International Journal of AI, BigData, Computational and Management Studies
Noble Scholar Research Group | Volume 7, Issue 1, 186-190, 2026
ISSN: 3050-9416 | https://doi.org/10.63282/3050-9416.IJAIBDCMS-V7I1P129

Original Article

Addressing Technology Challenges in Banking and Financial Institutions Using AWS-Native Architectural Patterns

Tripatjeet Singh
Senior Cloud Engineer,Dallas-Fort Worth, USA.

Received On: 06/01/2026 ***Revised On: 10/02/2026*** ***Accepted On: 16/02/2026*** ***Published On: 28/02/2026***

Abstract: *Banking and financial institutions operate within highly regulated, security sensitive, and complex operational environments. The convergence of legacy system constraints, cybersecurity threats, regulatory mandates, and customer demand for real-time digital services requires a strategic approach to cloud transformation strategies. This paper presents an implementation-driven analysis of enterprise banking challenges and maps them to AWS-native architectural patterns aligned with industry frameworks such as the AWS Well-Architected Framework, and NIST Zero Trust Architecture. The paper uses examples from multi-account setups in regulated industries to create a model that links challenge-to-pattern in areas like modernization, governance, identity, observability, FinOps, disaster recovery, and AI governance. Observational insights and architectural trade-offs are assessed to provide a balanced, practitioner-oriented yet academically grounded contribution.*

Keywords: *AWS, Banking Technology, Financial Services, Cloud Architecture, Regulatory Compliance, Zero Trust, FinOps, Observability, Disaster Recovery, Event-Driven Systems.*

1. Introduction

Financial institutions function in one of the most demanding technology environments today. They are subject to stringent regulatory requirements, constant cybersecurity threats, and have zero to little tolerance for system downtime. At the same time, consumers demand real-time balance updates, immediate payment confirmations, and smooth digital experiences.

Many financial institutions today still use core systems which were built decades ago on mainframes. These systems typically rely on batch processing as well as tightly coupled application integrations. In contrast, modern applications such as mobile banking apps, APIs, fraud detection systems and so on utilize distributed, cloud-native architectures. This leads to a hybrid architecture environment where legacy infrastructures coexist with scalable cloud platforms. The makes stability and agility always at odds with each other.

Public cloud computing platforms such as Amazon Web Services (AWS) provide programmable infrastructure, elastic scaling, on-demand compute/storage and built-in governance capabilities. However, cloud adoption in financial services is not as straightforward as migrating applications and data to cloud. It needs to be integrated with established AWS Well-Architected Framework [1], cybersecurity and regulatory standard frameworks, such as Zero Trust Architectural principles [2] and NIST CSF 2.0 [3]. In regulated environments, architecture decisions are not only technical, but they are also compliance and risk decisions.

While the adoption of cloud technology has been increasing in the industry, there is still a noticeable gap between high-level architectural guidance and the practical implementation of secure, multi-account cloud environments in banking. Many academic discussions focus on scalability and distributed systems theory, but fewer address governance-first design in highly regulated financial ecosystems.

The intent of this paper is to help bridge gap between these two areas. First, this paper identifies key technology challenges faced by banking institutions in today's marketplace, maps those technology challenges against AWS-native architectural patterns, and offers implementation-driven insights learned from enterprise-scale environments. Second, this paper provides a well-structured and practical approach to cloud modernization that uses both innovation and regulatory compliance as its balancing principles.

