Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2022, 9(3):193-200

Research Article ISSN: 2394 - 658X

Cloud Without Borders: Software Development Strategies for Multi-Regional Applications

Venkata Baladari

Software Developer, Newark, Delaware, USA
vrssp.baladari@gmail.com

ABSTRACT

Multi-region cloud applications have become crucial for delivering uninterrupted performance, dependability, and adherence to local regulatory requirements. Deploying applications across multiple cloud regions boosts availability, disaster recovery, and user experience but raises complexities including data synchronization, network optimization, security, and cost management. This research examines the responsibilities of software developers in designing and overseeing multi-region cloud-based applications. The main goal is to identify and examine the essential factors and most effective methods that developers should adhere to when constructing reliable, expandable, and high-performance cloud-based applications, approaches to tackle these complexities, such as Infrastructure as Code (IaC) for automating resource allocation, Continuous Integration and Continuous Deployment (CI/CD) for streamlined software updates, and global load balancing for efficient traffic management. The importance of artificial intelligence, automation, and edge computing in streamlining cloud operations is also emphasized. By implementing these strategies, companies can establish cloud applications that are scalable, secure, and highly effective, and run smoothly in various geographic locations.

Keywords: Architecture, Multi-Region, Cloud, Environments, Kubernetes

---

INTRODUCTION

In the present digital age, companies are growingly depending on cloud computing to deploy applications that cater to users worldwide. As businesses expand their global presence, the demand for cloud applications that serve multiple regions has increased substantially. These applications function across a variety of geographically dispersed data centers, which guarantee high availability, minimal latency, and resistance to disruptions. Companies can improve performance, boost user satisfaction, and adhere to regional laws by using cloud computing systems that operate across multiple areas. Software developers have a crucial function in creating, implementing, and updating these applications, guaranteeing that they continue to be adaptable, productive, and secure within a distributed cloud infrastructure.

Overview of Multi-Region Cloud Applications

Cloud-based applications designed for multiple regions are engineered to operate efficiently across various geographical areas through the use of services from cloud providers such as Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. In contrast to deployments which are confined to a single area and can be prone to regional outages and delays, multi-region applications replicate workloads across several data centers. This method enhances efficiency by shortening the gap between users and application servers, resulting in quicker response times. Moreover, this approach boosts fault tolerance by guaranteeing that services remain accessible even if one region encounters disturbances. Multi-region strategies are especially advantageous for companies with a global client base, as they facilitate uninterrupted business operations, enhance disaster restoration capacities, and ensure compliance with data jurisdiction regulations [1],[2],[3].

Importance of Multi-Region Strategies in Software Development

Organizations need to adopt a multi-region cloud strategy in order to develop robust and expandable applications. One of the primary advantages is high uptime and redundancy, since applications can continue operating even in the event of a particular area's failure. Developers can design systems by distributing workloads across multiple regions, which automatically reroute traffic to operational areas, thus reducing system downtime. A significant benefit is low latency, resulting in enhanced user experience through quicker system responses. Locating

application instances near end-users facilitates faster data retrieval and processing, a consideration that is particularly crucial for real-time services including streaming services, gaming platforms, and financial systems [4]. Businesses operating in multiple countries must also consider regulatory compliance and data ownership issues. Numerous governments enforce rigorous rules governing the locations and methods for storing and processing data. Organizations can deploy data across multiple regions while ensuring uninterrupted global operations continue smoothly. In a multi-region setup, disaster recovery and business continuity are enhanced by redundant systems in various locations, which guarantee service availability in the event of natural disasters or cyber-attacks. Performance optimization and efficient traffic management are achieved through the dynamic allocation of resources across multiple regions, allowing for seamless scalability and load balancing, which in turn prevents bottlenecks and optimizes system performance [3],[4].

Research Objectives

The primary objective of this research is to examine the responsibilities of software developers in designing and overseeing multi-region cloud-based applications. The main goal is to identify and examine the essential factors and most effective methods that developers should adhere to when constructing reliable, expandable, and high-performance cloud-based applications. The initiative aims to pinpoint the difficulties encountered in cloud deployments spanning multiple regions, including data synchronization, network delay, and cost minimization, while offering practical solutions. This study also investigates the effects of multi-region strategies on factors including performance, security, and regulatory compliance.

