International Journal of Network and Communication Research, 9 (1), 36-51, 2025
ISSN: ISSN 2058-7155(Print),
ISSN: ISSN 2058-7163(Online)
Website: https://www.eajournals.org/
Published of the European Centre for Research Training and Development UK

Multi-Region Networking and Global Traffic Management for AWS

Piyush Dhar Diwan
The Ohio State University, USA
piyush.diwan.cloud@gmail.com

doi: https://doi.org/10.37745/ijncr.16/vol9n13651 Published April 13, 2025

Citation: Diwan P.D. (2025) Multi-Region Networking and Global Traffic Management for AWS, *International Journal of Network and Communication Research*, 9 (1), 36-51

Abstract: *Multi-region cloud networking on AWS has become essential for organizations building resilient, high-performance applications with global reach. This article explores the architecture, benefits, and implementation strategies for creating robust multi-region deployments on AWS. By distributing workloads and data across geographically diverse locations, businesses can enhance availability, reduce latency, ensure regulatory compliance, and strengthen disaster recovery capabilities. The article examines AWS's global networking foundation, including CloudWAN, Transit Gateway, and Global Accelerator, which form the backbone for multi-region architectures. It discusses global traffic management strategies through Route 53 Traffic Flow and CloudFront content delivery. The challenges of data consistency and replication are addressed through various database replication options and S3 Cross-Region Replication. The article emphasizes the importance of automation and observability through infrastructure as code and comprehensive health monitoring. Finally, it outlines best practices for implementing effective multi-region architectures, including establishing clear regional boundaries, implementing consistent tagging, centralizing identity management, designing for eventual consistency, testing failover scenarios, monitoring cross-region metrics, and optimizing for cost efficiency.*

Keywords: global traffic management, disaster recovery, data replication, cloud networking, regulatory compliance

INTRODUCTION

In today's interconnected world, businesses need infrastructure that spans the globe to deliver reliable, low-latency experiences to users regardless of their location. Multi-region cloud networking on AWS has emerged as a critical strategy for organizations seeking to build resilient, high-performance applications

with global reach. The demand for globally distributed applications continues to grow as businesses expand internationally and users expect consistent, high-quality experiences regardless of where they are located. Organizations implementing multi-region architectures can achieve significant improvements in application availability and performance metrics. These architectures support business continuity by ensuring that applications remain accessible even when an entire AWS region experiences disruption. When properly implemented, multi-region deployments can drastically reduce latency for global users by serving content from geographically proximate locations rather than requiring all traffic to traverse back to a single region [1].

The growing demand for multi-region deployments is driven by several factors, including the exponential growth in global digital commerce in recent years. Additionally, regulatory requirements have made geographic data distribution a necessity for many enterprises operating internationally, as data sovereignty laws often require customer data to remain within specific geographic boundaries.

AWS Global Accelerator provides a solution for routing traffic efficiently across regions. This service leverages the AWS global network infrastructure to optimize the network path from users to applications. Global Accelerator uses Anycast IP addresses to route user traffic to the optimal AWS endpoint based on factors including geographic proximity, endpoint health, and routing policies. This advanced traffic management capability allows businesses to implement sophisticated multi-region strategies that balance the needs for performance, reliability, and compliance. The service maintains connection persistence during endpoint failovers, which is essential for maintaining seamless user experiences during regional disruptions or deployments [1].

For organizations requiring comprehensive network management across regions, AWS Cloud WAN offers a centralized way to build, manage, and monitor a unified global network. Cloud WAN simplifies connectivity between cloud and on-premises environments by providing a single dashboard to define network policies that are automatically applied across the global network. This service enables connectivity for thousands of AWS VPCs and on-premises locations through a unified global network spanning multiple AWS regions. With Cloud WAN, network administrators can define policies once and have them consistently applied across the global network, eliminating the complexity of managing region-by-region configurations and ensuring consistent security practices throughout the organization's global infrastructure [2].

This article explores the architecture, benefits, and implementation strategies for creating robust multi-region deployments on AWS, with a particular focus on the networking technologies that enable truly global applications. We'll examine how these services work together to create resilient infrastructures capable of supporting high-performance applications with global reach.

