Efficient and robust disaster recovery system using cloud-based algorithms with data integrity

Gurumoorthi Gurulakshmanan[1], Raveendra Nandhavanam Amarnath[2]
[1]Project Manager, Mphasis, Texas, USA
[2]Project Lead, Mphasis, Campbell Dr. Melissa, Texas, USA

Article Info

Article history:

Received Oct 31, 2023
Revised Mar 7, 2024
Accepted Mar 20, 2024

Keywords:

Cloud-based algorithms
Data replication
Disaster recovery
Failover algorithms
Virtual machine migration
Consistency algorithms

ABSTRACT

Incorporating cloud-based algorithms for disaster recovery (DR), it explores data replication, failover, virtual machine (VM) migration, and consistency algorithms. These algorithms play a pivotal role in safeguarding data and system continuity during unforeseen disruptions. Data replication ensures redundancy, failover algorithms swiftly transition to backup resources, VM migration facilitates resource optimization, and consistency algorithms maintain data integrity. Leveraging cloud technology enhances the effectiveness of these algorithms, providing robust DR solutions critical for business continuity in today's digital landscape. The recent growth in popularity of internet services on a massive scale has also raised the demand for stable underpinnings. Despite the fact that DR for big data is frequently overlooked in security research, the majority of existing approaches use a narrow, endpoint-centric approach. The significance of DR strategies has grown as cloud storage has become the norm for more data. But traditional cloud-centric DR techniques may be expensive, thus less expensive alternatives are being sought. There is persistent concern in the information technology (IT) community about whether or not cloud service providers (CPs) can guarantee data and service continuity in the event of a disaster.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Gurumoorthi Gurulakshmanan
Project Manager, Mphasis
Texas, USA
Email: gurumoorthig198@gmail.com

1. INTRODUCTION

Disaster recovery (DR) systems use a variety of methods, such as regularly scheduled backups, continuous data synchronization, and the creation of a new, parallel system at an offsite location [1]. The distance between the main and secondary replica location should increase proportionally with the "larger" the possible effect of a catastrophe. However, performance may suffer under the weight of the ensuing long-distance delay [2]. Data should be backed up and accessible in case of cloud failure or loss. If the cloud were to become corrupted or destroyed, all of the data stored there would be lost [3]. The storage and security of data in the cloud has evolved greatly since its debut to the business sector. There are various security and privacy hazards and availability difficulties, especially for companies, on the internet since it is an open network for exchanging information and performing transactions [4]. Big data has several dimensions beyond just its sheer size that need exploring and balancing, including velocity, diversity, validity, and value. Data processing velocity refers to how quickly enormous datasets can be processed [5].

Due to the rapid rise of information technology, data security challenges are growing. The frequency and complexity of natural and man-made disasters have increased, making DR increasingly crucial. Security strategy's DR protects enterprises from unexpected calamities [6]. It need specialized data protection methods and well-thought-out preparation to endure calamities. Tolerating calamities explicitly calls for separating main and backup infrastructures physically so that the same calamity does not take down both [7]. The Internet is a dangerous place to send data, despite its cheap cost. Unencrypted Internet data is useful, but vital data should be encrypted [8]. Protecting the integrity and privacy of cloud-based data services is crucial for cloud computing since cloud storage providers may be untrustworthy and the data they store is sensitive. Most modern firms use cloud computing to save money on infrastructure and take advantage of IT [9]. Recent advancements in cloud computing provide a low-cost, low-overhead replacement for conventional DR Plan (DRP)s, making them accessible even to small and medium-sized enterprises [10].

A DR service may also offer business continuity (BC) by reducing downtime and data loss in the event of a catastrophe, although doing so usually comes at an additional expense [11]. This includes determining the causes of the disaster, determining the steps that must be taken in the aftermath, determining the entities (real or virtual) that will be involved, and establishing the priorities that will be applied to these tasks [12]. There are still managerial issues to address, despite the widespread use of technology. Effective catastrophe management is the most pressing issue in the future of cloud computing [13]. Keeping IT assets and processes available, functioning, and resilient to meet the organization's overall business continuity goals is the primary emphasis of IT business continuity [14]. It reduces user-perceived latency by using advanced data storage, processing, and dynamic resource allocation for real-time calculations [15].

Firm continuity, including catastrophe recovery, focuses on restoring full operations following a tragedy. IT systems are critical to contemporary enterprises, thus DR is essential to company continuity [16]. This procedure does not inspire confidence in security, unfortunately, for a number of reasons. Internet-based identity verification based on the "trust but verify" principle is used. A domain authentication certificate may be obtained by an attacker who has the opportunity to temporarily impersonate a domain [17].

As technology improved, better low-cost options became a need. At this point, the concept of "Cloud computing" began to arise. Cloud computing is a paradigm change in the way data is stored and shared over a network so that services may be delivered on demand [18]. The service level agreement is the sole binding contract between the service provider and the customer. The service description must be factually correct and include exhaustive technical details [19]. Performance may be boosted with the use of cloud services like Amazon elastic compute cloud and Google Cloud platform (GCP). For enormous information flow between government and other organizations, cloud computing may be optimal [20].