This study aims to examine the role of DevOps practices, automation tools, and deployment methods in improving the efficiency and maintenance of applications deployed across multiple regions. The study also showcases industry leaders through case studies, demonstrating how companies have effectively implemented multi-region cloud architectures to attain scalability and robustness. Additionally, the research delves into developing trends including edge computing, AI-driven cloud optimization, and serverless computing, which are influencing the future of multi-region deployments.

UNDERSTANDING MULTI-REGION CLOUD DEPLOYMENTS

What Are Multi-Region Cloud Applications

Cloud applications spanning multiple regions are software systems that run across various cloud areas supplied by platforms like Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. A cloud region comprises several data centers, deliberately located within a defined geographical zone, to provide dependable services characterized by minimal latency. Deploying an application across multiple regions in a distributed architecture enables improved performance, scalability, and disaster recovery resilience for an IT system, typically spanning two or more geographic areas [1],[2].

These applications utilize distributed computing methods, such as Content Delivery Networks (CDNs), data replication, and traffic routing, to guarantee users are served from the nearest and most efficient location. By distributing workloads across various geographic areas, companies can improve user experience while protecting against unplanned disruptions. Multi-region cloud architectures enable organizations to meet regulatory demands by storing data within designated geographical areas [5],[6].

Benefits of Multi-Region Deployments

Deploying applications across several cloud regions provides a multitude of advantages, making them a popular option for businesses that need high availability and worldwide scalability. A major benefit is the ability to maintain high uptime and quick recovery from disasters. Multi-region deployments enable scalability and load balancing, thereby allowing businesses to dynamically distribute traffic across various locations. Preventing an excessive load on a single data center allows applications to effectively manage sudden increases in user demand. By dispersing services across multiple geographical areas, companies can guarantee that their applications remain available even if a particular area experiences a power failure. In the event of a data centre going offline, traffic is automatically redirected to a different region, thus reducing downtime and improving business resilience. By enabling users to connect to the nearest data center, multi-region cloud deployments expedite request processing and enhance the overall user experience. Applications that necessitate immediate responses, like video streaming, financial dealings, and online gaming, place a significant emphasis on this requirement. Cloud service providers offer automatic scaling capabilities that dynamically allocate resources in line with demand, thus minimizing costs while ensuring high performance levels. Another key benefit is that businesses can optimize content dissemination by employing content delivery networks (CDNs) and regional caching techniques, thereby enhancing the delivery of static and dynamic content, lowering bandwidth expenses and increasing user access globally. Compliance with data regulations is also a significant benefit as numerous nations implement rigorous data sovereignty regulations, mandating that companies store and process data within designated geographic areas. Companies can implement a multi-region strategy to meet regulatory requirements while ensuring smooth worldwide operations.

Challenges in Multi-Region Cloud Computing

Cloud deployments across multiple regions come with various benefits, but also pose several issues that software developers and cloud architects need to resolve. Maintaining data consistency and synchronization across all sources is a major obstacle. Ensuring data consistency across various regions poses a significant challenge for organizations, who need to guarantee that updates are mirrored accurately and uniformly. Some cloud architectures choose strong consistency, which ensures real-time data accuracy, but this can lead to increased latency, whereas others adopt eventual consistency, allowing for minor temporary data discrepancies and as a result, enhancing performance [10].

Effective management of costs and resources is a major challenge. Providing infrastructure support across multiple geographic areas necessitates the allocation of extra resources such as data storage, networking capabilities, and computing power. Companies need to thoroughly assess their financial plans and adjust how they distribute resources in order to achieve a balance between productivity and cost efficiency. Multi-region deployments can also lead to networking and latency problems. Although these architectures are intended to decrease latency for users, interactions between various geographic areas can still cause delays. Developers should put in place efficient methods for replicating data, caching, and directing traffic, in order to reduce performance slowdowns.