The Case for Multi-Region Deployments

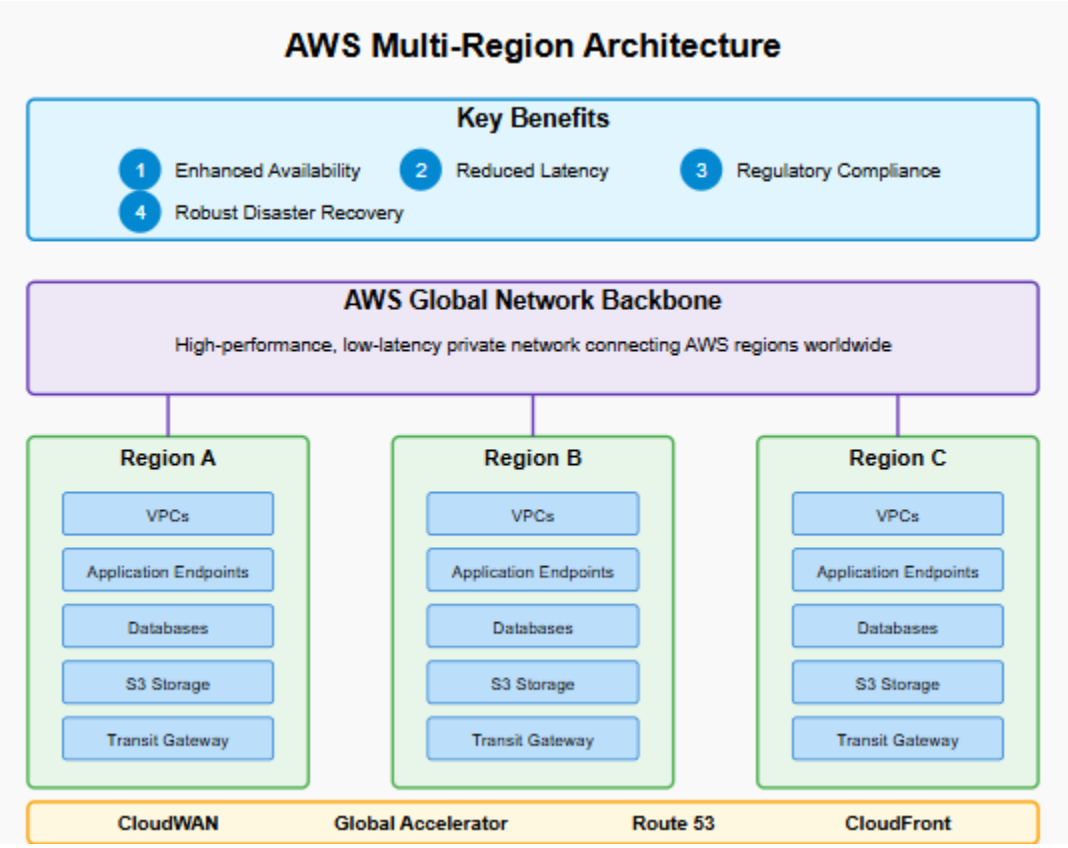
Multi-region architectures distribute workloads and data across geographically diverse locations, providing several key advantages for organizations with global user bases and mission-critical applications. The primary benefit of multi-region deployments is enhanced availability, which has become increasingly important as digital services become central to business operations. By deploying applications across multiple AWS regions, organizations can design systems that maintain operational continuity even when an entire region experiences an outage. AWS Global Accelerator enhances this resilience by continuously monitoring the health of application endpoints across regions and automatically redirecting traffic away from unhealthy endpoints within seconds. This capability ensures that user traffic is only directed to healthy application instances, minimizing disruption during regional incidents and maintaining a consistent user experience even during failover scenarios [3].

Latency optimization represents another compelling reason for implementing multi-region architectures. When applications are deployed across strategically selected global locations, users can access resources from the geographically nearest region, significantly improving response times and overall application performance. AWS Global Accelerator addresses this challenge by utilizing Anycast IP addresses that automatically route user traffic to the closest AWS network entry point and then through the AWS global network backbone to reach the nearest healthy application endpoint. This approach can provide substantial performance improvements compared to public internet routing, particularly for users in geographically remote locations or regions with less developed internet infrastructure. The performance benefit is especially noticeable for clients connecting from international locations, where Global Accelerator can reduce time to first byte by as much as 60% compared to standard internet routing [3].

Regulatory compliance has emerged as a driving factor for multi-region deployments as data sovereignty laws have proliferated worldwide. Organizations operating globally must navigate complex requirements regarding where customer data can be stored and processed. AWS Cloud WAN addresses these challenges by enabling organizations to create isolated network segments for different geographic regions while maintaining unified management. The service's core network policies can be customized to create network segments that adhere to specific regulatory requirements, while the centralized policy management ensures consistent security controls across the global network. This segmentation capability allows businesses to implement data residency controls that respect regional regulations while still maintaining efficient connectivity between authorized regions and resources [4].

