IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15350

Serverless ELT Pipeline for Scalable Data Processing Using AWS Glue and AWS Lambda

Isacc Moses S[1], Mr. Daniel Nesa Kumar C[2]

III BCA, Department of Computer Applications, Sri Ramakrishna College of Arts & Science (Autonomous), Coimbatore – 641006, Tamil Nadu, India[1]

Assistant Professor, Department of Computer Applications, Sri Ramakrishna College of Arts and Sciences (autonomous), Coimbatore – 641012, Tamil Nadu, India[2]

Abstract: The Serverless ELT (Extract, Load, Transform) pipeline leveraging AWS Glue and AWS Lambda offers a modern, fully managed approach for handling large-scale data processing workloads. By adopting a serverless architecture, this solution eliminates the need for manual server provisioning, configuration, and maintenance, allowing organizations to focus solely on data operations and insights. AWS Lambda serves as the orchestration layer, automating the triggering of data ingestion workflows from multiple sources including relational databases, APIs, and object storage systems. This event-driven execution ensures that data pipelines operate efficiently, respond to changes in real-time, and scale seamlessly without manual intervention. Once ingested, data is loaded into Amazon S3, forming a centralized, durable, and secure data lake. AWS Glue then handles automated data cataloging, schema discovery, and transformation using its distributed Apache Spark-based processing engine. The serverless nature of this pipeline provides significant operational and cost advantages through pay-as-you-go pricing, automated logging, monitoring, and error handling features that reduce operational overhead and enhance system reliability.

Keywords: AWS Lambda, AWS Glue, Serverless Architecture, ELT Pipeline, Amazon S3, Cloud Computing, Data Transformation, Apache Spark, Amazon CloudWatch, Python.

I. INTRODUCTION

In today's data-driven world, organizations generate vast volumes of data from diverse sources such as databases, APIs, IoT devices, and cloud applications. Efficiently extracting, transforming, and loading (ETL) this data into a centralized repository is crucial for analytics, reporting, and machine learning initiatives. Traditional ETL systems often rely on dedicated servers and complex infrastructure, which can be costly, difficult to scale, and require continuous maintenance. A serverless ELT pipeline using AWS Glue and AWS Lambda addresses these challenges by offering a fully managed and scalable solution for data processing. AWS Lambda, a serverless compute service, orchestrates data workflows and triggers automated ingestion processes based on events or schedules. Meanwhile, AWS Glue, a managed ETL service, enables data cataloging, schema discovery, and distributed transformations using its Spark-based engine. Together, they allow seamless integration of data from multiple sources into Amazon S3, forming a secure and centralized data lake.
The benefits of adopting a serverless ELT pipeline extend beyond scalability and performance. By leveraging event-driven execution, organizations can optimize operational efficiency and reduce costs through pay-as-you-go pricing. Automatic monitoring, logging, and error handling further simplify pipeline management while enhancing reliability. Furthermore, serverless ELT pipelines support faster time-to-insight by reducing the delays associated with traditional ETL setup and maintenance, allowing data engineers and analysts to focus more on data quality, transformation logic, and analytics rather than infrastructure management.

II. RELATED WORK

Traditional ETL (Extract, Transform, Load) pipelines are widely used to move data from multiple sources into centralized storage for analytics and reporting. These systems typically rely on dedicated servers or on-premises infrastructure to execute scheduled extraction, transformation, and loading tasks. While functional, such systems face several limitations when handling large-scale or rapidly changing datasets, including manual provisioning, configuration, and maintenance of servers.

Research on serverless computing has demonstrated that AWS Lambda significantly reduces infrastructure overhead while providing automatic scaling capabilities. Event-driven serverless architectures have been shown to improve system efficiency and reduce operational costs compared to traditional approaches. Studies have further highlighted that

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15350

Lambda-style functions are well-suited for event-driven, stateless workloads such as data transformation and pipeline orchestration.

AWS Glue has been recognized as a scalable and reliable managed ETL service that provides automated data cataloging, schema inference, and Spark-based distributed transformations. Its integration with Amazon S3 and the Glue Data Catalog creates a powerful foundation for modern data lake architectures. Various studies have also explored event-driven scheduling using Amazon EventBridge and monitoring using CloudWatch. However, few platforms combine all these capabilities into a fully serverless, end-to-end ELT pipeline — a gap this project directly addresses.

III. EXISTING SYSTEMS AND DRAWBACKS

In many organizations, data collection and processing are still handled through traditional server-based ETL pipelines. These systems require significant hardware resources, maintenance costs, and human effort to operate. They are batch-oriented, processing data at fixed intervals, which results in delayed insights and limits the ability to perform real-time analytics.

Key drawbacks of existing systems include:

- Manual Data Processing – Traditional ETL systems require manual setup and execution of data extraction, transformation, and loading tasks.
- No Real-Time Processing – Batch-oriented pipelines process data at fixed intervals, causing delays in analytics and reporting.
- Poor Scalability – Existing systems struggle to adapt to fluctuating workloads or sudden spikes in data volume.
- High Infrastructure Costs – Servers need to remain operational even during low workload periods, leading to inefficient utilization.
- Security and Compliance Risks – Managing data security, encryption, and access controls manually increases the risk of breaches.
- Lack of Automation – Scheduling, monitoring, and alerting often require custom scripts and manual oversight.
- Limited Accessibility – On-premises systems cannot be accessed easily from remote locations or cloud platforms.