Additional obstacles arise in multi-region cloud computing from security and regulatory compliance requirements. Implementing security policies across various geographic locations necessitates robust access controls, encryption methods, and adherence to international regulations including General Data Protection Regulation (GDPR), California Consumer Privacy Act (CCPA), and the Health Insurance Portability and Accountability Act (HIPAA). Companies must guarantee that their data protection strategies are compliant with local laws, thereby increasing the intricacies of cloud security administration [9]. Advanced monitoring and automation tools are essential for overcoming operational and maintenance challenges. Efficiently resolving issues requires observability solutions with robust capabilities, centralized logging, and proactive incident response strategies to monitor application performance across various geographic regions.

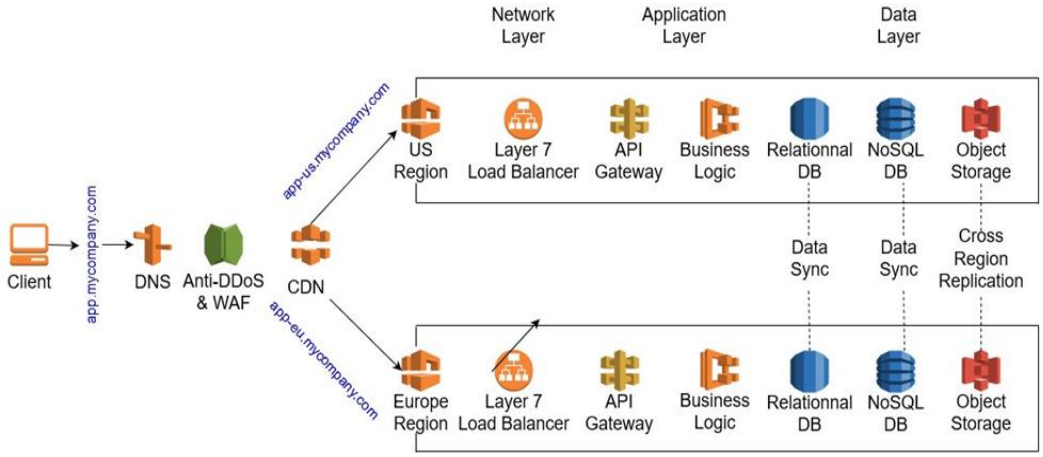

Figure 1: Multi-region architecture

(Accessed from https://www.europeclouds.com/blog/using-the-cloud-to-build-multi-region-architecture)

SOFTWARE DEVELOPER'S ROLE IN MULTI-REGION CLOUD APPLICATIONS

Designing for Scalability and Performance

For multi-region cloud applications, scalability is a crucial attribute, necessitating the ability to manage fluctuating levels of user traffic effectively across multiple locations. Application developers need to create architectures that can adapt and expand as needed in response to changing demand levels without compromising on their optimal functioning efficiency. Achievement of this goal can be attained through the implementation of microservices-based architectures, which split applications into smaller, independent services that can be independently deployed and scaled across various regions.

One way to achieve scalability is through the implementation of auto-scaling features offered by cloud-based infrastructure providers. Auto-scaling allows applications to automatically adjust their computing resources in response to fluctuations in traffic, thereby avoiding resource waste during periods of low demand and guaranteeing adequate capacity during periods of peak activity. To enhance application performance, developers should implement caching strategies, including content delivery networks and edge computing, to decrease latency by retrieving frequently accessed data from the nearest regional server. Implementing database sharding can improve scalability by dispersing database workloads across various instances located in different geographic regions, thus preventing any one node from becoming a performance bottleneck.

Ensuring High Availability and Fault Tolerance

Ensuring high availability and fault tolerance is vital for applications that span multiple regions to provide uninterrupted service, even when regional outages occur. It is essential for developers to create architectures that minimize downtime and are capable of automatic recovery from any failures that may occur. A crucial approach to attaining this is implementing duplicate infrastructure in various geographic areas. Replicating application instances and databases across various locations allows businesses to maintain continuity in the event of an outage in one region.