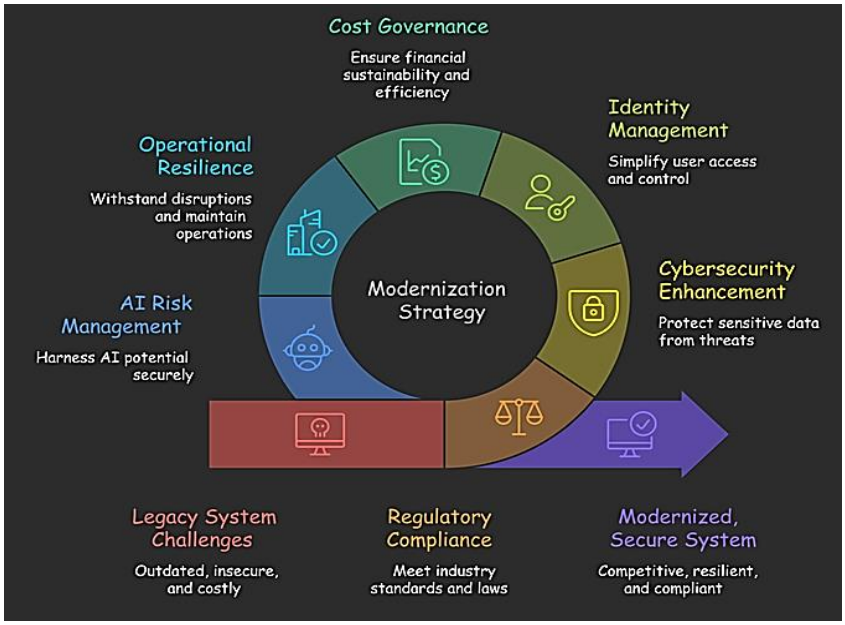

Fig 1: Modernization Strategy

2. Legacy Modernization in Banking

Core banking systems were developed well before the advent of real-time digital services. These systems were built on mainframe technology, used COBOL applications and still utilize batch processing systems that only update data overnight when they run a batch process. This model is quite stable but does not meet today's customer expectations for instant balance updates, real-time payments, and seamless integration through APIs. The difference between batch-based core banking systems and real-time digital channels can cause inconsistent system states and increase middleware complexity , particularly in regulated financial environments [8].

Modernizing systems must happen incrementally rather than making a complete overhaul of an entire operational environment all at once. Attempting to replace a core banking system is too much of a risk. By using a well-known software development pattern, the Strangler pattern, you can allow cloud-native services to slowly take on the responsibility of very specific pieces of work, while retaining the legacy system as the source of truth. In AWS environments, Change Data Capture using AWS DMS can stream updates into Amazon EventBridge, triggering Lambda or containerized services. This event-driven approach reduces latency from hours to seconds and improves modularity without putting additional stress on your core operations, , consistent with architectural best practices outlined in the AWS Well-Architected Framework [1].

3. Regulatory Compliance and Governance

Ensuring compliance to the regulatory standards in the banking domain is part of their daily operations rather than an activity that is performed periodically. The auditors request evidence of access controls, encryption policies, and their adherence during each audit they conduct. When there are multiple AWS accounts in an organization, it can often be challenging to ensure consistency of controls applied to each individual account without a centralized governance model , particularly when aligning to frameworks such as NIST CSF 2.0 [3] and NIST SP 800-53 Rev. 5 [4].

AWS Organizations and Service Control Policies (SCPs) help enforce preventative guardrails across accounts. Organization-wide CloudTrail logging and AWS Config provide traceability and continuous compliance with respect to resource configuration [7]. Encryption through AWS KMS and immutable storage using Amazon S3 Object Lock facilitate the preservation of evidence.

In reality, centralized compliance architectures have decreased the time required for audit preparation by 30–50 percent, shifting compliance from reactive document process to continuous and automated enforcement process integrated directly into the architecture, aligning with modern cybersecurity risk management outcomes defined in NIST CSF 2.0 [3].

4. Zero Trust Security Model

Banks and other financial institutions are constantly facing threats such unauthorized access, abuses of APIs, and credential compromise. The old model of security, which focuses on protecting the perimeter network, will no longer be adequate to address the risks associated with banking operations today and into the future. The Zero Trust security concept changes the traditional model by shifting the network trust concept to an identity-based verification model. For any interaction to be initiated, the systems must authenticate and authorize [2][6] the identity that initiated the interaction.