Rather of dismissing natural catastrophes as random acts of nature, improved governance and sustainable development should be implemented to safeguard economies and populations. Urban floods, gas explosions, fire hazards, improper waste management, and other illnesses are all examples of manmade catastrophes that plague developing nations [21]. Information technology and robots improve staff capacities and speed digital data collecting tactics for post-pandemic disaster management decision-makers. These events have influenced this region's countries' social, economic, and well-being [22]. Community members and external stakeholders create, apply, and evaluate recovery simulation modeling to anticipate realistic DR trajectories for future hazard occurrences. Participatory modeling examines community resilience assessment frameworks for recovery [23]. Central management systems and communication base stations (BSs) may be damaged or destroyed during a natural catastrophe, producing a supply shortage and communication breakdown. Restarting telecom and internet services may take days or weeks, depending on damage [24]. In the event that a site where a possible replica is kept goes down, there are backups accessible since it permits the generation of multiple copies of data that may be stored at various locations. Users may access their data with no regard for network traffic or reliance on the underlying network infrastructure thanks to replication [25].

2. PROPOSED SYSTEM

Organizational resilience may be strengthened by implementing DR utilizing cloud-based algorithms. Using the power of cloud-based algorithms, DR plans may be made more efficient and successful than ever before. These algorithms automate the evaluation of crisis scenarios, allowing for immediate action and the recovery of lost data. This game-changing method protects vital operations and data integrity in the event of any kind of calamity, whether it be a natural disaster, a cyberattack, or a hardware breakdown. Using cloud-based algorithms for DR strategies goes above and beyond more conventional approaches, giving businesses the ability to proactively protect digital assets and limit downtime, which in turn protects their operations and reputation in an ever-changing and unpredictable environment.

Implementing DR using cloud-based algorithms has as its primary goal and purpose the improvement of resilience and continuity in the face of unforeseen interruptions. The ultimate goal is to implement algorithms in the cloud that will automate and improve the DR procedure. The time it takes to restore mission-critical systems and data will be cut down significantly. Furthermore, it aspires to provide businesses with the flexibility to quickly adjust in the face of a wide range of crisis scenarios, such as natural disasters, cyberattacks, and infrastructure breakdowns. In order to ensure that essential systems and services continue operating normally during and after disruptive events, it is necessary to strengthen organizational preparation, protect key data assets, keep business operations running, and protect the organization's reputation.

2.1. Data replication algorithms for DR using cloud: strategies and implementation

Data replication algorithms are crucial to contemporary DR, especially when used with cloud technology. This method provides data availability and business continuity during unexpected outages. Duplicating data across numerous sites helps reduce disaster damage to companies. Cloud-based DR solutions are unmatched. Data availability is improved by the cloud's broad architecture and geographically distributed data centers [26]. Various data replication techniques do this. Synchronous replication requires real-time data mirroring between main and secondary nodes. Despite its consistency, this strategy may cause delay by requiring secondary site validation. Alternatively, asynchronous replication allows data mirroring delays to reduce latency. When network costs are an issue, snapshot-based replication is efficient. This method transfers data snapshots to secondary locations at predefined intervals. Cost-effective, it may lag behind instant modifications. In (1) shows the replication efficiency Equation where $Efficiency$ represents the replication efficiency as a ratio.

$$Efficiency = \frac{D_{replicated}}{D_{total}} \times \frac{T_{replication}}{T_{total}} \tag{1}$$

$D_{replicated}$ is the amount of data successfully replicated (in bytes). D_{total} is the total amount of data to be replicated (in bytes). $T_{replication}$ is the time spent on data replication (in seconds). D_{total} is the total time available for replication (in seconds).

2.2. For DR using cloud using the failover algorithms: strategies and implementation

Modern DR solutions rely on failover algorithms, especially when used with cloud technology. This method provides company continuity during unexpected interruptions. Organizations may limit catastrophe damage by smoothly transferring jobs and operations to alternative platforms. Cloud-based DR solutions are unmatched. The cloud's vast architecture allows geographically distributed data centers to improve operational dependability. This crucial job is performed by various failover techniques. The Priority-based Failover method assigns specified priority to systems, providing a smooth transition from main to secondary arrangements. In (2) shows the service downtime.

$$Downtime = T_{failover} + T_{sync} \tag{2}$$

Where, $Downtime$ represents the total service $Downtime$ during the failover process (in seconds). $T_{failover}$ is the time taken for the failover process itself (in seconds). T_{sync} is the time required for data synchronization between primary and secondary systems (in seconds). To minimize downtime, organizations often focus on optimizing the failover process (reducing $T_{failover}$) and minimizing the time needed for data synchronization (reducing T_{sync}). Techniques such as continuous data replication, real-time synchronization, and efficient failover mechanisms are employed to reduce these components of downtime. Figure 1 show 2015–2020 data replication and failover algorithm usage, important components of cloud-based DR systems. Failover algorithms quickly switch to backup resources during disturbances, while data replication provides data redundancy. data replication and failover algorithms are rising in popularity as the cloud age emphasizes effective DR solutions.