IV. OBJECTIVES AND CHALLENGES

Primary Objectives

- Design a fully serverless ELT pipeline using AWS cloud services eliminating the need for server management.
- Implement scalable cloud-based data ingestion using AWS Lambda triggered by events or schedules.
- Develop AWS Glue transformation scripts using Spark for large-scale data cleaning, normalization, and aggregation.
- Enable centralized, secure, and durable data storage using Amazon S3 as the primary data lake.
- Automate data cataloging and schema discovery using the AWS Glue Data Catalog.
- Monitor system performance, execution logs, and errors using Amazon CloudWatch.

Development Challenges

Configuring IAM roles and policies with least-privilege access to allow Lambda and Glue to securely interact with S3 and other services required careful design. Ensuring correct schema handling during Glue transformations, including type conversions, deduplication, and null value management, added complexity to the transformation logic. Designing event-driven triggers that respond to S3 events in near real-time while also supporting scheduled batch runs required balancing two execution models within the same architecture.

V. SYSTEM ARCHITECTURE

The system follows a serverless, event-driven layered architecture designed for scalability, reliability, and cost efficiency. The architecture eliminates the need for dedicated server infrastructure by delegating all compute, storage, and orchestration responsibilities to AWS managed services. Each layer serves a distinct purpose and communicates through well-defined interfaces.

Source data originates from relational databases, REST APIs, and object storage systems. AWS Lambda functions detect incoming data events or scheduled triggers and initiate the ingestion and transformation workflows. AWS Glue performs the core ELT operations — extracting raw data, applying cleaning and transformation logic, and loading processed

DOI: 10.17148/IJARCCE.2026.15350

datasets into Amazon S3. The Glue Data Catalog maintains metadata, schema definitions, and table structures for all datasets. Amazon CloudWatch continuously monitors execution logs and performance metrics throughout the pipeline. The architecture layers are organized as follows:

- Source Data Layer – Databases, APIs, Files
- Orchestration Layer – AWS Lambda (Event Detection, Trigger, Error Logging)
- ETL Processing Layer – AWS Glue (Extract, Clean, Transform, Load)
- Storage Layer – Amazon S3 Data Lake
- Metadata Layer – AWS Glue Data Catalog
- Analytics Layer – Amazon Athena / BI Tools / ML Pipelines
- Monitoring Layer – Amazon CloudWatch

VI. IMPLEMENTATION

Data Ingestion Module
The Data Ingestion Module is responsible for extracting data from multiple sources including relational databases, APIs, and object storage systems. AWS Lambda functions automate the extraction process, triggered by scheduled events or incoming data notifications. The module validates incoming data for completeness, consistency, and correct formatting before loading it into Amazon S3 as raw datasets.

Data Transformation Module
The Data Transformation Module uses AWS Glue to process and transform raw data into structured, analytics-ready formats. Glue's Spark-based engine handles large-scale data efficiently, performing operations such as type conversions, deduplication, normalization, and aggregation. Outputs are stored in Parquet or CSV format, partitioned and compressed for optimal performance and storage efficiency.

Orchestration Module
The Orchestration Module coordinates execution of the ELT pipeline. AWS Lambda functions trigger data ingestion, initiate Glue jobs, and manage the flow of datasets between raw, transformed, and analytics-ready stages. Event-driven execution allows the system to respond to new data in near real-time while supporting batch processing for large datasets.

Monitoring and Logging Module
This module leverages AWS CloudWatch to monitor pipeline performance, track job execution, and log errors or warnings. Automated alerts are generated for failures or performance bottlenecks, allowing administrators to quickly identify and resolve issues.

Security and Access Module
The Security and Access Module enforces strict access controls using AWS IAM roles and policies. Sensitive data stored in S3 is encrypted at rest and in transit, while Lambda and Glue roles restrict operations to authorized functions only. This ensures that data remains secure and compliant with industry best practices.

Metadata Management Module
AWS Glue Data Catalog maintains metadata for all datasets, including table names, schemas, data types, and partitions. This module ensures discoverability, consistency, and easy querying of datasets, supporting analytics, reporting, and machine learning workflows.

Deployment Module
AWS Lambda functions and Glue jobs are version-controlled and deployed through AWS CloudFormation or CI/CD pipelines. The serverless deployment eliminates the need for dedicated infrastructure while enabling automatic scaling, high availability, and efficient resource utilization across development, testing, and production environments.