Disaster recovery planning becomes more robust with multi-region deployments. Traditional disaster recovery often focuses on recovering from local data center failures, but cloud-native applications require protection against larger-scale events that might affect an entire cloud region. AWS Cloud WAN enhances disaster recovery capabilities by providing automatic routing optimization across the global network. The service continuously monitors network paths across regions and dynamically selects the optimal route based on performance and availability metrics. In the event of a regional disruption, Cloud WAN can automatically reroute traffic through alternative paths to maintain connectivity to healthy regions. The

service supports sophisticated network designs with multiple diverse paths, providing natural redundancy that can significantly reduce recovery times during disruptive events and ensure that critical business operations continue without interruption [4].



AWS Global Networking Foundation

AWS provides several core services that form the backbone of multi-region networking, enabling organizations to build sophisticated global architectures that balance performance, reliability, and security requirements.

AWS CloudWAN

CloudWAN simplifies the management of global wide area networks by providing a central dashboard to build, manage, and monitor a unified network that connects your data centers, branch offices, and AWS VPCs. This service allows organizations to define network policies consistently across their global footprint without managing complex region-by-region configurations. AWS Cloud WAN addresses the challenge of managing complex global networks through its policy-based approach to network management. The service allows network administrators to define core network policies using either a visual policy builder or a JSON policy document. These policies specify connectivity requirements, security controls, and routing behaviors that are then automatically implemented across the global network. Cloud WAN's architecture is built on a

backbone of AWS Transit Gateways that are automatically provisioned and managed by the service, eliminating the operational overhead of manually configuring and maintaining these connections. The service supports complex network designs with multiple network segments that can be used to isolate traffic for different applications, organizational units, or compliance domains while still providing connectivity between segments where appropriate [4].

```
// CloudFormation example for CloudWAN core network policy
{
  "Resources": {
    "GlobalNetwork": {
      "Type": "AWS::NetworkManager::GlobalNetwork",
      "Properties": {
        "Description": "Global network for multi-region connectivity"
      }
    },
    "CoreNetwork": {
      "Type": "AWS::NetworkManager::CoreNetwork",
      "Properties": {
        "GlobalNetworkId": { "Ref": "GlobalNetwork" },
        "PolicyDocument": {
          "CoreNetworkConfiguration": {
            "AsnRanges": ["64512-65534"]
          },
          "Segments": [
            {
              "Name": "prod",
              "Description": "Production segment"
            },
            {
              "Name": "dev",
              "Description": "Development segment"
            }
          ],
          "SegmentActions": [
            {
              "Action": "share",
              "Mode": "attachment-route",
              "Segment": "prod",
              "ShareWith": ["dev"]
            }
          ],
          "AttachmentPolicies": [
```

```
            {
              "RuleNumber": 100,
              "Conditions": [
                {
                  "Type": "tag-value",
                  "Key": "environment",
                  "Value": "production"
                }
              ],
              "Action": {
                "AssociateWithSegment": "prod"
              }
            }
          ]
        }
      }
    }
  }
}
```

AWS Transit Gateway

Transit Gateway acts as a regional network hub, connecting VPCs and on-premises networks through a central point. When implementing multi-region architectures, Transit Gateway peering enables seamless connectivity between regions, creating a unified global network fabric that maintains security and routing policies consistently. Within a multi-region architecture managed by Cloud WAN, Transit Gateways serve as the regional connectivity points that aggregate traffic from various network resources including VPCs, VPN connections, and Direct Connect gateways. Cloud WAN's automation capabilities extend to the management of routing tables and attachments for these Transit Gateways, ensuring consistent configuration across regions while reducing operational complexity. The service automatically establishes inter-region peering connections between Transit Gateways, creating a fully connected global network that optimizes traffic routing across regions. This automated approach to Transit Gateway management ensures that routing policies are consistently applied throughout the global network while eliminating the manual configuration typically required for complex multi-region architectures [4].

AWS Global Accelerator

Global Accelerator improves application availability and performance using the AWS global network. By leveraging edge locations worldwide, Global Accelerator routes user traffic to the optimal regional endpoint based on health, geography, and routing policies, ensuring minimal latency and maximum reliability. The service provides two static Anycast IP addresses that serve as the entry point for application traffic, with these addresses automatically advertised from multiple AWS edge locations to attract user traffic to the

closest point of presence. Once traffic enters the AWS network, Global Accelerator routes it over the AWS global network backbone to the optimal endpoint, which significantly improves performance compared to routing over the public internet. The service includes sophisticated health-checking capabilities that continuously monitor the status of application endpoints across regions and automatically direct traffic away from unhealthy endpoints within seconds. Global Accelerator's traffic management features include traffic dials that allow precise control over the percentage of traffic directed to different endpoint groups, which is particularly valuable during gradual regional migrations or when implementing blue/green deployment strategies across regions [1].

Global Traffic Management Strategies

Effective traffic management is essential for multi-region architectures, enabling organizations to optimize user experience, maintain high availability, and implement sophisticated deployment strategies across geographic boundaries.