Figure 2 explains data backup model and it shows SA_1, the data catastrophe recovery service provider. Users individuals, corporations, and cloud service providers are all SA_1 customers. Their SA_1 accounts and privileges are valid. SA_1 uses cloud resources from SA_2-CP_J and others. Data DR clients and many cloud service providers make up DR-Cloud.

The least-common-denominator cloud storage interface of each SA receives/sends user data. SA_1 request buffer holds data backup requests for one time period. Replica scheduler reads request buffer, makes three replicas, and sends to SAs. Resource manager tracks SA resource consumption changes. Metadata contains SA resource utilization and replica locations. Figure 3 shows cloud service providers CP_X, CP_Y, and CP_Z with data recovery request replicas. The recovery manager of CP_1 receives and evaluates the recovery request, then chooses a CP with at least one duplicate. Recovery proxy is a customer-side agent that restores CP data.

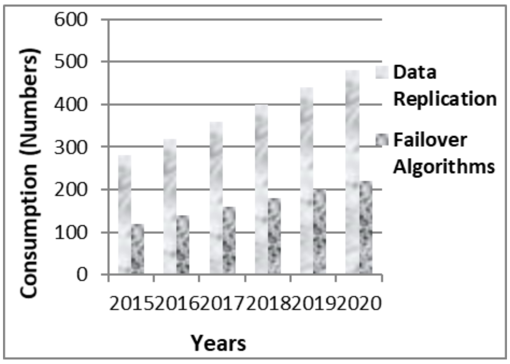

Figure 1. Year wise consumption (2015-2020) of data replication and failover algorithms for disaster recover

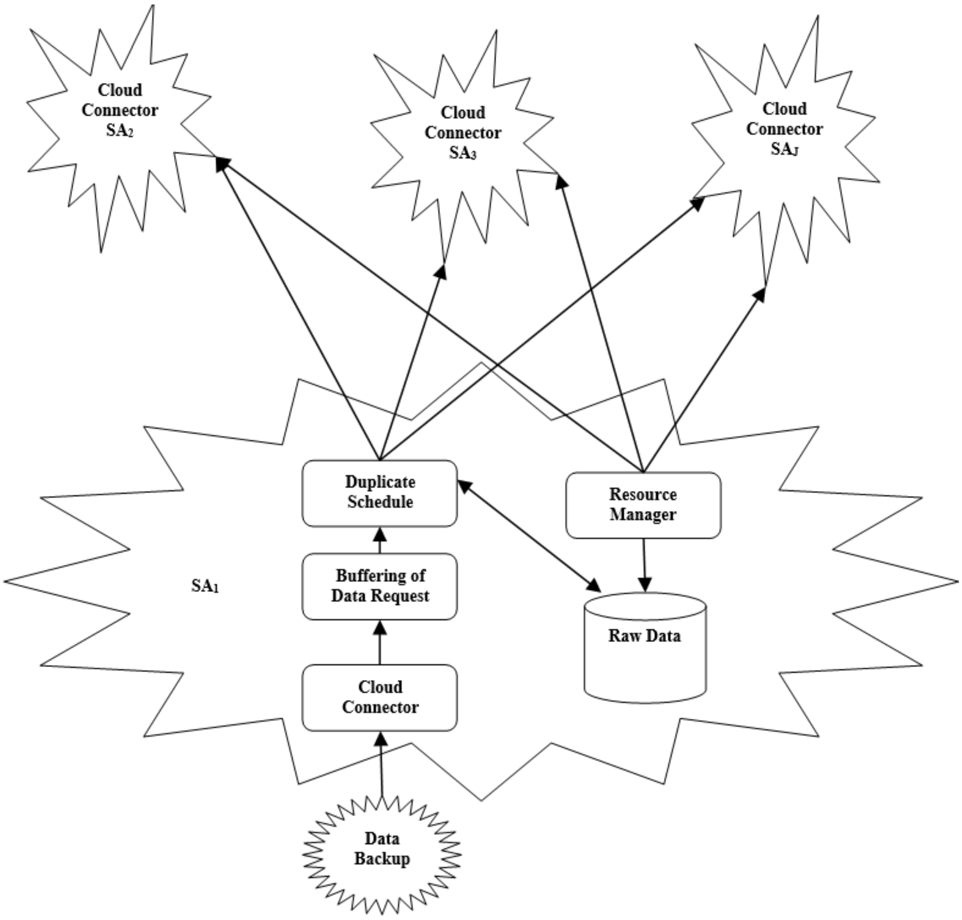

Figure 2. Data backup model

Table 1 details many approaches of use cloud-based DR. Different strategies are aimed at catering to concerns about data loss, recovery time, and budget. The aforementioned methods range from little preparation (data backup) to maximum preparation (complete cloud-based replica). The recovery time objective (RTO) and recovery point objective (RPO) targets of a business will determine the solution they go with.

