Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cross-Account ML Service Access Using AWS PrivateLink: Architecture and Governance Models

Sriram Ramakrishnan
Independent Research

ARTICLE INFO

Received: 08 July 2025
Revised: 12 Aug 2025
Accepted: 22 Aug 2025

ABSTRACT

Enterprise organizations increasingly require secure access to centralized machine learning services across multiple AWS accounts while maintaining strict governance controls and operational efficiency. This comprehensive framework introduces a governance-first methodology for implementing cross-account ML service architectures using AWS PrivateLink technology. The proposed framework directly tackles key issues faced when deploying machine learning (ML) within enterprise systems by providing safe network connectivity and removing vulnerabilities associated with being internet-based but still allowing limited access to unique shared ML resources. The framework supports strong Identity and Access Management processes, also includes multi-layered security controls and incorporates compliance auditing capabilities. It also provides capabilities that are required for compliance auditing within the many regulations across diverse industry verticals. The key architectural patterns supported include centralized ML platforms, hub-and-spoke governance models, and hybrid types, allowing for effective use of shared resources while maintaining different security boundaries. The operational strategies supported in the framework include worker nodes with highly sophisticated service discovery, performance auditing and automatic fail-over capabilities to ensure resiliency. Multiregion is taken into consideration with regards to resiliency/disaster recovery, compliance and regulation enforcement through intelligent request routing and data sync capabilities. Cost savings strategies will enable organizations to realize substantial savings in operating costs through capacity planning and resource allocation. The framework also supports the evolution of the organization's ML systems towards serverless and containerized ML platforms, using emerging MLOps capabilities, while still maintaining enterprise security and governance.

Keywords: AWS PrivateLink, Cross-Account Architecture, Machine Learning Governance, Enterprise Security, MLOps

1. Introduction

The proliferation of machine learning workloads across enterprise environments has created a fundamental challenge: how to efficiently and securely share ML services across organizational boundaries while maintaining strict governance controls. Large enterprises now operate ML workloads across multiple AWS accounts, with numerous distinct business units requiring access to centralized ML services [1]. As organizations get increasingly mature in their machine learning operations (MLOps) practices, they will require centralized ML services that can be shared among multiple business units and their frequencies of deployment will shift over time from monthly releases to continuous deployment.

Traditional methods of enabling service access across accounts increasingly depend on connections over the internet and/or complicated VPN sets, both of which create security concerns and add overhead. Organizations using internet-based cross-account access experience higher latency compared to private network solutions, exposed ML endpoints during initial deployment phases. These methods frequently fail to meet the stringent compliance requirements of regulated industries, where financial services organizations and healthcare enterprises require private network connectivity for ML data processing, while simultaneously creating bottlenecks that impede ML innovation and deployment velocity.

The complexity of MLOps implementations across distributed enterprise environments presents unique challenges in model lifecycle management, version control, and service orchestration [1]. Architectures for cross account ML services must consider data governance, model reproducibility, and operational

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

oversight across the organizational silos that are inevitable with all service architectures. Traditional monolithic ML platforms struggle to meet the varied requirements of different business units while providing the necessary centralized control over model quality, security standards, and resource utilization.

AWS PrivateLink presents a compelling opportunity to begin doing this through a secure and scalable way of exposing services across accounts without crossing the public internet. PrivateLink connections provide high uptime service level agreements, and significant reductions in latency for data transfer across accounts compared to the public internet equivalent. PrivateLink allows an organization to create private connectivity between virtual private clouds (VPCs) across the AWS accounts and maintain network isolation, while providing controlled access to central ML resources. Enterprise implementations demonstrate the capability to process substantial volumes of ML inference requests across multiple accounts with minimal response times.