In AWS environments, IAM Identity Center enables centralized identity control across accounts. Amazon GuardDuty detects anomalous activity within the environments. Additionally, AWS WAF and Network Firewall secure the application and network layers from being attacked, consistent with AWS Security Reference

Architecture guidance [7]. VPC endpoints and multi-account segmentation reduce exposure and limit blast radius.

In practice, the implementation of Zero Trust architectures has reduced lateral movement risks and improved detection times [6]. Using identity as the basis of security and adding monitoring and segmentation to support it will allow for increased security without sacrificing scalability or operational flexibility.

5. Enterprise Identity and Access Management

Managing access in a multi-account cloud environment is not an easy task, and many financial services organizations depend on enterprise identity providers to implement these access controls, particularly in alignment with identity-centric security principles defined under Zero Trust Architecture [2]. Without adequate automation, access provisioning can take much longer to complete than expected due to the number of accounts that must be accessed. The manual role assignments increased the risk of access provisioning delays, as well as it can also prolong the onboarding process for members of your organization.

AWS IAM Identity Center allows you to have a centralized way of controlling access to your systems and applications. You can federate your identities through identity providers such as Microsoft Entra ID and use the SCIM protocol to automatically create user groups and update existing ones when provisioning your users [5]. Permission Sets provide a standardized way of managing role-based access control (RBAC) across multiple accounts and reduce the amount of configuration drift between the accounts [7]. For external users, Amazon Cognito provides secure authentication without exposing internal identity systems.

In enterprise environments, automated identity workflows have reduced onboarding time from days to hours. Additionally, these workflows have identified other areas of dependencies, such as SCIM synchronization latency, which proves the importance of implementing a good monitoring system and processes that enable fallback options. A well-designed identity architecture will strengthen the security as well as improve operational efficiency.

6. Real-Time Data Processing and Fraud Detection

Banks and financial institutions must detect instances of fraudulent behavior in near real time. If they depend on nightly batch processing to identify suspicious transactions, this creates a gap where these transactions can go unnoticed. Modern payment systems generate continuous streams of events, and the detection systems must work to identify those events as quickly as they occur, consistent with the Detect function outlined in NIST CSF 2.0 [3].

A cloud-native and more-accurate solution for fraud detection would be based on the ingestion of transaction events through Amazon Kinesis Data Streams. After ingestion, event data can be enriched and/or routed via Amazon EventBridge and subsequently processed through AWS Lambda functions (update in real time). For inference, traditional fraud models can be deployed as managed endpoints in Amazon SageMaker, allowing low-latency scoring at scale [1]. Amazon Bedrock would be used to access foundational models to support advanced pattern identification, anomaly detection and clarification, or summarization of risk [10].

In enterprise implementations, this streaming architecture has enabled fraud decisions within seconds rather than hours. By combining event-driven processing with scalable model endpoints, institutions move from reactive batch detection to continuous, real-time risk evaluation.

7. Observability and Incident Response

As financial organizations migrate their systems to cloud-native, distributed architectures, gaining visibility into the application's operational capabilities has become more complicated. Logs and metrics are often distributed across multiple AWS accounts and environments, which makes it difficult for teams to quickly locate the data they need to address the incident quickly, leading to increased mean time to resolution (MTTR) and higher risk for operational failure [3].

A centralized observability approach helps address this problem. CloudWatch Cross-Account Observability can aggregate metrics and logs across environments into a single monitoring layer. Log streams can be delivered through Kinesis Firehose into Amazon OpenSearch Service, where they are indexed and searchable in near real time [7]. Central dashboards and automated alerts give operations teams a consistent view of system health [1].

In enterprise implementations, this model has reduced MTTR by approximately 30–40 percent. More importantly, it has improved root cause identification across distributed services. Observability, when designed intentionally, becomes not just a monitoring function but a core part of operational resilience.