Table 2 compares cloud-based DR data replication and failover algorithms. Data replication uses synchronous, asynchronous, or snapshot-based techniques to transfer data from source to destination systems. Failover algorithms minimize downtime by smoothly switching from a main to a backup system after failure. The methods used affect data integrity, recovery time, and system availability. Understanding these factors helps create cloud DR plans.

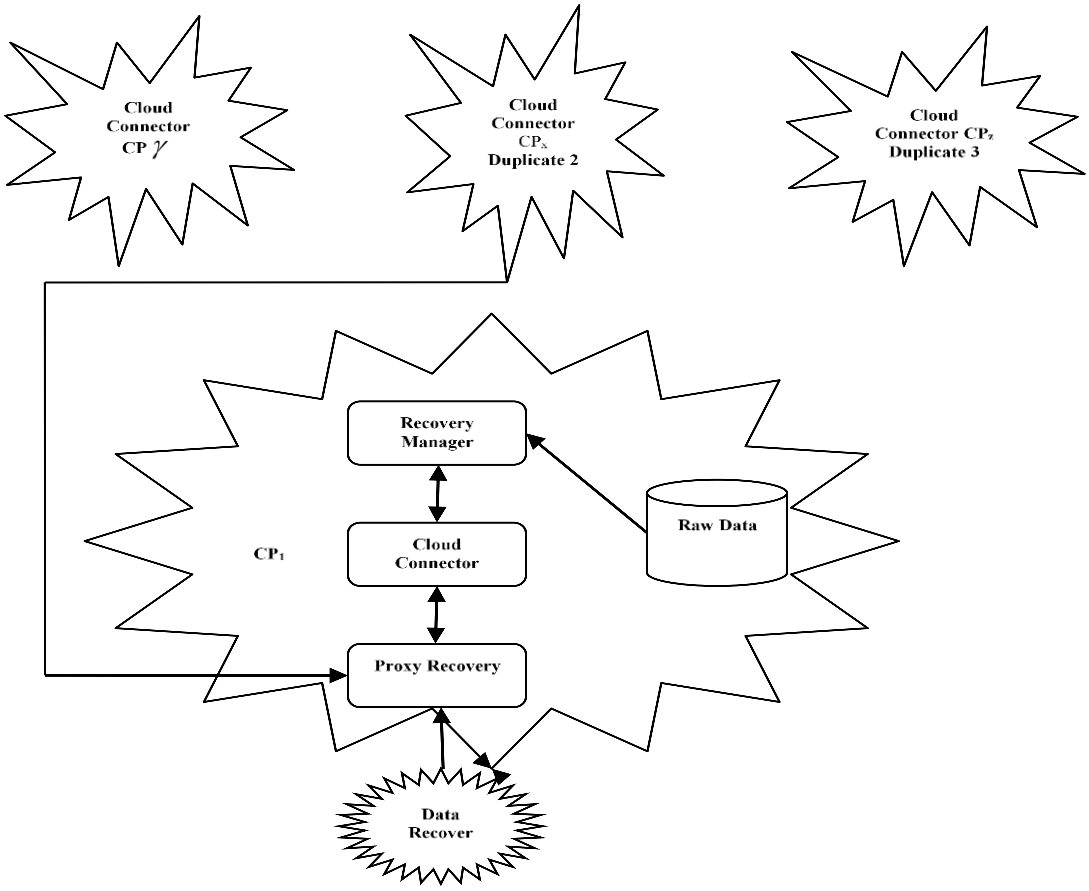
Cloud Connector CP γ
Cloud Connector CP_x Duplicate 2
Cloud Connector CP_z Duplicate 3
Recovery Manager
Cloud Connector
Proxy Recovery
Raw Data
CP_1
Data Recover

Figure 3. Data recovery model

Table 1. DR strategies using cloud resources: description, implementation, and strategies

Description	Implementation	Strategies
Backup and restore	Take frequent cloud backups and restore them after a disaster.	Scheduled backups, automatic restores.
Pilot light	Keeping a minimum cloud infrastructure ready to scale up in a crisis.	Pre-configured virtual servers, scaling scripts.
Warm standby	Partially active clouds recover quicker than cold standbys.	Automatic data syncing and failover.
Hot standby	Running a completely working cloud copy of production.	Continuous data reproduction, load balancers for failover.
Data replication	Synchronizing on-premises and cloud data for rapid recovery.	Log shipping, database replication, data mirroring.