The governance-first approach presented in this review emphasizes the critical importance of establishing robust access controls, audit trails, and service-level agreements before implementing technical solutions [2]. Organizations implementing governance frameworks before technical deployment experience fewer security policy violations and achieve regulatory audit compliance faster than those implementing governance retroactively. This methodology ensures that ML service sharing aligns with organizational security policies, regulatory requirements, and operational excellence standards from the outset, with measured compliance scores demonstrating superior performance compared to traditional implementation approaches. This paper examines the architectural patterns, security considerations, and operational challenges associated with implementing cross-account ML service access using AWS PrivateLink. Through detailed analysis of governance models covering extensive enterprise ML deployments, technical implementation strategies validated across multiple industry verticals, and real-world scaling scenarios supporting substantial concurrent ML model endpoints, a comprehensive framework for enterprise ML service architecture is provided that supports both current operational demands and future growth trajectories.

2. Architecture and Technical Implementation

2.1 AWS PrivateLink Fundamentals

AWS PrivateLink operates on a service provider and service consumer model, where ML services are exposed through Virtual Private Cloud (VPC) endpoints. The architecture eliminates the need for internet gateways, VPN connections, or AWS Direct Connect links, instead creating secure tunnels within the AWS backbone infrastructure. PrivateLink connections demonstrate significantly reduced latencies within the same Availability Zone compared to cross-zone communications, representing substantial improvements over internet-based connectivity patterns [3].

The core architectural components include VPC Endpoint Services on the provider side, which expose specific ML services such as Amazon SageMaker endpoints, model registries, or custom ML APIs running on Amazon ECS or EKS clusters. Enterprise deployments typically support numerous concurrent VPC endpoints per service, with each endpoint capable of handling high throughput volumes and processing substantial request loads under optimal conditions. Service consumers create VPC Endpoints in their accounts, establishing private connectivity to these services through elastic network interfaces (ENIs) deployed within their subnets.

Each VPC Endpoint Service accommodates multiple consumer connections simultaneously, with connection establishment times varying based on network proximity and caching mechanisms. The architecture supports automatic failover mechanisms with rapid recovery capabilities, ensuring high availability for mission-critical ML inference workloads. Network data transfer through PrivateLink endpoints demonstrates consistent throughput rates for sustained workloads and enhanced burst capabilities for short-duration traffic spikes.

2.2 ML Service Exposure Patterns

Several architectural patterns emerge when exposing machine learning services across accounts, each optimized for different organizational structures and performance requirements. The Centralized ML

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Platform pattern involves a dedicated ML account hosting shared services such as model training infrastructure, feature stores, and inference endpoints. Consumer accounts access these services through standardized interfaces, promoting consistency and reducing operational overhead. Implementation experiences show that centralized platforms typically serve multiple consumer accounts simultaneously, maintaining optimal response times for inference requests and model metadata queries.

This pattern demonstrates significant cost optimization benefits, with shared infrastructure substantially reducing per-account ML service costs compared to distributed deployments. Centralized platforms process substantial daily inference request volumes across all connected accounts, with peak loads occurring during standard business hours. Model training workloads benefit from resource pooling, achieving improved GPU utilization rates and reduced training times for standard deep learning models.

The Hub-and-Spoke Model extends this concept by establishing a central ML governance account that manages service catalogs, access policies, and monitoring dashboards. Spoke accounts represent individual business units or development teams that consume ML services according to predefined governance policies. Hub-and-spoke architectures support extensive spoke account configurations per hub, with each spoke maintaining multiple active ML service connections. Central governance hubs process substantial policy evaluations hourly and maintain comprehensive audit trails containing extensive monthly access events.

Hybrid Deployment patterns combine centralized and distributed elements, where core ML infrastructure remains centralized while allowing business units to deploy specialized services that can be selectively shared across the organization [4]. Hybrid implementations demonstrate accelerated deployment times for new ML services and support comprehensive cross-account service discovery for distributed endpoints. These architectures achieve optimal resource utilization by balancing centralized compute resources with specialized edge services for on-demand workloads.

