Investment Domain Modernization: Data Ingestion & Egress Patterns in AWS

Version: 1.0

Date: July 8, 2025

Author: [Your Name]

# Table of Contents

1. 1. Overview
2. 2. Data Storage Formats
3. 3. Ingestion Patterns (Data Ingress)
4. 4. Consumption Patterns (Data Egress)
5. 5. Common Interfaces and Services
6. 6. Logging & Monitoring
7. 7. Pros, Cons, Limitations, and Security Considerations
8. 8. Appendix

# 1. Overview

This document outlines the various data flow mechanisms for ingestion and consumption within the AWS ecosystem for Investment Domain Modernization. It highlights storage formats, ingress and egress methods, supported services, and security/logging practices.

# 2. Data Storage Formats

- Amazon S3 (Iceberg Table Format): Used for large-scale, ACID-compliant, analytical datasets.

- Amazon Redshift: Columnar storage for fast querying and BI integrations.

# 3. Ingestion Patterns (Data Ingress)

## A. Pull-based Ingestion

- API Calls: Applications or services fetch data using APIs.

- Scheduled DB Calls: Using JDBC/ODBC to fetch from databases.

- SCP/SFTP Scheduled Pulls: Automated file transfers.

- SharePoint/Confluence: Documents extracted manually or via automation.

## B. Push-based Ingestion

- Transfer Family (SFTP): Secure file transfers to S3.

- API Push: Clients push data directly to APIs.

- AWS CLI/SDK (not recommended): Direct S3 uploads.

- NAS to S3 (AWS DataSync): Sync files from network shares.

- AWS Glue: ETL pipeline to ingest, clean, and transform data into S3 or Redshift.

- Lambda: Serverless event-driven ingestion via API Gateway or S3 triggers.

- API Gateway: Entry point for real-time or batch ingestion via REST/HTTP APIs.

# 4. Consumption Patterns (Data Egress)

## A. Push-based Egress

- SSH/SCP to Target Server: Secure file transfers.

- SFTP from S3: Using Transfer Family to send to external systems.

- NAS Sync: Using AWS DataSync to move files from S3.

- SharePoint Upload: Manual or automated file transfer.

## B. Pull-based Egress

- Athena/Redshift Access: Using JDBC/ODBC.

- API Calls: Services query or retrieve data.

- S3 Shortcuts (MS Fabric): External access via shortcuts.

- IAM User Access: Credential-based download or access.

- Glue or Lambda jobs: Query or transform S3 data for downstream applications.

# 5. Common Interfaces and Services

- Amazon Athena / Redshift: SQL-based querying.

- AWS DataSync: NAS integrations.

- Transfer Family: SFTP support.

- SharePoint/Confluence: Document access.

- JDBC/ODBC Drivers: Database access.

- MS Fabric (S3 Shortcuts): Non-AWS integration.

- AWS Glue: ETL and transformation jobs.

- AWS Lambda: Event-based processing.

- API Gateway: Secure API entry points.

# 6. Logging & Monitoring

- CloudTrail: API activity tracking.

- S3 Access Logs: Bucket-level access information.

- Athena Query History: Tracks user queries.

- Redshift Audit Logs: Tracks queries and access.

- AWS CloudWatch: Monitors Lambda/API Gateway/DataSync.

- Glue Job Metrics: ETL performance monitoring.

# 7. Pros, Cons, Limitations, and Security Considerations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | Pros | Cons | Limitations | Security |
| API Ingestion | Real-time, Flexible | Requires API dev | Rate-limited | IAM/Auth, HTTPS |
| SFTP (Transfer Family) | Secure, Compatible | Setup overhead | Max 5k sessions | Key-based auth |
| JDBC/ODBC | Standardized | Connection mgmt | No streaming | VPC, Auth creds |
| DataSync | Reliable, Parallel | Costly | Not real-time | IAM roles, VPC config |
| MS Fabric Shortcuts | External Access | Access control | Manual management | Azure AD, S3 policy |
| SharePoint/Confluence | Familiar UI | Manual steps | Unstructured data | Role-based access |
| Athena | Serverless | Cold start latency | Query limits | Fine-grained permissions |
| Redshift | High performance | Costly | Scaling issues | IAM roles, Network ACL |
| Glue | ETL automation | Can be complex | Long run times | KMS encryption, IAM |
| Lambda | Event-driven | Timeout limits | Memory limits | Execution role, VPC config |
| API Gateway | Secure API entry | Throttling | Payload size limits | IAM/Auth, WAF |

# 8. Appendix

- IAM Best Practices: Use least privilege principle. Use roles instead of users where possible.

- Data Classification: Ensure tagging and classification for regulatory compliance.

- Encryption: S3 SSE, KMS for data at rest. TLS for in-transit.

- Automation Suggestions: Use Step Functions or EventBridge for scheduling and workflow control.