Table 2. Comparison of data replication and failover algorithms in cloud-based DR: insights into implementation and strategies

Aspects	Data replication algorithms	Failover algorithms
Definition	Data replication helps catastrophe recovery by transmitting data.	Main and backup systems are smoothly switched during failover to minimize service disruption.
Types	1. Synchronous replication: waits for target confirmation before continuing to ensure real-time data consistency. 2. Asynchronous replication: copies data slowly, which may cause inconsistencies. 3. Data snapshot-based replication: takes snapshots periodically.	1. Active-passive failover: standby systems are idle until primary fail. 2. Active-active failover: primary and backup systems share traffic burden. 3. Shared nothing failover: when the main fails, a secondary system takes over, frequently remotely.
Latency	Latency: sync, confirmation; async, lag; snapshot, interval-dependent.	Backup algorithms activate failover systems to reduce service interruption.
Consistency	Sync: consistency, delays; async: divergence, lag; snapshot: discrepancies, timing.	Risk data algorithm flaws. Backup activates sync. Consistency globally. Unaltered consistency.
Implementation	Network and data determine replication. Moments for balance, sync for urgency, async for tolerance.	Active-passive assures data consistency; active-active needs synchronization and balancing; shared nothing requires comprehensive synchronization.

3. RESULTS AND DISCUSSION

3.1. VM migration algorithms for DR using cloud: strategies and implementation

In the event of a catastrophe, the ability to quickly and easily migrate VMs to a different cloud server is essential for business continuity. These algorithms have been carefully constructed to allow for the swift and automatic migration of VMs from a compromised or underperforming cloud environment to a stable and functional one. Figure 4 explains a single-cloud architecture. VM migration algorithms analyze resource consumption, network circumstances, and performance data in real time to make educated judgments that maximize resource allocation and reduce downtime. Within the context of DR, they allow for rapid responses to catastrophic events, keeping mission-critical applications and services available and secure. Unsung heroes of cloud-based DR, VM migration algorithms allow for the speedy restoration of IT services, protect data integrity, and strengthen business continuity across the board, giving businesses the leeway they need to lessen the blow of disasters and keep services running smoothly. VM migration Algorithms are essential to modern DR, especially when used with cloud technology.

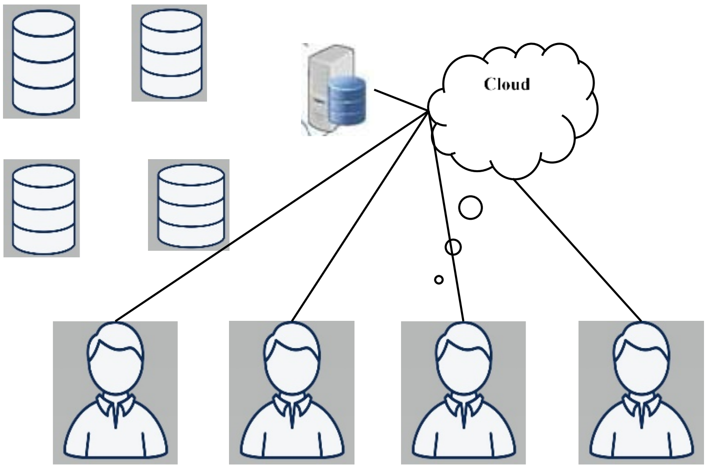

Figure 4. Single-cloud architecture

3.2. Usage of cloud consistency algorithms in DR: strategies and implementation

Consistency algorithms are crucial to modern DR tactics, especially when used with cloud technology. This method assures data integrity and operational continuity during unexpected outages. Organized data updates across remote systems may reduce disaster damage to critical activities. Cloud-based DR solutions are unmatched. The cloud's vast architecture allows globally scattered data centers, improving data availability and dependability. A variety of consistency methods support this vital duty. A strict consistency approach requires all nodes to access the same data version at all times. This method maintains homogeneity but may cause delay and worse performance owing to continual synchronization. The eventual consistency model enables distributed systems to temporarily diverge data, ensuring that all copies converge. This method emphasizes availability and response above exact conformity, making it ideal for urgent situations. In conflict-reduction situations, the causal consistency method works. If one action causally precedes another, all nodes perceive them in the same sequence.

Figure 5 shows the 2015–2020 use of VM migration and consistency algorithms, essential components of cloud-based DR systems. VM migration algorithms optimize resources and transitions, while consistency algorithms protect data during interruptions. The data shows that these algorithms are becoming more important in cloud-based disaster recovery, where resource management and data integrity are crucial.

When it comes to DR in the cloud [27], consistency algorithms are crucial for maintaining data integrity and dependability despite interruptions. These techniques are meant to keep dispersed data systems consistent in the face of network partitions, failed hardware, or other unforeseen events [28]. They ensure that DR procedures go smoothly, with little data loss or inconsistencies, by managing data synchronization and replication processes. When using cloud infrastructure, this is extremely important since it helps businesses keep running, keeps data accurate, and speeds DR. DR plans rely heavily on consistency algorithms because they increase the robustness of cloud-based systems and protect vital data assets, allowing businesses to weather interruptions with confidence and minimum downtime [29]. Multi-cloud architectures shown in Figure 6, involve a customer placing an order with many service providers [30].