VII. EVALUATION RESULTS AND DISCUSSION

The system was evaluated by uploading CSV datasets to Amazon S3 and triggering both Lambda orchestration and Glue ETL jobs. Generated outputs were verified for data accuracy, correct transformations, and schema consistency across multiple test runs. Integration with Amazon Athena confirmed that processed datasets were queryable immediately after pipeline completion.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15350

Module / Component	Accuracy (%)	Avg. Response Time	Reliability (%)
Data Ingestion (Lambda)	98	~2.3 sec	99
Data Transformation (Glue)	99	~4.5 sec	99
Event-Driven Orchestration	100	—	100
S3 Data Storage & Load	100	~1.5 sec	99
CloudWatch Monitoring	100	~0.5 sec	100

Table 1. System Evaluation Results

The Lambda function consistently orchestrated pipeline stages and triggered Glue jobs without errors. Glue transformations handled deduplication, type conversion, and null value management reliably across multiple dataset sizes. The serverless model proved highly cost-effective since compute charges are incurred only during function execution with no idle server costs. The system demonstrated strong scalability potential as Lambda and Glue automatically handle concurrent invocations and increasing data volumes.

transaction_id	customer_name	product_name	category	quantity	unit_price
T10007	Denise Hall	Backpack	Sports & Outdoors	9	449.84
T10301	Amy Coleman	Biography	Books	1	185.7
T10629	Andrew Strickland	Toaster	Home & Kitchen	1	422.23
T10831	Brian Taylor	Cookware Set	Home & Kitchen	3	370.17
T10962	Krista Lawson	Biography	Books	9	445.9

Table 2. Sample Processed Dataset Output (Amazon Athena Query)

VIII. CONCLUSION

The Serverless ELT Pipeline using AWS Glue and AWS Lambda was successfully developed and validated. The system demonstrates how modern cloud computing services can be integrated to build an efficient, scalable, and fully automated data processing pipeline without the overhead of traditional server-based infrastructure. By combining AWS Lambda for orchestration, AWS Glue for distributed transformations, and Amazon S3 as a centralized data lake, the system achieves high accuracy, reliability, and cost efficiency.

The modular and automated design of the pipeline improves reliability, maintainability, and operational efficiency. Event-driven workflows allow data ingestion and transformation to occur automatically in response to triggers or schedules, while Glue's Spark-based engine ensures large datasets are processed efficiently. Monitoring and error-handling mechanisms provided by AWS CloudWatch enhance visibility and operational control, ensuring that pipeline failures or performance issues can be quickly detected and resolved.

Overall, this project demonstrates that a serverless ELT pipeline can significantly reduce operational overhead while improving data quality, consistency, and accessibility. The architecture is secure, scalable, and future-ready. By implementing this pipeline, organizations can achieve faster insights, more reliable analytics, and a flexible framework for modern data-driven applications.

IX. FUTURE ENHANCEMENTS

Although the proposed system successfully automates data ingestion, transformation, and loading, several enhancements can further extend its capabilities:

- Real-time data processing using Amazon Kinesis Data Streams or Apache Kafka to enable streaming analytics alongside batch workflows.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15350

- Integration of machine learning models within AWS Glue for anomaly detection, predictive transformations, and intelligent data quality checks.
- Metadata versioning and data lineage tracking through the Glue Data Catalog or a dedicated data governance tool for regulatory compliance.
- Interactive analytics dashboards using Amazon QuickSight or Redshift for business intelligence reporting on processed datasets.
- Automated data archiving and retention policies to manage storage costs efficiently in Amazon S3.
- Expansion to support multi-source data aggregation from multiple DynamoDB tables, external REST APIs, and third-party data platforms.

By implementing these enhancements, the system could evolve into a comprehensive cloud-based business intelligence and data engineering platform capable of handling complex, multi-dimensional data processing tasks at enterprise scale.

REFERENCES

[1]. Amazon Web Services, "AWS Glue Developer Guide," AWS Official Documentation, 2024.
[2]. Amazon Web Services, "AWS Lambda Developer Guide," AWS Official Documentation, 2024.
[3]. Amazon Web Services, "Amazon S3 Documentation," AWS Official Documentation, 2024.
[4]. Amazon Web Services, "Amazon CloudWatch User Guide," AWS Official Documentation, 2024.
[5]. M. Syahrul Mubarok and M. Izman Herdiansyah, "Implementasi Cloud Computing Amazon Web Services (AWS)," KLIK: Kajian Ilmiah Informatika & Komputer, vol. 4, no. 2, Oct. 2023.
[6]. Jonas, E. et al., "Cloud Programming Simplified: A Berkeley View on Serverless Computing," UC Berkeley TR, 2019.
[7]. Armbrust, M. et al., "A View of Cloud Computing," Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.
[8]. A. Shizuka Hutagaol et al., "Designing a Data Warehouse to Optimize the Hotel Booking Monitoring Dashboard," INTECOMS J. Information Technology & Computer Science, vol. 8, no. 5, 2025.
[9]. Jeong Yang and Anoop Abraham, "Analyzing the Features, Usability, and Performance of Deploying a Containerized Mobile Web Application on Serverless Cloud Platforms," Future Internet, vol. 16, no. 12, Dec. 2024.
[10]. "Serverless Data Processing and ETL Workflows with AWS Glue," ResearchGate, 2026.