2.3 Network Architecture Considerations

Implementing hub and spoke model for ML services requires careful consideration of network topology and routing policies, with network design directly impacting both performance and security posture. DNS resolution requires careful governance in cross-account ML architectures, as consumer accounts must resolve service endpoints to private PrivateLink ENI addresses rather than public endpoints. This fundamental shift creates governance challenges around service discovery consistency, access control enforcement, and operational oversight across organizational boundaries. Organizations must establish centralized DNS management policies that define naming conventions, resolution hierarchies, and failover procedures for cross-account ML services. DNS governance frameworks should address service endpoint authentication, prevent DNS hijacking attacks that could redirect ML traffic to unauthorized endpoints, and maintain audit trails of service resolution patterns. Enterprise implementations typically require DNS policies that automatically route requests based on account permissions, data classification levels, and regional compliance requirements, ensuring that consumer accounts can only resolve and access ML services they are authorized to use.

PrivateLink provides DNS resolution to the PrivateLink ENI rather than public addresses. Route 53 Private Hosted Zones provide elegant solutions for DNS management, allowing centralized ML accounts to maintain authoritative DNS records while enabling consumer accounts to resolve service names correctly. Private hosted zones support extensive DNS record configurations per zone and handle substantial daily query volumes with high availability guarantees. This approach supports service discovery mechanisms and simplifies client configuration across multiple accounts, significantly reducing service endpoint configuration errors compared to manual IP address management approaches.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Architectural Component	Implementation Approach	Key Performance and Security Characteristics
Centralized ML Platform	Dedicated ML account hosting shared services including model training infrastructure, feature stores, and inference endpoints accessed through standardized interfaces	Promotes consistency across consumer accounts while reducing operational overhead through shared infrastructure and resource pooling for optimal GPU utilization
Hub-and-Spoke Model	Central ML governance account managing service catalogs, access policies, and monitoring dashboards with spoke accounts representing individual business units	Supports extensive spoke account configurations with comprehensive policy evaluation processing and detailed audit trail maintenance for governance compliance
Hybrid Deployment Pattern	Combines centralized core ML infrastructure with distributed specialized services that can be selectively shared across organizational boundaries	Achieves optimal resource utilization by balancing centralized compute resources with specialized edge services while accelerating deployment times for new ML services
VPC Endpoint Services	Provider-side architectural components exposing ML services through private connectivity using elastic network interfaces deployed within consumer subnets	Eliminates internet gateway dependencies while supporting multiple consumer connections with automatic failover mechanisms and consistent throughput rates
DNS and Network Security	Route 53 Private Hosted Zones for centralized DNS management combined with security groups and NACLs for traffic flow control	Simplifies service discovery mechanisms across multiple accounts while maintaining least-privilege access principles and reducing configuration complexity

Table 1: AWS PrivateLink ML Service Deployment Models and Network Infrastructure Comparison [3, 4]

3. Governance and Security Framework

3.1 Cross-Account IAM Policy Architecture

Implementing strong security for cross-account access for ML services requires deep Identity and Access Management (IAM) strategies that satisfy the tension of security and operational efficiency. The governance framework must create a model to manage authentication, authorization, and accountability across accounts, all while adhering to the principle of least privilege. Enterprise implementations typically manage extensive cross-account IAM policies per ML platform, with policy evaluation processes supporting substantial operational demands during peak periods [5].

Resource-Based Policies form the foundation of cross-account access control, with production environments maintaining multiple resource-based policies per ML service endpoint. ML service endpoints, S3 buckets containing model artifacts, and other shared resources must include explicit trust relationships with consumer accounts. These policies should specify not only which accounts can access resources but also the conditions under which access is granted. Well-structured resource-based policies significantly reduce unauthorized access attempts while maintaining minimal authorization latency for legitimate requests.