8. FinOps and Cost Governance

Cloud platforms provide flexibility and scale, but that flexibility can also introduce cost unpredictability. Large enterprises looking to manage several dozen accounts within AWS have had challenges with clarity of visibility into their cloud costs and consistency of compliance with their budgets [8]. Without appropriate management and structured oversight, organizations frequently see costs begin to escalate before business leaders are even aware.

A centralized FinOps approach within the organization helps bring discipline to cloud spending. AWS Cost and Usage Reports (CUR) can be exported to Amazon S3 and a detailed analysis of costs by account is performed using Amazon Athena. Additionally, by using Amazon QuickSight dashboards, finance teams and senior executives can visualize their cloud cost trends and usage patterns across the different environments. Budget creation and alerts can further be automated to provide the financial guardrails needed by an

organization to ensure that cloud cost limitations are enforced at the time of new account and project onboarding, aligned with the cost optimization pillar of the AWS Well-Architected Framework [1].

In enterprise environments managing multiple accounts, this model has significantly improved cost transparency and enabled proactive budget enforcement. FinOps, when embedded into the architecture, aligns cloud spending with business accountability and supports long-term cloud maturity.

9. Disaster Recovery and Operational Resilience

In financial services, downtime is more than an inconvenience that can impact customer trust and regulatory standing. With region-level outages, network failures and configuration errors, financial institutions need to be able to quickly failover their systems with minimum disruption of service. Therefore, disaster recovery should be incorporated into the architecture, rather than being an afterthought [3].

Multi-region strategies using Amazon Route 53 enable DNS-based failover when a primary endpoint becomes unavailable. Replicating data through services such as Aurora Global Database and cross-region S3 replication, financial institutions can ensure they are able to continue to access their data even if there is a region outage [9]. However, having a failover design on paper alone won't provide confidence that your failover actually works, so we must test our systems using the AWS Fault Injection Simulator regularly to confirm our assumptions about failover under controlled conditions.

By regularly testing disaster recovery processes, we can improve our ability to recover from an outage and find areas of improvement for both automation and configuration consistency. These improvements help to align our disaster recovery practices with the expectations of the regulatory authorities for operational resilience [4], thereby strengthening both our technical and compliance posture.

10. AI Governance and Responsible Innovation

In the financial domain, artificial intelligence is increasingly being used, ranging from fraud detection to customer support. However, there are several critical considerations surrounding these AI systems: explainability, data privacy, and regulatory compliance. In a regulated industry, a model must not only perform well, but it must also be transparent and properly governed [10].

By utilizing AWS services such as Amazon Bedrock and SageMaker, organizations host their models within private VPC environments or use PrivateLink to decrease the risk of exposure to external threats. Data governance through AWS Lake Formation helps control who can access sensitive datasets, while Amazon Comprehend can assist in identifying and masking personally identifiable information (PII). These above-mentioned controls mitigate risk exposure to unauthorized data during model training and inference [11][12].

Enterprise experience shows that strong AI governance frameworks improve risk visibility, but technical safeguards alone are not sufficient. Human-in-the-loop model for validating AI output, is critical to reviewing output, monitor performance of the AI models and ensuring regulatory alignment and ethical standards [10].

11. Challenge-to-Pattern Mapping Framework

Modernizing banking systems is not only about adopting new cloud services. It requires a structured way to connect real business challenges to repeatable architectural patterns. In this paper, enterprise banking challenges are grouped into practical domains such as modernization, security, identity, governance, observability, cost management, resilience, and AI oversight.

For each domain, AWS-native patterns provide consistent and scalable solutions. This structured mapping helps avoid one-off implementations and promotes standardization across accounts and environments. In addition to this, they can further ensure compliance with referenced, widely accepted industry frameworks such as the AWS Well-Architected Framework [1], NIST Zero Trust Architecture [2], NIST Cybersecurity Framework (CSF) 2.0 [3], and the NIST AI Risk Management Framework (AI RMF) [10]. By clearly linking challenges to patterns and recognized standards, institutions can improve architectural consistency, reduce compliance risk, and better connect technology decisions to business and regulatory expectations.

