABLE II. COMPARATIVE ANALYSIS OF AI-DRIVEN SELF-HEALING TECHNIQUES IN CLOUD DATABASES FOR OUTAGE RESILIENCE

Author(s)	Focus Area	Key Findings	Challenges	Limitations
Saxena & Singh (2025)	Self-healing & fault-tolerant cloud-based digital twin management (SF-DTM)	Proposed FL-based DT processing with 13.2% improvement in service availability	Integrating federated learning with real-time fault detection	May not generalize across all cloud architectures
Giannopoulos et al. (2025)	AI-based dynamic server activation in cloud-native systems	Enhanced SLA and energy efficiency using MOF, CCF & AIMLF frameworks	Forecasting resource demand in real-time	Requires high data quality and training
Barclay (2025)	AI-powered observability for anomaly detection	Real-time detection & resolution of cloud anomalies to enhance availability	Complexity in anomaly classification	Dependency on AI model performance
Luo et al. (2024)	Self-healing in distributed energy systems	Achieved >80% fault coverage using load-weighted self-healing model	Handling variability in renewable energy sources	Specific to distributed power networks
Padamati (2023)	Infrastructure as Code (IaC) with AI for self-remediation	Enhanced resilience via AI-driven predictive automation in IaC	Synchronizing IaC templates with dynamic configs	Lack of standardized AI integration patterns
Zhang & Tan (2022)	ML-based auto-tuning of cloud databases	Used contextual Bayesian Optimization to tune DBs in dynamic conditions	Adapting to workload fluctuations	Focused only on online tuning, no long-term performance data

Conclusion and Future Work

Self-healing cloud databases represent a transformative shift in how modern cloud environments approach resilience, operational continuity, and autonomous fault recovery. By integrating real-time observability, intelligent automation, and AI-driven diagnostics, these systems enable proactive monitoring, swift recovery, and adaptive response mechanisms that significantly reduce downtime and human dependency. As demonstrated throughout this study, the application of self-healing techniques ranging from fault detection, automated failover, and resource elasticity to AI-enabled decision-making has redefined service reliability across diverse cloud scenarios. Furthermore, enabling tools such as predictive analytics, container orchestration, and infrastructure-as-code (IaC) have made scalable, self-correcting infrastructures not only possible but increasingly necessary. However, despite significant progress, challenges persist in anomaly classification, data quality, and system adaptability across varying architectures and workloads. The limitations highlighted in recent literature indicate a pressing need for more standardized frameworks, improved context-aware learning models, and generalized fault-tolerant architectures. Future research should focus on developing self-learning systems that continuously refine their diagnostic and recovery capabilities with minimal supervision, as well as ensuring interoperability across hybrid and multi-cloud environments. Additionally, advancements in federated learning, context-sensitive resource optimization, and AI explainability are poised to enhance transparency and performance. Bridging the gap between automation and reliability through stronger SRE and DevOps integrations will also play a key role. To sum up, self-healing cloud databases are a commercial need rather than just a technological advancement, opening the door for really independent, robust, and intelligent cloud-native ecosystems that can meet the demands of always available digital services.

References

[1] Y. E. Gelogo and S. Lee, "Database Management System as a Cloud Service," *Int. J. Futur. Gener. Commun. Netw.*, vol. 5, no. 2, pp. 71–76, 2012.

[2] V. Shah, "Analyzing Traffic Behavior in IoT-Cloud Systems : A Review of Analytical Frameworks," pp. 877–885, 2023.

[3] D. Patel, "Leveraging Database Technologies for Efficient Data Modeling and Storage in Web Applications," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 10, no. 4, pp. 357–369, Jul. 2024, doi: 10.32628/CSEIT25113374.

[4] S. H. S. Ariffin, M. A. Baharuddin, M. H. M. Fauzi, N. M. A. Latiff, S. K. S.- Yusof, and N. A. A. Latiff, "Wireless water quality cloud monitoring system with self-healing algorithm," in *2017 IEEE 13th Malaysia International Conference on Communications (MICC)*, IEEE, Nov. 2017, pp. 218–223. doi: 10.1109/MICC.2017.8311762.

[5] L. Zhang, K. Pang, J. Xu, and B. Niu, "JSON-based control model for SQL and NoSQL data conversion in hybrid cloud database," *J. Cloud Comput.*, 2022, doi: 10.1186/s13677-022-00302-9.

[6] Gopikrishna Maddali, "Enhancing Database Architectures with Artificial Intelligence (AI)," *Int. J. Sci. Res. Sci. Technol.*, vol. 12, no. 3, pp. 296–308, May 2025, doi: 10.32628/IJSRST2512331.