When using a multi-cloud architecture, workloads and data are strategically distributed among different cloud providers to increase redundancy, reduce the risk of vendor lock-in, and maximize performance. However, in order to avoid fragmentation and inefficiencies, it needs expert management and close tracking of costs. The task of overseeing a system that spans several clouds is a difficult one. In order to successfully coordinate resources across different cloud platforms while maintaining security and compliance, advanced technologies and procedures are required. Fragmentation, higher operating costs, and cost overruns are possible if this is not done.

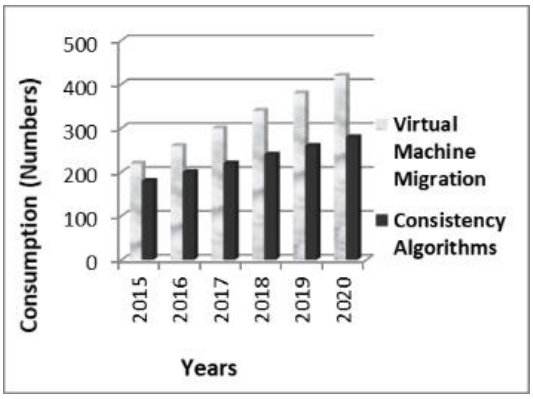

Figure 5. Year wise consumption (2015-2020) of VM migration and consistency algorithms for DR

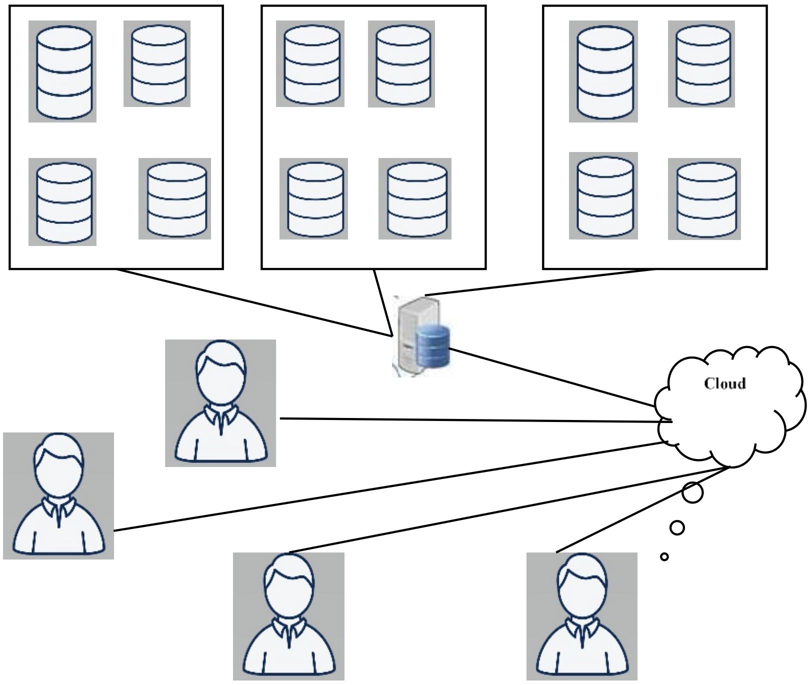

Figure 6. Multi cloud architecture

4. CONCLUSION

By allowing users to pool resources from several cloud providers, DR Cloud greatly simplifies the DR process. Through a uniform interface, customers communicate with a single service provider, hiding the underlying heterogeneity of the cloud from view. Although cloud platforms are often sought after as backup sites owing to their low costs, significant latency might be a performance bottleneck when using conventional replication techniques. It examines at the current status of DR in cloud computing, discussing the many forms of DR and why a multifaceted strategy is required for effective data restoration in private clouds and big data services. It highlights cloud-based DR solutions for enhanced dependability and scalability and gives illustrated DR scenarios that show the need of preparation. The effectiveness of distributed encryption systems is contingent on striking a balance between security and performance. Size, algorithm adequacy, and replication probability are crucial. It examines past research and presents a roadmap for assessing approaches to data recovery problems in both single-cloud and multi-cloud environments. Costs should be estimated using accessible calculators or price guides once application requirements, computing needs, and catastrophe scenarios have been considered.

REFERENCES

[1] Y. Gu, D. Wang, and C. Liu, "DR-cloud: multi-cloud based disaster recovery service," *Tsinghua Science and Technology*, vol. 19, no. 1, pp. 13–23, Feb. 2014, doi: 10.1109/TST.2014.6733204.

[2] T. Wood, H. A. Lagar-Cavilla, K. K. Ramakrishnan, P. Shenoy, and J. Van Der Merwe, "PipeCloud: using causality to overcome speed-of-light delays in cloud-based disaster recovery," *Proceedings of the 2nd ACM Symposium on Cloud Computing, SOCC 2011*, 2011, doi: 10.1145/2038916.2038933.