To improve reliability, developers can implement multi-region redundancy strategies, where traffic is automatically rerouted to a functioning secondary region if the primary region experiences an outage. To achieve this, it's necessary to set up automated failover processes, similar to those offered by cloud providers' disaster recovery services. Developers should integrate circuit breaker patterns into applications to mitigate cascading failures, by briefly halting requests to malfunctioning services and redirecting traffic to functioning instances.

Conducting frequent health checks and ongoing monitoring is crucial for maintaining high availability. Cloud providers supply developers with observability tools, which allow them to identify irregularities, track application performance, and initiate automatic recovery processes when necessary. Developers can gain real-time insights into system performance by integrating logging and monitoring tools thereby resolving issues before they affect users [7],[8].

Implementing Efficient Load Balancing and Traffic Routing

Optimizing performance and averting overloads necessitate the use of load balancing and traffic routing techniques to disperse incoming user requests across numerous regional instances. Software developers need to set up global load balancing solutions that route traffic to the nearest operational data center, taking into account factors including latency, server health, and available capacity. Cloud service providers offer global load balancers, including AWS's Global Accelerator, Google Cloud's Load Balancing, and Azure's Traffic Manager, which route traffic across multiple regions in an intelligent manner [5].

Developers can improve efficiency by implementing geo-based routing strategies that connect users to the nearest regional instance, thereby minimizing latency and speeding up response times. This method is especially advantageous for applications with an international user base, such as e-commerce platforms and streaming services. Furthermore, weighted load balancing can be used to distribute traffic proportionally based on the available resources in each region, thereby preventing overloads and ensuring the optimal use of cloud infrastructure.

Traffic failover policies are a further key aspect to take into account. Automated traffic rerouting should be set up by developers to divert users to backup locations when systems are down. Businesses can reduce the frequency of service interruptions by implementing cloud provider failover strategies in conjunction with DNS-based routing protocols.

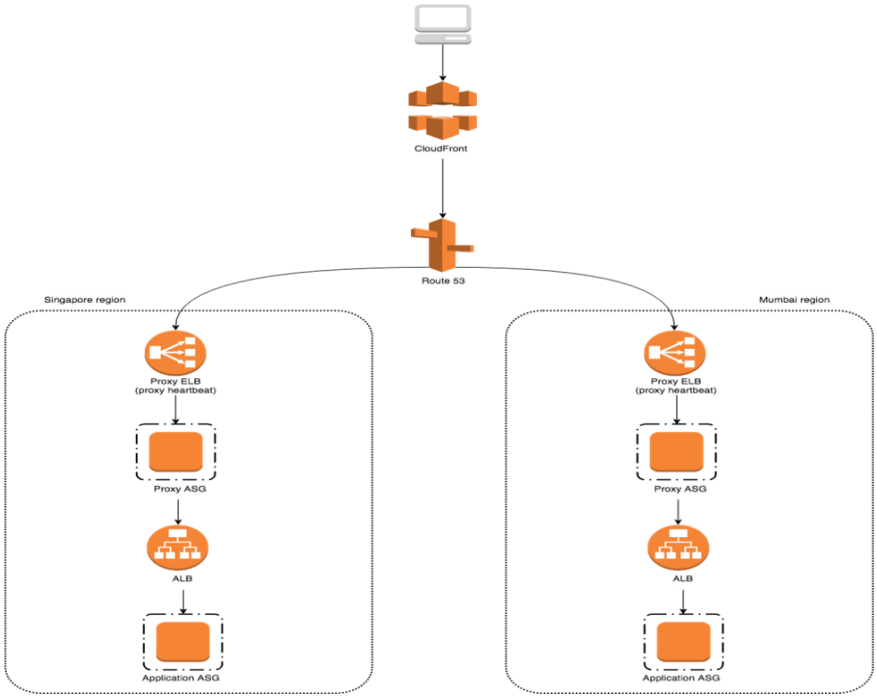

Figure 2: Multi region traffic routing
(Accessed from https://geeks.wego.com/multi-region-traffic-engineering/)

Managing Data Consistency and Synchronization

Maintaining data consistency across numerous cloud regions is a particularly difficult task in multi-region cloud computing environments. Ensuring that data is synchronized across multiple locations is crucial when applications function in various geographic locations, necessitating a unified source of truth. Developers need to make a choice between strong consistency and eventual consistency, depending on the specific needs of their application [11].