Cross-account resource policies typically include multiple conditional statements per policy, covering temporal access controls, source IP restrictions, and MFA requirements. Organizations implementing comprehensive resource-based policies report substantially fewer security incidents and achieve high compliance rates with least-privilege principles. Policy validation processes identify and remediate policy violations regularly across enterprise ML deployments.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cross-Account IAM Roles provide a secure mechanism for assuming permissions across account boundaries, with enterprise ML platforms typically defining numerous distinct cross-account roles to support different operational scenarios. The ML platform account establishes roles with specific permissions for different types of ML operations, including model inference roles, training job submission roles, and model registry access roles. Consumer accounts create roles that can assume these cross-account roles when needed, with role assumption processes maintaining high success rates under normal operating conditions.

Cross-account role assumption processes complete efficiently, including credential validation and policy evaluation. Enterprise deployments demonstrate substantial capacity for handling concurrent role assumptions across all consumer accounts, with automatic scaling supporting significant peak loads. Role session durations vary based on operational requirements, with most sessions lasting several hours for batch ML workloads.

Attribute-Based Access Control (ABAC) takes role-based access to a new level by taking into account more dynamic attributes, such as request timestamp, source IP range, or MFA status. In doing this, ABAC allows for policies that are subject to fine-grained access control and provide situational awareness while keeping in line the permissible boundary of security. ABAC implementations in ML environments typically evaluate multiple attributes per access request, with efficient attribute resolution capabilities. Organizations utilizing ABAC report substantially more granular access control capabilities and significant reduction in over-privileged access scenarios compared to traditional RBAC implementations.

3.2 Service Level Security Controls

ML services exposed through PrivateLink require multiple layers of security controls to protect against unauthorized access and ensure data privacy. API Gateway integration provides authentication and authorization capabilities, request throttling, and detailed logging for ML service endpoints. Production API Gateway configurations typically handle substantial request volumes with optimal response times for authentication and complete request processing including backend ML service communication [6]. Request throttling mechanisms maintain service availability by limiting client requests based on service tier and resource allocation. API Gateway implementations demonstrate excellent uptime with automatic failover capabilities recovering rapidly from detected service degradation. Detailed logging captures comprehensive API interactions, generating substantial log data daily for typical enterprise ML service deployments.

3.3 Audit and Compliance Framework

Comprehensive audit capabilities are essential for meeting regulatory requirements and maintaining operational visibility. AWS CloudTrail provides detailed logging of all API calls related to PrivateLink connections, IAM role assumptions, and ML service interactions. Enterprise CloudTrail configurations typically generate substantial audit logs daily, capturing comprehensive administrative actions and data plane operations across cross-account ML service architectures.

Security Framework Component	Implementation Strategy	Governance and Operational Capabilities
Cross-Account IAM Policy Architecture	Sophisticated Identity and Access Management strategies balancing security with operational efficiency across account boundaries while maintaining least privilege principles	Addresses authentication, authorization, and accountability requirements with extensive cross-account policy management supporting substantial operational demands during peak periods
Resource-Based Policies and ABAC	Foundation of cross-account access control through explicit trust relationships combined with Attribute-	Enables fine-grained access control policies that adapt to operational contexts while significantly reducing

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

	Based Access Control incorporating dynamic attributes like timestamps and source IP ranges	unauthorized access attempts and maintaining minimal authorization latency
API Gateway Security Integration	Multi-layered security controls providing authentication, authorization capabilities, request throttling, and detailed logging for ML service endpoints exposed through PrivateLink	Handles substantial request volumes with optimal response times while maintaining excellent uptime through automatic failover capabilities and comprehensive API interaction logging
TLS and Mutual TLS Authentication	Data protection in transit using current encryption standards with mutual TLS authentication requiring client certificate validation for enhanced security	Provides minimal overhead per request with negligible CPU utilization while substantially reducing unauthorized service access attempts and maintaining high certificate validity rates
Comprehensive Audit Framework	Comprehensive audit logging through AWS CloudTrail for all API calls, IAM role assumptions, and ML service interactions, with centralized log aggregation and analysis capabilities	Essential for meeting regulatory requirements and maintaining operational visibility with substantial daily audit log generation and comprehensive administrative action capture