[3] R. Damaševičius, N. Bacanin, and S. Misra, "From sensors to safety: internet of emergency services (IoES) for emergency response and disaster management," *Journal of Sensor and Actuator Networks*, vol. 12, no. 3, p. 41, May 2023, doi: 10.3390/jsan12030041.

[4] A. Z. Abualkishik, A. A. Alwan, and Y. Gulzar, "Disaster recovery in cloud computing systems: an overview," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, pp. 702–710, 2020, doi: 10.14569/IJACSA.2020.0110984.

[5] V. Chang, "Towards a big data system disaster recovery in a private cloud," *Ad Hoc Networks*, vol. 35, pp. 65–82, 2015, doi: 10.1016/j.adhoc.2015.07.012.

[6] S. Hamadah, "Cloud-based disaster recovery and planning models: an overview," *ICIC Express Letters*, vol. 13, no. 7, pp. 593–599, 2019, doi: 10.24507/icicel.13.07.593.

[7] J. Alcantara, T. Oliveira, and A. Bessani, "GINJA: one-dollar cloud-based disaster recovery for databases," *Middleware 2017 - Proceedings of the 2017 International Middleware Conference*, pp. 248–260, 2017, doi: 10.1145/3135974.3135985.

[8] E. Rice, P. Safonov, and D. Guster, "Distributed key systems: enhancing security, fault tolerance and disaster recovery in cloud computing," *Issues in Information Systems*, vol. 14, no. 2, pp. 444–451, 2013, doi: 10.48009/2_iis_2013_444-451.

[9] M. M. Alshammari, A. A. Alwan, A. Nordin, and I. F. Al-Shaikhli, "Disaster recovery in single-cloud and multi-cloud environments: issues and challenges," in *2017 4th IEEE International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, IEEE, Nov. 2017, pp. 1–7. doi: 10.1109/ICETAS.2017.8277868.

[10] O. H. Alhazmi and Y. K. Malaiya, "Evaluating disaster recovery plans using the cloud," in *2013 Proceedings Annual Reliability and Maintainability Symposium (RAMS)*, IEEE, Jan. 2013, pp. 1–6. doi: 10.1109/RAMS.2013.6517700.

[11] D. M. Kesa, "Ensuring resilience: integrating IT disaster recovery planning and business continuity for sustainable information technology operations," *World Journal of Advanced Research and Reviews*, vol. 18, no. 3, pp. 970–992, 2023, doi: 10.30574/wjarr.2023.18.3.1166.

[12] L. Tomas *et al.*, "Disaster recovery layer for distributed openstack deployments," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 112–123, Jan. 2020, doi: 10.1109/TCC.2017.2745560.

[13] R. Javed, S. Anwar, K. Bibi, M. U. Ashraf, and S. Siddique, "Prediction and monitoring agents using weblogs for improved disaster recovery in cloud," *International Journal of Information Technology and Computer Science*, vol. 11, no. 4, pp. 9–17, Apr. 2019, doi: 10.5815/ijitcs.2019.04.02.

[14] B. Meenakshi, A. Vanathi, B. Gopi, S. Sangeetha, L. Ramalingam, and S. Murugan, "Wireless sensor networks for disaster management and emergency response using SVM classifier," in *2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, IEEE, Aug. 2023, pp. 647–651. doi: 10.1109/SmartTechCon57526.2023.10391435.

[15] K. Grolinger, E. Mezghani, M. A. M. Capretz, and E. Exposito, "Collaborative knowledge as a service applied to the disaster management domain," *International Journal of Cloud Computing*, vol. 4, no. 1, p. 5, 2015, doi: 10.1504/IJCC.2015.067706.

[16] N. A. Gloria, I. Chinagulm, S. O. Anigbogu, and K. Usman, "Development of an enhanced cloud deployment model for resilient internet disaster recovery and management," *International Journal of Innovations in Engineering Research and Technology*, vol. 8, no. 6, pp. 151–162, 2021.

[17] M. G. Aruna, M. K. Hasan, S. Islam, K. G. Mohan, P. Sharan, and R. Hassan, "Cloud to cloud data migration using self sovereign identity for 5G and beyond," *Cluster Computing*, vol. 25, no. 4, pp. 2317–2331, Aug. 2022, doi: 10.1007/s10586-021-03461-7.

[18] D. Bajpai and R. K. Thulasiram, "Comparing replication strategies for financial data on openstack based private cloud," *CLOUD COMPUTING 2016 : The Seventh International Conference on Cloud Computing, GRIDs, and Virtualization*, no. March, pp. 139–144, 2016.

[19] B. R. Kandukuri, R. P. V., and A. Rakshit, "Cloud security issues," in *2009 IEEE International Conference on Services Computing*, IEEE, 2009, pp. 517–520. doi: 10.1109/SCC.2009.84.