Table 1: Challenge-To-Pattern Mapping Overview

Challenge Domain	AWS-Native Pattern	Industry Framework Alignment
Modernization	CDC (DMS), EventBridge, ECS/EKS (Strangler Pattern)	AWS Well-Architected Framework [1]
Identity	IAM Identity Center, SCIM (Entra ID), Permission Sets, Amazon Cognito	NIST Zero Trust Architecture [2], NIST Digital Identity Guidelines [5]
Security	GuardDuty, WAF, Network Firewall, VPC Segmentation	NIST Zero Trust Architecture [2], NIST CSF 2.0 [3]
Governance	AWS Organizations, SCPs, CloudTrail, AWS Config	NIST CSF 2.0 [3], NIST SP 800-53 Rev.5 [4]
Observability	CloudWatch Cross-Account, OpenSearch	NIST CSF 2.0 – Detect/Respond [3], AWS Well-Architected [1]
Cost (FinOps)	CUR (S3 + Athena), QuickSight, Budgets	AWS Well-Architected – Cost Optimization Pillar [1]

Resilience	Route 53 Failover, Aurora Global DB, FIS	NIST CSF 2.0 – Recover Function [3], AWS DR Guidance [9]
AI Governance	Bedrock, SageMaker, Lake Formation, Comprehend	NIST AI Risk Management Framework (AI RMF) [10]

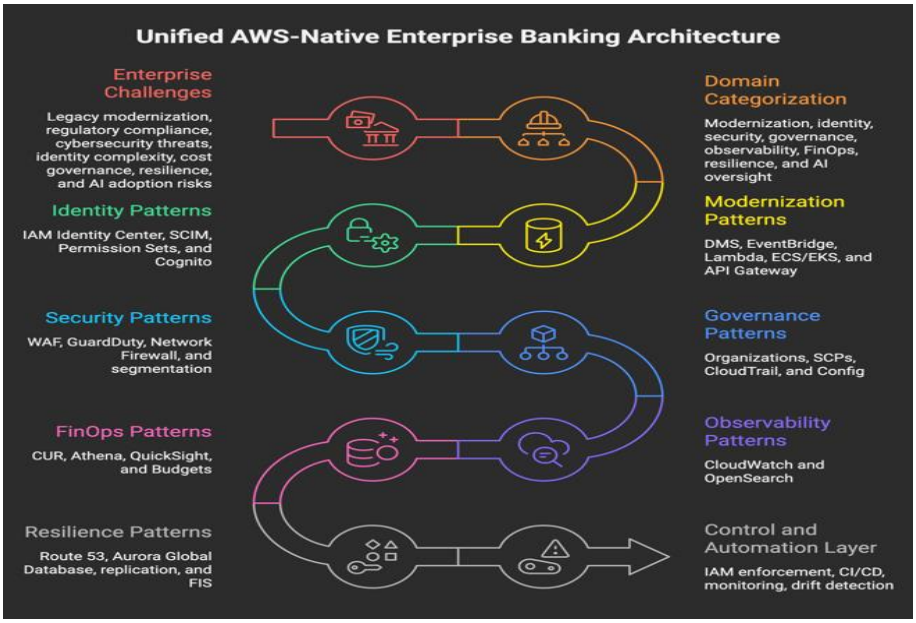

Fig 2: Unified AWS-Native Enterprise Banking Architecture

12. Limitations and Trade-Offs

While AWS-native architectures offer clear benefits, they also introduce significant trade-offs that requires acknowledgement. Multi-account environments improve isolation and the overall governance, but they also increase operational complexity [1]. To manage identity, networking, logging, and deployment across dozens of accounts, organizations need to implement a robust DevSecOps automation and disciplined processes [7].

The use of strict governance controls, such as Service Control Policies (SCPs), can mitigate compliance risk [3], however if designed poorly can slow development. Real-time, event-driven architectures improve responsiveness, but streaming services and replicating data across region can add a lot of cost [1]. Federated identity systems can also introduce delays or other dependency risk when provisioning user access.