Table 2: Identity Management and Security Architecture Components for Enterprise ML Deployments [5, 6]

4. Operational Excellence and Monitoring

4.1 Service Discovery and Registration

Effective service discovery mechanisms are crucial for managing complex cross-account ML service topologies, with enterprise environments typically maintaining extensive collections of registered ML services across multiple accounts. AWS Service Catalog provides a centralized repository for approved ML services, enabling consumer accounts to discover and provision access to shared resources through self-service portals. Production deployments demonstrate optimized service catalog query response times, with catalog databases supporting substantial service discovery request volumes during peak operational periods [7].

Service catalog implementations typically maintain multiple service portfolios per business unit, with each portfolio containing numerous approved ML service configurations. Consumer account provisioning processes complete efficiently for standard ML service access, including policy validation, resource allocation, and network configuration. Organizations utilizing centralized service catalogs report significant reduction in manual provisioning errors and achieve substantially faster service deployment times compared to manual registration processes.

Amazon EventBridge facilitates event-driven service registration and deregistration, automatically updating service catalogs when new ML services become available or existing services undergo maintenance. EventBridge implementations process substantial volumes of service lifecycle events daily, with efficient event delivery across cross-account notifications. This approach ensures that consumer applications can adapt to service topology changes without manual intervention, with automated service discovery updates completing promptly following service deployment.

Event-driven architectures demonstrate high accuracy in maintaining service registry consistency across distributed environments, with automated reconciliation processes identifying and correcting registry discrepancies rapidly. Service registration workflows handle significant peak loads of simultaneous service deployments, with event processing throughput supporting substantial event volumes without degradation.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Custom service registries built on Amazon DynamoDB or other NoSQL databases provide flexibility for organizations with specific service discovery requirements, supporting high query volumes with optimal response times. These solutions can incorporate business logic for service approval workflows, cost allocation, and usage tracking. Custom registry implementations typically store extensive service metadata records, with automatic data replication maintaining high availability across multiple regions.

4.2 Performance Monitoring and Optimization

Cross-account machine learning service access creates additional network hops and potential latency sources that can be tricky to monitor and optimize. Amazon CloudWatch is very data rich with metrics concerning the performance of the PrivateLink endpoints including connection time, data transfer rates, and errors rates. CloudWatch monitoring configurations could create many individual metric data points each day for successful enterprise ML deployments, while the data points or times to collect the data for some metrics were also optimized for key performance metrics [8].

Application Performance Monitoring (APM) tools like AWS X-Ray can support distributed tracing across account boundaries to some extent, giving visibility to end-to-end request flows and performance bottlenecks as trace serves and state across multiple services. The X-Ray implementations can execute traces that capture and tabulate end-to-end inference requests across the monitoring period and produce large volumes of trace data for standard enterprise ML workloads. Where this is particularly useful is in ML inference pipelines that cross many services and accounts as trace analysis has often identified performance bottlenecks on the most latency encountered issues.

Custom metrics and dashboards that focus on the characteristics of the ML workloads to help operational teams gather insights into service performance, resource consumption, and user experience metrics. Enterprise dashboards vary. They sometimes contain hundreds of actual performance metrics, and refresh often so that there is no lag in the operational visibility as just-in-time displays of performance metrics. These dashboards incorporate both technical metrics such as latency, throughput, and error rates, alongside business metrics including model accuracy, prediction volume, and cost per inference.

4.3 Incident Response and Service Resilience

Cross-account architectures require sophisticated incident response procedures that account for the distributed nature of service dependencies. Automated failover mechanisms using Route 53 health checks can redirect traffic to alternative service endpoints when primary services experience degradation. Organizations implementing automated failover report excellent service availability despite individual service incidents.