Route 53 Traffic Flow

AWS Route 53 provides sophisticated global DNS routing policies that form the foundation of many multi-region traffic management strategies. When implementing multi-region architectures, Route 53 serves as the global traffic manager that directs users to the appropriate regional endpoint based on various factors. The service supports multiple routing policies that can be tailored to specific application requirements and business objectives. As detailed in AWS's multi-region application architecture guidance, Route 53 health checks play a critical role in maintaining application availability by continuously monitoring endpoint health and automatically removing unhealthy endpoints from the routing pool. These health checks can be configured to verify not just basic connectivity but also application functionality by checking specific URLs or API endpoints. For multi-region applications with complex dependencies, Route 53 supports calculated health checks that combine the results of multiple individual health checks to make routing decisions based on the overall health of a regional deployment rather than just individual components. The service also enables organizations to implement weighted routing for gradual migrations between regions, allowing traffic to be shifted incrementally from one region to another while monitoring application performance and user experience [5].

```
# Terraform example for Route 53 multi-region traffic management

# Primary region endpoint (us-east-1)
resource "aws_route53_health_check" "primary" {
  fqdn              = "api-east.example.com"
  port              = 443
  type              = "HTTPS"
  resource_path     = "/health"
  failure_threshold = 3
```

```
  request_interval  = 30

  tags = {
    Name = "Primary-Region-Health-Check"
  }
}

# Secondary region endpoint (us-west-2)
resource "aws_route53_health_check" "secondary" {
  fqdn              = "api-west.example.com"
  port              = 443
  type              = "HTTPS"
  resource_path     = "/health"
  failure_threshold = 3
  request_interval  = 30

  tags = {
    Name = "Secondary-Region-Health-Check"
  }
}

# Latency-based routing with failover
resource "aws_route53_record" "api" {
  zone_id = aws_route53_zone.main.zone_id
  name    = "api.example.com"
  type    = "A"

  latency_routing_policy {
    region = "us-east-1"
  }

  health_check_id = aws_route53_health_check.primary.id
  set_identifier  = "us-east-1"

  alias {
    name                   = aws_lb.primary.dns_name
    zone_id                = aws_lb.primary.zone_id
    evaluate_target_health = true
  }
}

resource "aws_route53_record" "api_secondary" {
```

```
  zone_id = aws_route53_zone.main.zone_id
  name    = "api.example.com"
  type    = "A"

  latency_routing_policy {
    region = "us-west-2"
  }

  health_check_id = aws_route53_health_check.secondary.id
  set_identifier  = "us-west-2"

  alias {
    name                   = aws_lb.secondary.dns_name
    zone_id                = aws_lb.secondary.zone_id
    evaluate_target_health = true
  }
}
```

Content Delivery with CloudFront

AWS CloudFront accelerates content delivery by caching content at edge locations worldwide. This service reduces latency for global users while decreasing the load on origin servers. CloudFront integrates with AWS WAF and Shield to provide security at the edge, protecting your multi-region infrastructure from DDoS attacks and malicious traffic. In multi-region architectures, CloudFront serves as the first line of defense against various security threats while simultaneously improving application performance. The service's integration with AWS Certificate Manager (ACM) simplifies the management of SSL/TLS certificates for secure content delivery, automatically handling certificate renewal and deployment across all edge locations. For applications that must adhere to specific compliance requirements, CloudFront supports field-level encryption that protects sensitive data throughout the entire application delivery path. This capability is particularly valuable for multi-region deployments where data may traverse multiple networks and processing environments. CloudFront's origin failover capability enhances reliability by automatically redirecting requests to a secondary origin when the primary origin is unavailable, which complements Global Accelerator's health-aware routing to provide multiple layers of redundancy in multi-region architectures [3, 5].

Data Consistency and Replication

Managing data across regions requires careful consideration of consistency models and replication strategies, as these decisions directly impact application performance, availability, and compliance with regulatory requirements.