For applications needing precise real-time data, strong consistency guarantees that all locations mirror the latest updates simultaneously. This method can lead to increased latency and decreased system performance because of the time needed for synchronization across different regions. In contrast, eventual consistency enables disparate areas to store temporarily inconsistent data, which ultimately stabilizes, but this can result in brief periods of inaccuracy.

Developers can utilize database replication techniques to efficiently manage data synchronization, including active-active replication and active-passive replication. Active-active replication enables multiple regions to process both read and write operations concurrently, leading to lower latency; however, it necessitates conflict resolution mechanisms. Active-passive replication allocates one primary region for write operations while designating other regions for read operations, thereby maintaining consistency, albeit at the potential cost of increased read latency [11].

In situations where updates are happening at the same time in various areas, conflict resolution strategies need to be available to manage discrepancies in the data. Conflicts can be resolved using timestamp-based reconciliation, version control systems, or machine learning algorithms that intelligently handle discrepancies. Cloud service providers offer multi-region database options like AWS DynamoDB Global Tables, Google Spanner, and Azure Cosmos DB, that aid developers in implementing data replication with synchronized consistency models [12].

DEVOPS AND AUTOMATION IN MULTI-REGION CLOUD DEVELOPMENT

Infrastructure as Code (IaC) for Multi-Region Deployments

Defining and managing cloud infrastructure through the use of machine-readable configuration files is a core component of the Infrastructure as Code (IaC) DevOps practice. In a multi-region cloud setting, Infrastructure as Code (IaC) facilitates consistent infrastructure provisioning across various locations, thereby ensuring that application deployments adhere to uniform configurations and compliance requirements. Cloud automation tools, such as IaC, streamline the process of setting up cloud resources by region, thereby minimizing the potential for human mistakes and accelerating deployment times [2].

One of the primary benefits of Infrastructure as Code in multi-region environments is its ability to scale effectively. Firms can leverage Infrastructure as Code scripts to rapidly expand their infrastructure across numerous geographical regions according to user requirements. Automating disaster recovery and failover processes via Infrastructure as Code (IaC) enables backup regions to swiftly take over in the event of primary region failures. Furthermore, IaC facilitates compliance enforcement by allowing security policies and governance rules to be embedded within the underlying infrastructure code, thereby ensuring that all deployments adhere to organizational and regulatory standards.

Software teams can gain increased agility, efficiency, and reliability in managing cloud infrastructure across multiple regions by incorporating Infrastructure as Code into their DevOps workflows. The ability to define infrastructure declaratively and deploy it uniformly across different regions is what makes Infrastructure as Code a vital part of contemporary cloud development approach.

Continuous Integration and Continuous Deployment (CI/CD) Pipelines

Automated code testing, integration, and deployment are key benefits of implementing Continuous Integration (CI) and Continuous Deployment (CD) practices within a DevOps framework. In a cloud environment that spans multiple regions, CI/CD pipelines enable smooth and glitch-free software deployment across geographically dispersed cloud locations. These pipelines aid development teams in automating the software release process, thereby facilitating the timely and uniform deployment of updates across all regions.

Typically, Continuous Integration/Continuous Deployment (CI/CD) pipelines comprise multiple essential stages, beginning with code integration and automated testing. Code alterations by developers are submitted to a common repository, which then activates automated construction and testing processes to confirm the operation's reliability, safety, and efficiency. Developers can leverage tools such as Jenkins, GitHub Actions, GitLab CI/CD, CircleCI, and AWS CodeBuild to identify errors during the development phase and avoid deploying flawed code [13],[14].