Circuit breaker patterns implemented at the client level prevent cascade failures by temporarily disabling calls to unhealthy services while attempting periodic recovery probes. This approach maintains system stability when individual ML services experience issues, with circuit breakers preventing substantial portions of potential cascade failures during service degradation events.

Operational Component	Implementation Approach	Key Capabilities and Operational Benefits
Service Discovery and Registration	AWS Service Catalog providing centralized repository for approved ML services with consumer account self-service portals for resource discovery and provisioning	Enables efficient management of complex cross-account ML service topologies with optimized query response times and substantial service discovery request handling during peak periods
Event-Driven Service Management	Amazon EventBridge facilitating automated service registration and deregistration with real-time catalog updates when ML services become available or undergo maintenance	Ensures consumer applications can adapt to service topology changes without manual intervention through automated service discovery updates and high accuracy in maintaining service registry consistency
Performance Monitoring and Optimization	Amazon CloudWatch providing comprehensive metrics for PrivateLink endpoint performance	Addresses additional network hops and latency sources in cross-account ML service access through substantial daily metric collection and

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

	including connection establishment times, data transfer rates, and error rates	optimized monitoring intervals for critical performance indicators
Application Performance Monitoring	AWS X-Ray enabling distributed tracing across account boundaries with end-to-end request flow visibility and performance bottleneck identification	Particularly valuable for ML inference pipelines spanning multiple services and accounts, capturing comprehensive inference request traces and successfully identifying performance bottlenecks in investigated latency issues
Incident Response and Service Resilience	Automated failover mechanisms using Route 53 health checks with circuit breaker patterns implemented at client level to prevent cascade failures	Sophisticated incident response procedures accounting for distributed service dependencies, maintaining system stability during individual ML service issues and preventing substantial cascade failures during service degradation events

Table 3: Service Discovery, Performance Monitoring, and Resilience Management for Enterprise ML Architectures [7, 8]

5. Scaling Considerations and Future Directions

5.1 Multi-Region Scalability

As machine-learning workloads grow in scope and criticality, organizations often require multi-region deployments of ML systems, to support disaster recovery, regulatory compliance, and performance. Enterprise multi-region ML workloads generally are multi-region deployments within the AWS ecosystem, where there are considerable cross-region data transfer volumes involved with large-scale deployments. Cross-region PrivateLink connections enable secure access to ML services across geographic boundaries while maintaining network isolation, with cross-region latencies varying based on geographic distance and network path optimization [9].

Multi-region PrivateLink implementations demonstrate substantial capacity for handling cross-region ML inference requests daily, with connection establishment times optimized for both initial cross-region connections and cached connections utilizing regional endpoint proximity. Organizations implementing multi-region architectures report excellent service availability during regional outages, with automatic failover processes redirecting traffic efficiently upon detecting regional service degradation.

Global service catalogs must account for regional service availability, data residency requirements, and compliance constraints, typically maintaining extensive collections of regionally distributed service definitions across enterprise deployments. Organizations should develop policies that automatically route requests to appropriate regions based on data classification, user location, and performance requirements. Intelligent regional routing algorithms process substantial volumes of routing decisions hourly, with high percentages of requests automatically directed to optimal regional endpoints based on predefined policies.

Regional service availability matrices demonstrate that typical enterprise ML platforms maintain substantial service availability across all deployed regions, with critical services replicated to ensure comprehensive global availability. Cross-region request routing processes demonstrate efficient routing decision processing, with geographic proximity-based routing significantly reducing end-to-end request latencies compared to single-region deployments.

Data synchronization strategies are important when you need to copy ML models and training data between regions, as enterprise implementations typically copy large amounts of ML artifacts (potentially terabytes) daily across regions. AWS services enable organizations to take advantage of S3 Cross-Region Replication and DynamoDB Global Tables which provide automation of synchronization and have standard or high-speed replication commensurate to and minimize data consistency across regions.