[7] V. Shah, "Securing the Cloud of Things : A Comprehensive Analytics of Architecture , Use Cases , and Privacy Risks," vol. 3, no. 4, pp. 158–165, 2023, doi: 10.56472/25832646/JETA-V3I8P118.

[8] S. S. Gill, I. Chana, M. Singh, and R. Buyya, "RADAR: Self-configuring and self-healing in resource management for enhancing quality of cloud services," *Concurr. Comput. Pract. Exp.*, vol. 31, no. 1, Jan. 2019, doi: 10.1002/cpe.4834.

[9] A. Arulappan, A. Mahanti, K. Passi, T. Srinivasan, R. Naha, and G. Raja, "DQN Approach for Adaptive Self-Healing of VNFs in Cloud-Native Network," *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3365635.

[10] G. Li, H. Dong, and C. Zhang, "Cloud databases," *Proc. VLDB Endow.*, 2022, doi: 10.14778/3554821.3554893.

[11] S. Petrenko, "Self-Healing Cloud Computing," *Vopr. kiberbezopasnosti*, vol. 1, no. 41, pp. 80–89, 2021, doi: 10.21681/2311-3456-2021-1-80-89.

[12] R. Patel, "Advancements in Renewable Energy Utilization for Sustainable Cloud Data Centers : A Survey of Emerging Approaches," *Int. J. Curr. Eng. Technol.*, vol. 13, no. 5, pp. 447–454, 2023.

[13] D. Patel and R. Tandon, "Cryptographic Trust Models and Zero-Knowledge Proofs for Secure Cloud Access Control and Authentication," *Int. J. Adv. Res. Sci. Commun. Technol.*, pp. 749–758, Dec. 2022, doi: 10.48175/IJARSCT-7744D.

[14] S. Dziubak, "Review of Cloud Database Benefits and Challenges," *Mod. Manag. Rev.*, vol. 28, no. 3, pp. 7–16, Sep. 2023, doi: 10.7862/rz.2023.mmr.14.

[15] H. J. Bhatti and B. B. Rad, "Databases in Cloud Computing: A Literature Review," *Int. J. Inf. Technol. Comput. Sci.*, vol. 9, no. 4, pp. 9–17, 2017, doi: 10.5815/ijitcs.2017.04.02.

[16] Y. Mansouri, V. Prokhorenko, F. Ullah, and M. A. Babar, “Resource Utilization of Distributed Databases in Edge-Cloud Environment,” *IEEE Internet Things J.*, 2023, doi: 10.1109/JIOT.2023.3235360.

[17] M. Amran *et al.*, “Self-Healing Concrete as a Prospective Construction Material: A Review,” *Materials*. 2022. doi: 10.3390/ma15093214.

[18] V. Prajapati, “Cloud-Based Database Management: Architecture, Security, challenges and solutions,” *J. Glob. Res. Electron. Commun.*, vol. 01, no. 1, pp. 07–13, 2025.

[19] H. W. Dong, C. Zhang, G. L. Li, and J. H. Feng, “Survey on Cloud-native Databases,” *Ruan Jian Xue Bao/Journal of Software*. 2024. doi: 10.13328/j.cnki.jos.006952.

[20] X. Feng, C. Guo, T. Jiao, and J. Song, “A maturity model for AI-empowered cloud-native databases: from the perspective of resource management,” *J. Cloud Comput.*, 2022, doi: 10.1186/s13677-022-00318-1.

[21] M. Menghnani, “Modern Full Stack Development Practices for Scalable and Maintainable Cloud-Native Applications,” *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 2, 2025, doi: 10.5281/zenodo.14959407.

[22] G. C. Deka, “A Survey of Cloud Database Systems,” *IT Prof.*, vol. 16, no. 2, pp. 50–57, Mar. 2014, doi: 10.1109/MITP.2013.1.

[23] N. K. Prajapati, “Cloud-based serverless architectures : Trends , challenges and opportunities for modern applications,” vol. 16, no. 01, pp. 427–435, 2025.

[24] D. Patel, “The Role of Amazon Web Services in Modern Cloud Architecture: Key Strategies for Scalable Deployment and Integration,” *Asian J. Comput. Sci. Eng.*, vol. 9, no. 4, 2024, doi: 10.22377/ajcse.v9i04.215.

[25] H. B. Patel and N. Kansara, “Cloud Computing Deployment Models: A Comparative Study,” *Int. J. Innov. Res. Comput. Sci. Technol.*, 2021, doi: 10.21276/ijircst.2021.9.2.8.

[26] S. Garg, “AI/ML Driven Proactive Performance Monitoring, Resource Allocation and Effective Cost Management in SAAS Operations,” *Int. J. Core Eng. Manag.*, vol. 6, no. 6, pp. 263–273, 2019.