Database Replication Options

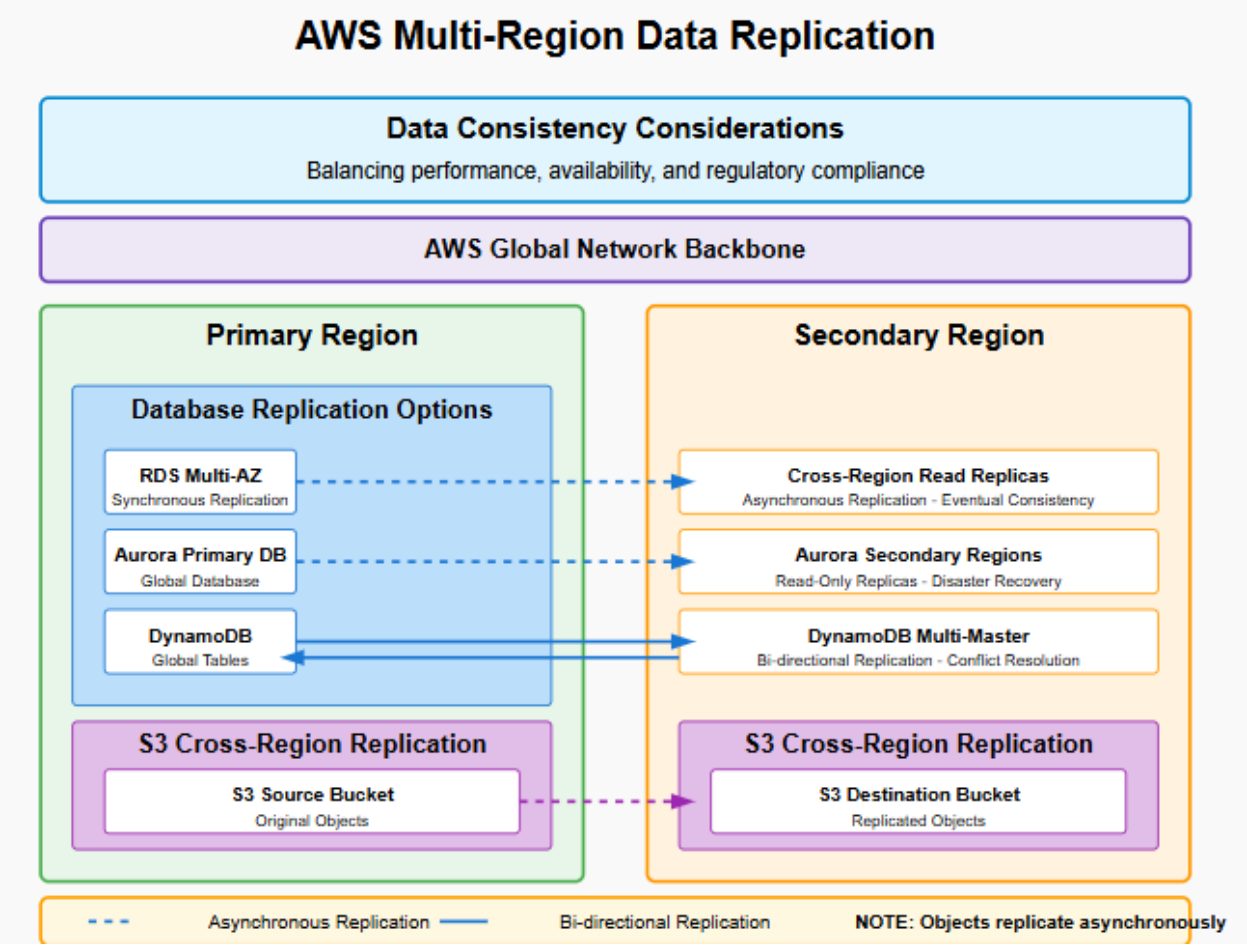
AWS offers several approaches to database replication that support different consistency requirements and access patterns in multi-region architectures. Amazon RDS Multi-AZ with Read Replicas provides a straightforward approach to extending database availability across regions. While the Multi-AZ deployment ensures high availability within a single region through synchronous replication, cross-region read replicas use asynchronous replication to create readable copies of the database in remote regions. This approach is particularly valuable for applications that have read-heavy workloads and can tolerate eventual consistency for read operations. When implementing multi-region database architectures, organizations must carefully consider the latency implications of cross-region replication and design application logic to account for potential data inconsistencies during replication lag. As outlined in AWS's multi-region architecture guidance, applications should be designed with eventual consistency in mind, implementing techniques such as version vectors or conflict resolution handlers to manage potential conflicts that might arise in multi-master scenarios [5].

Amazon Aurora Global Database extends Aurora's capabilities to span multiple AWS regions, allowing a single Aurora database to have primary and secondary regions with read-only replicas. This service was designed specifically for global applications that require both low latency local reads and disaster recovery capabilities across regions. For applications with more demanding consistency requirements, DynamoDB Global Tables provides multi-master, multi-region replication with eventual consistency guarantees. When implementing these database solutions in multi-region architectures, it's important to implement appropriate monitoring for replication lag and health, as replication delays can impact application performance and data consistency. The choice between different database replication options should be driven by specific application requirements for read/write patterns, consistency guarantees, and recovery time objectives during regional failures. For many global applications, a hybrid approach might be appropriate, using different database technologies for different components of the application based on their specific consistency and performance requirements [5].