Once code has successfully cleared the Continuous Integration stage, the Continuous Deployment process automates the deployment of applications across multiple cloud regions. Tools like AWS CodeDeploy, Google Cloud Build, and Azure DevOps enable teams to deploy updates to various regions with minimal interruption and guaranteed high uptime. Blue-green deployments and canary releases are frequently employed deployment tactics in multi-region settings to mitigate risks. Blue-green deployments require the maintenance of two identical environments, allowing traffic to be switched between them during updates to minimize disruptions. Canary

releases enable teams to roll out changes to selected areas first, then deploy them fully, allowing for performance monitoring and swift rollback in the event of problems.

CI/CD pipelines assist in enforcing security guidelines while automating deployment processes for applications that span multiple cloud regions. Organizations can prevent security breaches and meet regulatory standards by incorporating automated security scans, vulnerability analysis, and compliance reviews into their processes. Observability and monitoring tools can be integrated into CI/CD pipelines to track application performance and identify anomalies in various cloud zones.

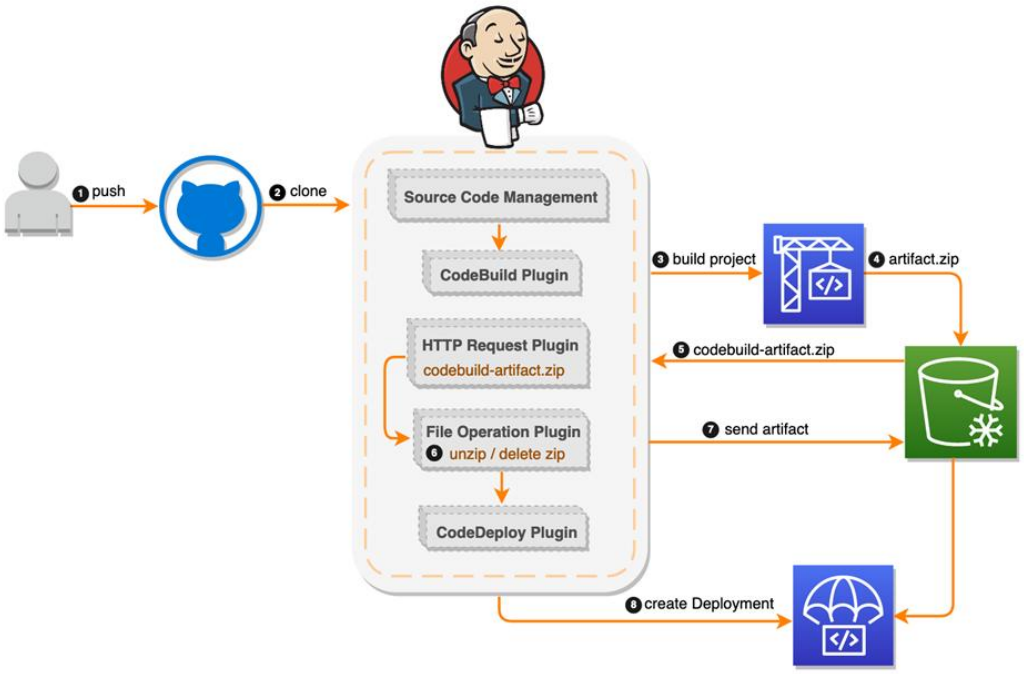

Figure 3: CI/CD Jenkins Pipeline deploy to AWS
(Accessed from https://aws.amazon.com/blogs/devops/setting-up-a-ci-cd-pipeline-by-integrating-jenkins-with-aws-codebuild-and-aws-codedeploy/)

CHALLENGES AND FUTURE CONSIDERATIONS IN MULTI-REGION CLOUD DEVELOPMENT

Overcoming Latency and Network Performance Issues

A key difficulty in multi-region cloud computing is network delay, happening when information moves between widely separated cloud locations. Significant latency can substantially affect the performance of applications, particularly those requiring real-time processing, such as video streaming, financial transactions, and online gaming. In order to address this problem, software developers need to put in place efficient traffic routing and data caching approaches.

Caching frequently accessed data at locations closer to end-users through content delivery networks and edge computing decreasing latency. Cloud providers offer global load balancers and intelligent traffic management systems that optimize network performance by directing users to the closest or most efficient cloud data center. Solutions for inter-region networking, such as AWS Global Accelerator and Google Cloud Interconnect, can reduce latency by offering fast, direct connections between cloud locations.