[27] R. Ara, A. Rahim, S. Roy, and U. K. Prodhan, “Cloud Computing : Architecture , Services , Deployment Models , Storage , International Journal of Trend in Scientific Research and Development (IJTSRD) Cloud Computing : Architecture , Services , Deployment Models , Storage , Benefits and Challenges,” *Int. J. Trend Sci. Res. Dev.*, vol. 4, no. 4, pp. 837–842, 2020.

[28] V. Shah, “Managing Security and Privacy in Cloud Frameworks: A Risk with Compliance Perspective for Enterprises,” *Int. J. Curr. Eng. Technol.*, vol. 12, no. 06, pp. 1–13, 2022, doi: https://doi.org/10.14741/ijcet/v.12.6.16.

[29] W. Al Shehri, “Cloud Database Database as a Service,” *Int. J. Database Manag. Syst.*, vol. 5, no. 2, pp. 1–12, Apr. 2013, doi: 10.5121/ijdms.2013.5201.

[30] T. H. Shareef, K. H. Shareef, and B. N. Rashid, “A Survey of Comparing Different Cloud Database Performance: SQL and NoSQL,” *Passer J. Basic Appl. Sci.*, vol. 4, no. 1, pp. 45–57, 2022, doi: 10.24271/psr.2022.301247.1104.

[31] C. Li and J. Gu, “An integration approach of hybrid databases based on SQL in cloud computing environment,” *Softw. - Pract. Exp.*, 2019, doi: 10.1002/spe.2666.

[32] A. Rashid and A. Chaturvedi, “Cloud Computing Characteristics and Services A Brief Review,” *Int. J. Comput. Sci. Eng.*, vol. 7, no. 2, pp. 421–426, 2019, doi: 10.26438/ijcse/v7i2.421426.

[33] E. Ok, G. John, and P. Chris, “Autonomous Infrastructure & Self-Healing Clouds,” 2024.

[34] T. Adewale, “Building a self-healing cloud system through predictive insights,” 2024.

[35] O. Johnphill *et al.*, "Self-Healing in Cyber–Physical Systems Using Machine Learning: A Critical Analysis of Theories and Tools," *Futur. Internet*, vol. 15, no. 7, 2023, doi: 10.3390/fi15070244.

[36] P. Robertson and B. Williams, "Automatic recovery from software failure," *Commun. ACM*, vol. 49, no. 3, pp. 41–47, Mar. 2006, doi: 10.1145/1118178.1118200.

[37] S. Kabade and A. Sharma, "Securing Pension Systems with AI-Driven Risk Analytics and Cloud-Native Machine Learning Architectures," vol. 12, no. 4, pp. 5405–5416, 2024.

[38] S. Kumar, S. Singh, and H. Nerella, "Resource Management in AI-Enabled Cloud Native Databases: A Systematic Literature Review Study," *Orig. Res. Pap. Int. J. Intell. Syst. Appl. Eng. IJISAE*, vol. 12, no. 21, pp. 3621–3630, 2024.

[39] D. Saxena and A. K. Singh, "A Self-Healing and Fault-Tolerant Cloud-Based Digital Twin Processing Management Model," *IEEE Trans. Ind. Informatics*, vol. 21, no. 5, pp. 4233–4242, 2025, doi: 10.1109/TII.2025.3540498.

[40] A. Giannopoulos, S. Spantideas, P. Trakadas, J. Perez-Valero, G. Garcia-Aviles, and A. S. Gomez, "AI-Driven Self-Healing in Cloud-Native 6G Networks Through Dynamic Server Scaling," in *2025 IEEE 11th International Conference on Network Softwarization (NetSoft)*, 2025, pp. 43–48. doi: 10.1109/NetSoft64993.2025.11080631.

[41] T. Barclay, "Self-Healing Cloud Infrastructure Enabled by AI-Powered Observability and Monitoring Tools," 2022.

[42] L. Luo, M. Chen, R. Su, and W. Long, "Heuristic rule-based self-healing method for faults in distribution networks with distributed power sources," in *2024 4th International Conference on New Energy and Power Engineering (ICNEPE)*, 2024, pp. 828–832. doi: 10.1109/ICNEPE64067.2024.10860408.

[43] J. R. Padamati, "AI-Driven Self-Healing Infrastructure: The Next Frontier in Scalable Cloud Deployments," *Int. J. Adv. Res. Sci. Technol.*, vol. 13, pp. 506–517, 2023.

[44] X. Zhang, H. Wu, Y. Li, J. Tan, F. Li, and B. Cui, "Towards Dynamic and Safe Configuration Tuning for Cloud Databases," in *Proceedings of the 2022 International Conference on Management of Data*, New York, NY, USA: ACM, Jun. 2022, pp. 631–645. doi: 10.1145/3514221.3526176.