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5.2 Cost Optimization Tactics

Cross-account ML service architectures can generate significant costs through data transfer charges, PrivateLink endpoint fees, and compute resource utilization. Enterprise deployments report substantial monthly cross-account ML infrastructure costs, with data transfer charges and PrivateLink endpoint fees representing significant portions of total expenses. Cost allocation frameworks using AWS Cost and Usage Reports enable organizations to track spending across accounts and business units, facilitating chargeback and showback mechanisms that process extensive cost allocation records monthly [10].

Cost allocation implementations typically distribute expenses across multiple business units, with granular cost tracking achieving high accuracy rates in cross-account expense attribution. Automated chargeback mechanisms process monthly billing allocations with detailed cost breakdowns including compute usage, data transfer charges, storage expenses, and PrivateLink connectivity fees.

Intelligent request routing based on cost optimization algorithms can automatically direct ML inference requests to the most cost-effective service endpoints while maintaining performance requirements. This approach is particularly valuable for batch processing workloads with flexible timing requirements, achieving substantial cost reductions for non-time-sensitive ML workloads through dynamic endpoint selection based on real-time pricing and capacity availability.

5.3 Emerging Technologies and Future Considerations

The rapid evolution of machine learning technologies and cloud computing platforms presents both opportunities and challenges for cross-account ML service architectures. Serverless ML services such as AWS Lambda-based inference endpoints offer improved cost efficiency and automatic scaling capabilities while simplifying operational overhead, with serverless ML implementations demonstrating substantial cost reductions for sporadic inference workloads and extensive scaling capabilities.

Container-based ML platforms using Amazon EKS or ECS provide greater flexibility for custom ML workflows while maintaining the security and governance benefits of PrivateLink connectivity. These platforms support sophisticated service mesh architectures that enable advanced traffic management and security policies, with comprehensive microservices management and inter-service communication processing.

Machine learning operations (MLOps) infrastructures are changing and are phased to include automated model deployment, A/B testing frameworks, and continuous integration/continuous deployment pipelines. Additionally, cross-account ML service architectures must evolve to include these MLOps service integrations while adhering to the security and governance standards.

Scaling/Technology Component	Implementation Strategy	Key Capabilities and Strategic Benefits
Multi-Region Scalability	Enterprise multi-region ML deployments spanning multiple AWS regions with cross-region PrivateLink connections enabling secure access across geographic boundaries while maintaining network isolation	Supports disaster recovery, regulatory compliance, and performance optimization with substantial cross-region data transfer capabilities and optimized connection establishment times for regional endpoint proximity
Global Service Catalogs and Data Synchronization	Comprehensive regional service availability management with automated request routing based on data classification, user location, and performance requirements utilizing S3 Cross-Region Replication and DynamoDB Global Tables	Addresses regional service availability, data residency requirements, and compliance constraints while achieving high data consistency across regions with optimized replication latencies for standard and accelerated configurations

Journal of Information Systems Engineering and Management
2025, 10(59s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cost Optimization Strategies	Intelligent request routing algorithms and cost allocation frameworks using AWS Cost and Usage Reports to facilitate chargeback and showback mechanisms across multiple business units	Enables automatic direction of ML inference requests to cost-effective service endpoints while maintaining performance requirements, particularly valuable for batch processing workloads with flexible timing requirements
Serverless and Container-based ML Platforms	AWS Lambda-based inference endpoints offering improved cost efficiency with automatic scaling capabilities, and Amazon EKS/ECS platforms providing flexibility for custom ML workflows	Demonstrates substantial cost reductions for sporadic inference workloads while supporting sophisticated service mesh architectures that enable advanced traffic management and comprehensive security policy enforcement
Emerging Technologies and MLOps Integration	Machine learning operations platforms incorporating automated model deployment, A/B testing frameworks, and continuous integration/continuous deployment pipelines with edge computing integration	Evolving cross-account ML service architectures to support advanced MLOps capabilities while extending ML inference services to edge locations and IoT devices through secure connectivity models