S3 Cross-Region Replication

For object storage, S3 Cross-Region Replication enables automatic, asynchronous copying of objects across regions. This feature supports compliance requirements, minimizes latency for diverse user bases, and enhances disaster recovery capabilities by maintaining synchronized datasets across geographic boundaries. S3 Cross-Region Replication works by asynchronously copying new objects and updates from a source bucket to one or more destination buckets in different AWS regions. The replication process typically completes within minutes, though the actual time depends on the size of the objects and the distance between regions. Similar to the traffic management patterns employed by Global Accelerator, S3 Cross-Region Replication allows organizations to maintain copies of data close to users in different geographic locations, minimizing the latency associated with accessing content. This capability is particularly valuable for content distribution use cases where the same static assets need to be available globally with minimal latency [1].

The replication configuration supports various options, including the ability to replicate only specific objects based on prefixes or tags, which helps organizations control costs by replicating only necessary data. Organizations implementing multi-region architectures should consider combining S3 Cross-Region Replication with CloudFront for optimal performance and cost efficiency. While replication ensures that content is stored in multiple regions, CloudFront's global edge network further reduces latency by caching content at edge locations closer to end users. This combined approach provides multiple layers of redundancy while optimizing both performance and cost. For compliance-sensitive workloads, S3 replication can be configured to maintain object metadata, ownership controls, and encryption status, ensuring that security policies are consistently applied across regions [1, 5].



Automation and Observability

Effective multi-region deployments require sophisticated automation and monitoring to ensure consistent performance, rapid response to issues, and efficient management of globally distributed resources.

Infrastructure as Code

Using infrastructure as code tools to define infrastructure ensures consistency across regions. This approach eliminates configuration drift between regions and enables rapid recovery or expansion to new regions when needed. According to AWS's best practices for multi-region architectures, organizations should adopt a "templates with parameters" approach that uses a common infrastructure definition with region-specific parameters. This methodology allows for consistent deployment across regions while accommodating regional differences in resource availability or configuration requirements. AWS recommends implementing validation controls that prevent deployment if infrastructure drift is detected, ensuring that all regions maintain consistent configurations over time. For multi-region deployments, organizations should implement a deployment strategy that considers the order of regional updates, with common patterns including canary deployments (testing changes in a single region before wider deployment), blue/green deployments across regions, or parallel deployments for changes that must be implemented simultaneously. The infrastructure-as-code approach should extend beyond basic resource provisioning to include security controls, networking configuration, and application deployment, creating a comprehensive automation framework that ensures consistency across all aspects of the multi-region architecture [6].

Health Monitoring and Failover

AWS CloudWatch, Health Dashboard, and synthetics canaries provide comprehensive monitoring capabilities across regions. Implementing robust health checks and automated failover mechanisms ensures that traffic is dynamically routed away from unhealthy regions without manual intervention. AWS's event-driven approach to multi-region disaster recovery leverages serverless technologies to create highly automated failover mechanisms that can respond to disruptions without human intervention. This architecture uses CloudWatch alarms to detect issues, which then trigger Lambda functions through EventBridge to orchestrate the failover process. The event-driven model enables sophisticated failover logic that can consider multiple factors beyond simple health checks, including performance metrics, cost implications, and application-specific requirements. For applications with complex dependencies, implementing a "failover state machine" using AWS Step Functions provides a structured approach to coordinating the various components involved in the failover process, ensuring that they occur in the correct sequence and with appropriate validation at each step [7].

For comprehensive multi-region observability, AWS recommends implementing a central monitoring dashboard that aggregates metrics, logs, and events from all regions. According to AWS's multi-region fundamentals whitepaper, organizations should establish region-specific thresholds for key metrics that account for normal regional variations in performance and usage patterns. These monitoring systems should include both infrastructure-level application-level metrics to provide a complete view of system health. For multi-region architectures, implementing cross-region metric aggregation is essential for identifying patterns that might not be apparent when viewing regions in isolation. This aggregation can be accomplished using CloudWatch cross-account and cross-region dashboards or by implementing a central logging solution that collects and analyzes data from all regions. The monitoring framework should include

automated alerting with appropriate escalation paths based on the severity and scope of detected issues, ensuring that the right teams are engaged when problems arise [8].