Mitigating latency problems can also be achieved through data partitioning and replication. Developers can guarantee that users access data from the nearest available location by dispersing workloads and databases across various geographic regions. They will need to reconcile this strategy with data consistency models, as robust consistency methods can cause extra latency due to synchronization delays between areas.

AI and Automation in Multi-Region Cloud Management

Artificial intelligence and automation are assuming a more crucial part in managing multi-region cloud applications. AI-based solutions can assist in optimizing cloud resource allocation, improving security, and automating decision-making processes in order to enhance operational efficiency. Advanced machine learning techniques can study past usage patterns and forecast future resource needs, allowing for dynamic auto-scaling adjustments to be made to computing capacity across various geographic areas.

Cloud-based management tools enabled by artificial intelligence assist businesses in lowering expenses by examining usage habits and suggesting the most cost-efficient resource distribution methods. Self-healing

infrastructure, which relies on artificial intelligence, can automatically identify and rectify system failures, thereby reducing downtime without the need for human involvement.

Tools like Infrastructure as Code (IaC) and event-driven automation frameworks facilitate DevOps teams in simplifying the process of deploying applications across multiple regions and guaranteeing uniformity across various cloud settings. Businesses can streamline their DevOps workflows by incorporating AI technology, thereby enabling the automation of security threat detection, real-time monitoring enhancement, and compliance with regulatory standards across various geographic areas.

Evolving Developer Skill Sets for Multi-Region Cloud Environments

As cloud technologies continue to evolve, software developers need to keep updating their skill sets in order to effectively design and manage complex applications across multiple regions. Cloud-native architecture, microservices, and containerization technologies like Kubernetes and Docker require traditional development methodologies to undergo significant evolution [15]. To facilitate smooth deployment across various cloud regions, developers must acquire proficiency in Infrastructure as Code (IaC), automation frameworks, and DevOps methodologies.

Furthermore, expertise in artificial intelligence and machine learning will become more crucial, as AI-powered cloud management tools keep automating resource allocation, security threat identification, and performance tracking. Developers should also examine observability and monitoring tools, including Prometheus, Grafana, and AWS CloudWatch in order to track and optimize multi-region application performance efficiently [16].

In order to remain competitive in the ever-changing cloud environment, developers need to commit to ongoing learning and cultivate a flexible approach. Businesses should commit resources to cloud certification courses and practical training in order to provide their teams with the expertise necessary for managing the complexities of multi-region cloud systems. Developers can create highly scalable, resilient, and efficient cloud applications by keeping up with the latest trends and industry standards in order to meet the needs of a large user base spread across the globe.

CONCLUSION

Cloud computing across multiple regions is now a critical component for companies seeking to deliver dependable, high-performance, and universally accessible software applications. Placing applications across various cloud regions helps to decrease latency, enhance resilience to faults, and meet data sovereignty compliance requirements. The system also brings about issues including data synchronization, network performance, and security administration. Implementing effective solutions involves the use of automation and the integration of innovative technologies to improve efficiency and dependability. Developers managing multi-region applications should focus on designing scalable systems using microservices and serverless architecture to process diverse workloads efficiently. Automating the scaling of resources, distributing network traffic, and routing it effectively helps maintain continuous performance even with varying levels of traffic. Implementing a comprehensive disaster recovery plan that incorporates automatic failover capabilities and frequent data backups can significantly minimize the effects of localized service disruptions.

REFERENCES

[1]. R. Buyya, C. Vecchiola, and S. T. Selvi, "Cloud platforms in industry," in Mastering Cloud Computing, R. Buyya, C. Vecchiola, and S. T. Selvi, Eds. Morgan Kaufmann, 2013, pp. 315–351. doi: 10.1016/B978-0-12-411454-8.00009-7.

[2]. A. Rahman, R. Mahdavi-Hezaveh, and L. Williams, “A systematic mapping study of infrastructure as code research,” Inf. Softw. Technol., vol. 108, pp. 65–77, 2018.