Table 4: Multi-Region Scalability, Cost Optimization, and Emerging Technology Integration for Enterprise ML Architectures [9, 10]

Conclusion

The implementation of cross-account ML service access using AWS PrivateLink represents a significant advancement in enterprise ML architecture capabilities, enabling organizations to achieve optimal balance between operational efficiency and security compliance through sophisticated governance frameworks. The governance-first methodology detailed throughout this framework provides comprehensive guidance for organizations seeking to implement advanced ML service sharing architectures that scale across complex organizational boundaries. Critical success factors encompass comprehensive IAM policy design, proactive monitoring and alerting systems, and careful consideration of scaling requirements during initial implementation phases to ensure long-term architectural sustainability. The architectural patterns and governance models presented establish foundational elements for next-generation ML platforms that accommodate diverse business unit requirements while maintaining centralized control over security standards and resource utilization. Organizations investing in robust cross-account ML service architectures position themselves to leverage emerging ML technologies, including serverless inference endpoints, container-based platforms, and advanced MLOps capabilities while preserving enterprise-grade security and operational excellence. Future developments should focus on automation frameworks for governance policy management, advanced cost optimization algorithms for multi-account ML workloads, and integration patterns for emerging edge computing and serverless ML technologies. The continued evolution of cloud-native ML services necessitates ongoing refinement of architectural approaches to maintain effectiveness and relevance in rapidly changing technological landscapes, ensuring that enterprise ML platforms remain adaptable to future innovations while preserving established security and governance principles.

References

[1] Amandeep Singla, "Machine Learning Operations (MLOps): Challenges and Strategies," ResearchGate, 2023. [Online]. Available:

Research Article

https://www.researchgate.net/publication/377547044_Machine_Learning_Operations_MLOps_Challenges_and_Strategies
[2] AWS, "Guidance for Governance on AWS." [Online]. Available: https://aws.amazon.com/solutions/guidance/governance-on-aws/
[3] Pega Documentation, "Private connectivity using AWS PrivateLink," 2025. [Online]. Available: https://docs.pega.com/bundle/pega-cloud/page/pega-cloud/pc/pcs-connectivity-privatelink-overview.html
[4] Payoda, "Role of AI and Machine Learning in Hybrid Cloud DevOps" 2025. [Online]. Available: https://www.payoda.com/hybrid-cloud-devops-with-ai-and-machine-learning/
[5] Srikanth Gurram, "Identity and access management in multi-cloud environments: Strategies for enhanced security and governance," World Journal of Advanced Research and Reviews, 2025. [Online]. Available: https://journalwjarr.com/sites/default/files/fulltext_pdf/WJARR-2025-1329.pdf
[6] Joel Paul, "Integrating Machine Learning with API Gateway Security Solutions," ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/385711630_Integrating_Machine_Learning_with_API_Gateway_Security_Solutions
[7] Arash Heidari and Nima Jafari Navimipour, "Service discovery mechanisms in cloud computing: a comprehensive and systematic literature review," ResearchGate, 2021. [Online]. Available: https://www.researchgate.net/publication/352034065_Service_discovery_mechanisms_in_cloud_computing_a_comprehensive_and_systematic_literature_review
[8] Nitin Gouda, "AWS Observability - A Guide to Monitoring Cloud Performance," SigNoz, 2024. [Online]. Available: https://signoz.io/guides/aws-observability/
[9] Shanmugasundaram Sivakumar, "Performance Engineering for Hybrid Multi-Cloud Architectures," ResearchGate, 2021. [Online]. Available: https://www.researchgate.net/publication/386342285_Performance_Engineering_for_Hybrid_Multi-Cloud_Architectures
[10]Ankur Mandal, "Multi Cloud Cost Optimization: A Comprehensive Guide," Lucidity. [Online]. Available: https://www.lucidity.cloud/blog/multi-cloud-cost-optimization