Best Practices for Multi-Region Architectures

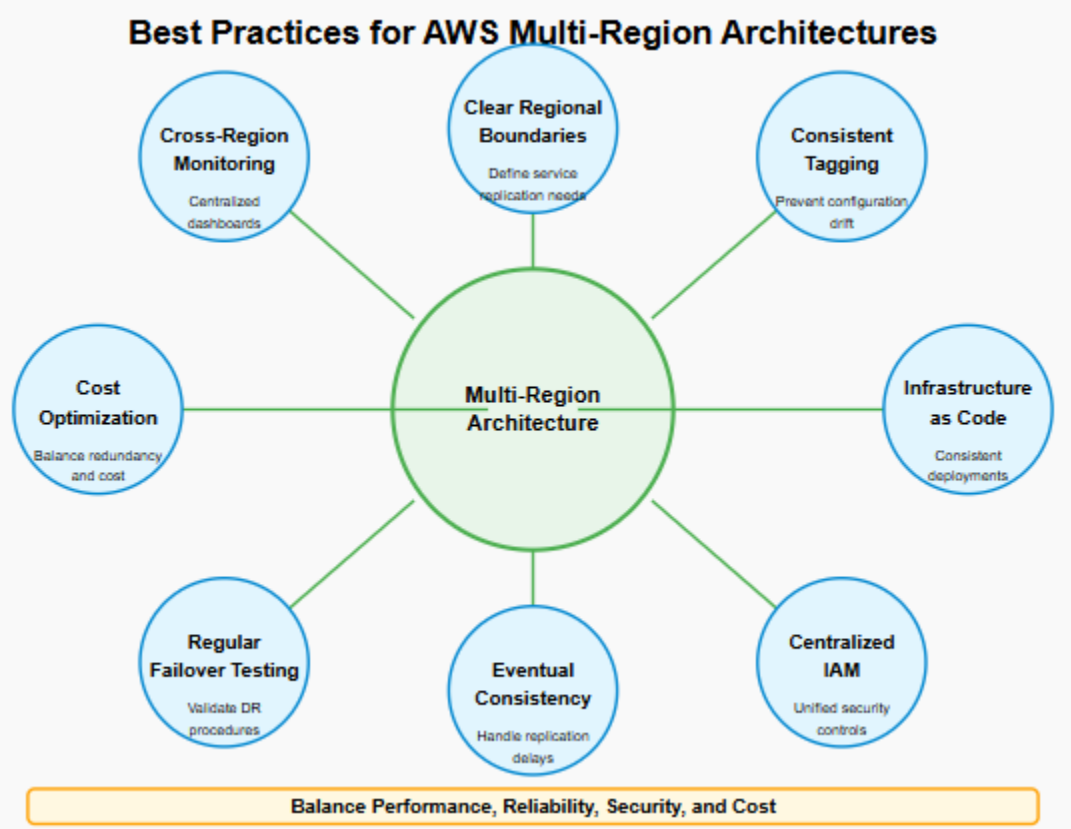
Implementing effective multi-region architectures requires careful planning and adherence to established best practices that balance performance, reliability, security, and cost considerations.Starting with clear regional boundaries is essential for managing complexity in multi-region deployments. Organizations should conduct a thorough analysis of their application components to determine which services must be replicated across regions for availability or performance reasons and which can remain regional to control costs. According to AWS's best practices for multi-region architectures, this service-by-service evaluation should consider factors such as data residency requirements, performance objectives for different user populations, and recovery time objectives. The analysis should identify stateful components that require special consideration for cross-region replication or synchronization, as these typically present the greatest challenges in multi-region architectures. Establishing these boundaries early in the design process creates a foundation for effective decision-making throughout the implementation process [6].

Implementing proper tagging and naming conventions becomes increasingly important as infrastructure scales across regions. Consistent tagging enables accurate cost allocation, resource management, and automation across the global deployment. AWS's multi-region fundamentals guidance emphasizes the importance of implementing standardized tagging strategies through automated deployment processes to prevent configuration drift between regions. For multi-region architectures, region-specific tags can help identify resources that are part of cross-region dependencies or failover groups, which is particularly valuable during incident response or disaster recovery scenarios. Naming conventions should incorporate region identifiers to simplify resource identification while maintaining consistency across the global infrastructure [6].

Centralizing identity and access management through AWS Organizations and IAM provides unified security controls across all regions, reducing administrative overhead and ensuring consistent policy enforcement. This approach allows security policies to be defined once and applied consistently across the global infrastructure, preventing regional variations in security posture that could create vulnerabilities. For multi-region architectures, implementing a least-privilege access model with appropriate separation of duties helps maintain security while enabling efficient operations. AWS recommends using Service Control Policies (SCPs) at the organization level to establish guardrails that prevent unauthorized actions across regions while still allowing teams sufficient autonomy to operate effectively [8].

Designing applications for eventual consistency acknowledges the reality that cross-region data will experience replication delays, which can range from seconds to minutes, depending on the services used and network conditions. Applications must be engineered to handle these delays gracefully, implementing appropriate conflict resolution strategies for scenarios where concurrent updates occur in different regions.