[20] M. Habiba and S. Akhter, "A cloud based natural disaster management system," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7861 LNCS, 2013, pp. 152–161. doi: 10.1007/978-3-642-38027-3_16.

[21] S. Koduru, P. Reddy PVGD, and P. Padala, "Integrated disaster management and smart insurance using cloud and internet of things," *International Journal of Engineering & Technology*, vol. 7, no. 2.6, p. 341, Mar. 2018, doi: 10.14419/ijet.v7i2.6.10777.

[22] C. C. Sekhar, V. V, K. Vijayalakshmi, M. B. Sahaai, A. S. Rao, and S. Murugan, "Cloud-based water tank management and control system," in *2023 Second International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, IEEE, Aug. 2023, pp. 641–646. doi: 10.1109/SmartTechCon57526.2023.10391730.

[23] S. B. Miles, "Participatory disaster recovery simulation modeling for community resilience planning," *International Journal of Disaster Risk Science*, vol. 9, no. 4, pp. 519–529, Dec. 2018, doi: 10.1007/s13753-018-0202-9.

[24] S. H. Alsamhi *et al.*, "UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation," *Drones*, vol. 6, no. 7, p. 154, Jun. 2022, doi: 10.3390/drones6070154.

[25] M. Sharfuddin and T. Ragunathan, "Improving performance of cloud storage systems using support-based replication algorithm," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 17, no. 1, pp. 14–26, Nov. 2022, doi: 10.37936/ecti-cit.2023171.247333.

[26] V. Shaik and N. Kalyanasundaram, “Assimilating sense into disaster recovery databases and judgement framing proceedings for the fastest recovery,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 4, p. 4234, Aug. 2023, doi: 10.11591/ijece.v13i4.pp4234-4245.

[27] I. A. Alameri, J. K. Mutar, A. N. Onaizah, and I. A. Koondhar, “Optimized image processing and clustering to mitigate security threats in mobile ad hoc network,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 1, p. 476, Feb. 2020, doi: 10.12928/telkomnika.v18i1.13914.

[28] T.R. Saravanan, A.R. Rathinam, A. Lenin, A. Komathi, B. Bharathi, and S. Murugan, “Revolutionizing Cloud Computing: Evaluating the Influence of Blockchain and Consensus Algorithms,” in *3rd International Conference on Smart Generation Computing, Communication and Networking*, pp. 1-6, 2023, doi: 10.1109/SMARTGENCON60755.2023.10442008.

[29] M. J. Kumar, S. Mishra, G. R. Elangovan, M. Rajmohan, S. Murugan, and N. A. Vignesh, “Bayesian decision model based reliable route formation in internet of things,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 34, no. 3, pp. 1677-1685, 2024, doi: 10.11591/ijeecs.v34.i3.pp1665-1673.

[30] A. R. Rathinam, B. S. Vathani, A. Komathi, J. Lenin, B. Bharathi, and S. Murugan, “Advances and Predictions in Predictive Auto-Scaling and Maintenance Algorithms for Cloud Computing,” in *2nd International Conference on Automation, Computing and Renewable Systems*, pp. 395-400, 2023, doi: 10.1109/ICACRS58579.2023.10404186.

BIOGRAPHIES OF AUTHORS

Gurumoorthi Gurulakshmanan is an experienced technical project manager with 15+ years of experience leading multiple large-scale programs, developing strategic Vision and Goals, driving digital transformation to the cloud and evolving IT delivery methodologies. Proven record of championing solutions to solve highly-complex business problems and delivering on digital transformation with cloud-native solutions. Strong collaborator, partnered closely with key stakeholders to drive strategic approach to maximize growth. Influential and effective cross-functional team and people leader with a focus on cost optimization initiatives. A thought leader in operational efficiency and team building to ensure the best customer experience with a DevOps mindset. Demonstrated history of working in the industry as an Engineer, Technical Manager and Sr. Program Manager. He can be contacted at email: gurumoorthig198@gmail.com.

Raveendra Nandhavanam Amarnath is an accomplished project manager at Mphasis, boasting over 18 years of invaluable experience in the software industry. Residing in Dallas, Texas, his expertise shines in the realm of testing solutions, emphasizing reliability and automation. His career is characterized by his relentless pursuit of excellence in testing and his dedication to managing complex projects with precision. His extensive technical knowledge and leadership skills have made him an invaluable asset in the world of software and data management. In his current role, Raveendra spearheads the management of one of North America's most ambitious data migration projects, involving the seamless transition of data from Teradata to Google Cloud. His extensive background spans across diverse domains, including Airlines, Banking and Investment, Non-Life Insurance, Pharmacovigilance, and the UK Government. Raveendra's technical prowess encompasses data engineering, ETL (Extract, Transform, Load), Informatica, Teradata, Big Query, as well as mastery in JAVA and .NET applications. He is equally well-versed in the intricacies of SOAP and REST APIs. He can be contacted at email: raveendra.techie@gmail.com.