Finally, when adopting AI, you will need to deal with/consider additional challenges associated with escalated governance requirements, as well as explainability and data governance requirements [10].

These trade-offs do not reduce the overall benefits of moving to the cloud; however, you will need to provide thoughtful design to maximize the opportunity of cloud transformation. Any successful modernization project depends on successfully balancing agility, cost, security and compliance, rather than just optimizing one dimension.

13. Conclusion

Financial establishments are continuously challenged to be successful, yet they cannot afford to compromise stability, security, or compliance. Legacy systems, strict regulatory expectations, and rising customer demands make transformation complex [3].

This paper has shown that AWS-native architectural patterns, when applied thoughtfully, provide a practical path forward [1]. Event-driven modernization reduces latency without destabilizing core systems. Zero Trust security strengthens protection in an evolving threat landscape [2]. Centralized observability improves operational response, while structured FinOps governance brings financial discipline. Multi-region resilience ensures that availability and regulatory expectations are met.

The motivation for modernization in banking is not compulsory adoption of new technologies, it is about leveraging structured, governance-aligned patterns that provide an equal balance between agility and accountability. With the right architectural approach, financial institutions will have the confidence to innovate while preserving the operational integrity and regulatory trust [10].

References

1. Amazon Web Services, *AWS Well-Architected Framework*, 2024. [Online]. Available: https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome.html
2. National Institute of Standards and Technology (NIST), *Zero Trust Architecture*, NIST SP 800-207, Aug. 2020.

[Online]. Available: https://doi.org/10.6028/NIST.SP.800-207
3. National Institute of Standards and Technology (NIST), *Cybersecurity Framework (CSF) 2.0*, Feb. 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf
4. National Institute of Standards and Technology (NIST), *Security and Privacy Controls for Information Systems and Organizations*, NIST SP 800-53 Rev. 5, Sept. 2020. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-53r5
5. National Institute of Standards and Technology (NIST), *Digital Identity Guidelines*, NIST SP 800-63-3 (Updated 2023). [Online]. Available: https://pages.nist.gov/800-63-3/
6. Cybersecurity and Infrastructure Security Agency (CISA), *Zero Trust Maturity Model Version 2.0*, Apr. 2023. [Online]. Available: https://www.cisa.gov/resources-tools/resources/zero-trust-maturity-model
7. Amazon Web Services, *AWS Security Reference Architecture*, 2025. [Online]. Available: https://docs.aws.amazon.com/prescriptive-guidance/latest/security-reference-architecture/introduction.html
8. Amazon Web Services, *Financial Services Industry Lens*, 2024. [Online]. Available: https://docs.aws.amazon.com/wellarchitected/latest/financial-services-industry-lens/welcome.html
9. Amazon Web Services, *Disaster Recovery of Workloads on AWS: Recovery in the Cloud*, 2024. [Online]. Available: https://docs.aws.amazon.com/whitepapers/latest/disaster-recovery-workloads-on-aws/recovery-in-the-cloud.html
10. National Institute of Standards and Technology (NIST), *AI Risk Management Framework (AI RMF 1.0)*, Jan. 2023. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf
11. Amazon Web Services, "Enabling AI Adoption at Scale Through Enterprise Risk Management Framework," AWS Security Blog, 2025. [Online]. Available: https://aws.amazon.com/blogs/security/enabling-ai-adoption-at-scale-through-enterprise-risk-management-framework-part-1/
12. Amazon Web Services, "Operational Risk Management and AI for Banks and Financial Services Customers," AWS Industries Blog, 2026. [Online]. Available: https://aws.amazon.com/blogs/industries/operational-risk-management-and-ai-for-banks-and-financial-services-customers/
13. Kaur, M., Bonkra, A., Verma, R., Khanna, N., Maken, P., & Sunkara, S. K. (2025). Comparative study of traditional and hybrid models in short-term financial forecasting using machine learning. In Innovations in Computing (pp. 13-18). CRC Press.