[3]. P. Tarne, A. Lehmann, and M. Finkbeiner, "A comparison of Multi-Regional Input-Output databases regarding transaction structure and supply chain analysis," J. Clean. Prod., vol. 196, pp. 1486-1500, 2018. doi: 10.1016/j.jclepro.2018.06.082.

[4]. S. Hamadah and D. Aqel, "A Proposed Virtual Private Cloud-Based Disaster Recovery Strategy," 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, 2019, pp. 469-473.

[5]. P. Kumar and R. Kumar, "Issues and Challenges of Load Balancing Techniques in Cloud Computing: A Survey," ACM Computing Surveys, vol. 51, no. 6, Art. no. 120, pp. 1–35, Feb. 2019. doi: 10.1145/3281010.

[6]. F. H. P. Fitzek, P. Seeling, T. Höschele, and B. Jacobfeuerborn, "On the need of computing in future communication networks," in Computing in Communication Networks, F. H. P. Fitzek, F. Granelli, and P. Seeling, Eds. Academic Press, 2020, pp. 3–45. doi: 10.1016/B978-0-12-820488-7.00011-6. https://dl.acm.org/doi/10.1145/3281010

[7]. A. Kushwaha, "Research Paper on AWS Cloud Infrastructure vs Traditional On-Premise," International Research Journal of Engineering and Technology (IRJET), vol. 7, no. 1, pp. 175, Jan. 2020. [Online]. Available: www.irjet.net.

[8]. M. R. Mesbahi, A. M. Rahmani, and M. Hosseinzadeh, "Reliability and high availability in cloud computing environments: a reference roadmap," Human-centric Computing and Information Sciences, vol. 8, no. 20, 2018. [Online]. Available: https://doi.org/10.1186/s13673-018-0143-8.

[9]. A. Hashmi, A. Ranjan, and A. Anand, "Security and Compliance Management in Cloud Computing," in Proc. 3rd Int. Conf. on Computers and Management (ICCM), Jaipur, Rajasthan, India, Jan. 2018, vol. 7, Apex Institute of Engineering and Technology.

[10]. O. Al-Hujran, E. M. Al-Lozi, M. M. Al-Debei, and M. Maqableh, "Challenges of cloud computing adoption from the TOE framework perspective," Int. J. E-Bus. Res. (IJEBR), vol. 14, no. 3, pp. 77–94, Jul. 2018, doi: 10.4018/IJEBR.2018070105.

[11]. Y. P. Faniband, I. Ishak, F. Sidi, and M. A. Jabar, "A review of data synchronization and consistency frameworks for mobile cloud applications," Int. J. Adv. Comput. Sci. Appl. (IJACSA), vol. 9, no. 12, 2018. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2018.091284.

[12]. Wu, Z., Wijaya, E., Uluyol, M., & Madhyastha, H. V. (2018). Bolt-On Global Consistency for the Cloud. In Proceedings of SoCC '18: ACM Symposium on Cloud Computing, Carlsbad, CA, USA, October 11–13, 2018 (SoCC '18), 13 pages. doi: 10.1145/3267809.3267835.

[13]. Amazon Web Services, Serverless Architectures with AWS Lambda: Overview and Best Practices, Nov. 2017.

[14]. F. Grigorio, D. M. de Brito, E. G. Anjos, and M. A. Zenha-Rela, "Using Statistical Analysis of FLOSS Systems Complexity to Understand Software Inactivity," Covenant Journal of Informatics and Communication Technology, vol. 2, no. 2, pp. 1–28, Dec. 2014.

[15]. Kim D, Muhammad H, Kim E, Helal S, Lee C. TOSCA-Based and Federation-Aware Cloud Orchestration for Kubernetes Container Platform. Applied Sciences. 2019; 9(1):191. https://doi.org/10.3390/app9010191

[16]. A. K. K, B. S. Vinutha, and B. V. Vinayaditya, "Real time monitoring of servers with Prometheus and Grafana for high availability," Int. Res. J. Eng. Technol. (IRJET), vol. 6, no. 4, pp. 5093, Apr. 2019. Available: www.irjet.net.