AWS's multi-region fundamentals whitepaper recommends that organizations clearly define their consistency requirements for different data types and implement appropriate replication mechanisms to satisfy these requirements. For many applications, eventual consistency with clearly defined conflict resolution strategies provides an appropriate balance between performance and data integrity [8].



Testing failover scenarios regularly is critical for ensuring that systems work as expected during actual outages. AWS's best practices guidance emphasizes the importance of incorporating disaster recovery testing into routine operational procedures rather than treating it as a separate, infrequent activity. These tests should include both technical verification of failover mechanisms and validation of operational runbooks to ensure that teams are prepared to respond effectively during actual incidents. The testing strategy should include a variety of scenarios, from isolated component failures to complete regional outages, with clear success criteria for each test. Organizations implementing multi-region architectures should establish a regular testing cadence, with results documented and used to drive continuous improvement of both technical systems and operational procedures [6].

Monitoring cross-region metrics through centralized dashboards provides essential visibility into global performance and helps identify emerging issues before they impact users. These dashboards should

aggregate key performance indicators from all regions, with appropriate thresholds that account for normal regional variations. AWS's multi-region fundamentals guidance recommends implementing a comprehensive observability strategy that includes both technical metrics and business KPIs, providing a complete view of system health across regions. This monitoring framework should include automated alerting with appropriate escalation paths based on the severity and scope of detected issues, ensuring that the right teams are engaged when problems arise [8].

Optimizing for cost efficiency while maintaining appropriate redundancy requires careful analysis of both technical and business requirements. Organizations should leverage AWS Savings Plans and Reserved Instances across regions to reduce costs for predictable workloads while maintaining flexibility to scale in response to changing demand patterns. AWS's best practices guidance highlights the importance of understanding the cost implications of different multi-region architectures and making informed trade-offs between cost, performance, and availability. For many organizations, implementing a primary-secondary regional architecture provides an appropriate balance between cost and reliability, with active-active deployments reserved for applications with the most demanding availability requirements [6, 8].

CONCLUSION

Multi-region networking on AWS delivers significant benefits in terms of availability, performance, and regulatory compliance. By leveraging AWS's global infrastructure and services like CloudWAN, Transit Gateway, and Global Accelerator, organizations can build truly global applications that provide consistent, low-latency experiences to users worldwide. Implementing such architectures requires careful planning, a deep understanding of AWS networking capabilities, and a commitment to automation and observability. When done correctly, multi-region deployments provide a competitive advantage by ensuring applications remain resilient in the face of regional outages while delivering exceptional performance to users regardless of their location. As global digital transformation continues to accelerate, multi-region architectures will become increasingly important for organizations seeking to maintain their competitive edge in an interconnected world.

REFERENCES

[1]Tino Tran, "Traffic Management with AWS Global Accelerator," 2019. [Online]. Available: https://aws.amazon.com/blogs/networking-and-content-delivery/traffic-management-with-aws-global-accelerator/

[2] AWS, "AWS Cloud WAN," Amazon Web Services. [Online]. Available: https://aws.amazon.com/cloud-wan/

[3] Amit Kumar and Vikas Purohit, "Deploying Multi-Region Applications in AWS Using AWS Global Accelerator," 2022. [Online]. Available: https://aws.amazon.com/blogs/networking-and-content-delivery/deploying-multi-region-applications-in-aws-using-aws-global-accelerator/

[4] Shubham Singh, Mandar Alankar, and Rizwan Mushtaq, "Achieve Optimal Routing with AWS Cloud WAN for Multi-Region Networks," 2023. [Online]. Available: https://aws.amazon.com/blogs/networking-and-content-delivery/achieve-optimal-routing-with-aws-cloud-wan-for-multi-region-networks/

[5] Joe Chapman and Sebastian Leks, "Creating a Multi-Region Application with AWS Services – Part 1, Compute, Networking, and Security," 2021. [Online]. Available: https://aws.amazon.com/blogs/architecture/creating-a-multi-region-application-with-aws-services-part-1-compute-and-security/

[6] Joe Chapman, "Best practices for creating multi-Region architectures on AWS," 2022. [Online]. Available: https://d1.awsstatic.com/events/Summits/aws-summits/Best_practices_for_creating-multi-Region_architectures_on_AWS_ARC301.pdf

[7] Vaibhav Shah, Cheryl Joseph, and Jyoti Tyagi, "Implementing Multi-Region Disaster Recovery Using Event-Driven Architecture," 2021. [Online]. Available: https://aws.amazon.com/blogs/architecture/implementing-multi-region-disaster-recovery-using-event-driven-architecture/

[8] AWS, "AWS Multi-Region Fundamentals," AWS Whitepaper. [Online]. Available: https://docs.aws.amazon.com/pdfs/whitepapers/latest/aws-multi-region-fundamentals/aws-multi-region-fundamentals.pdf