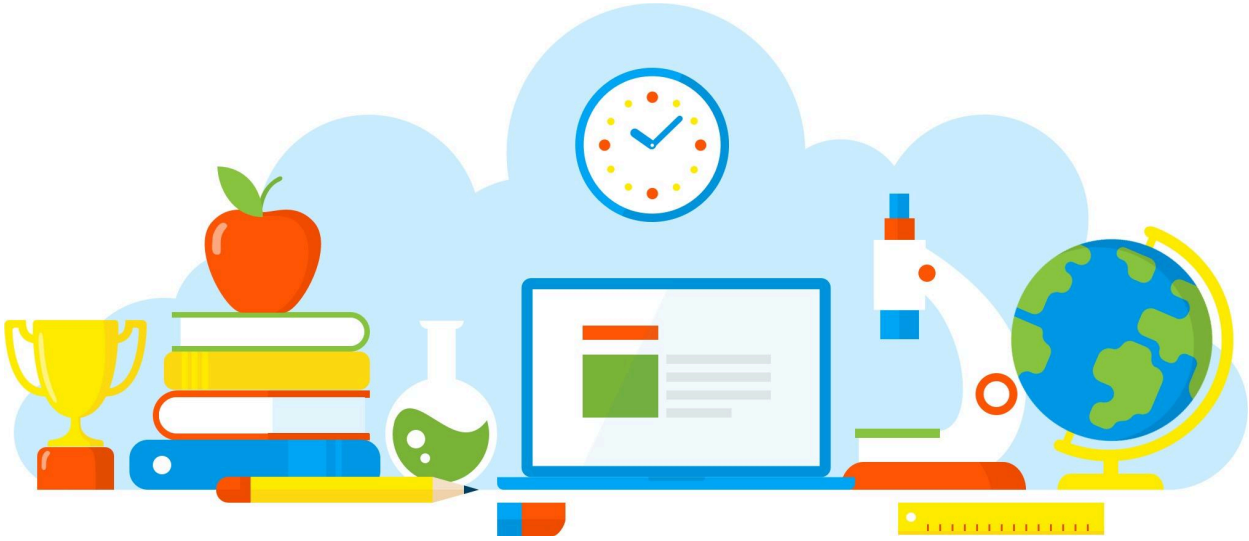
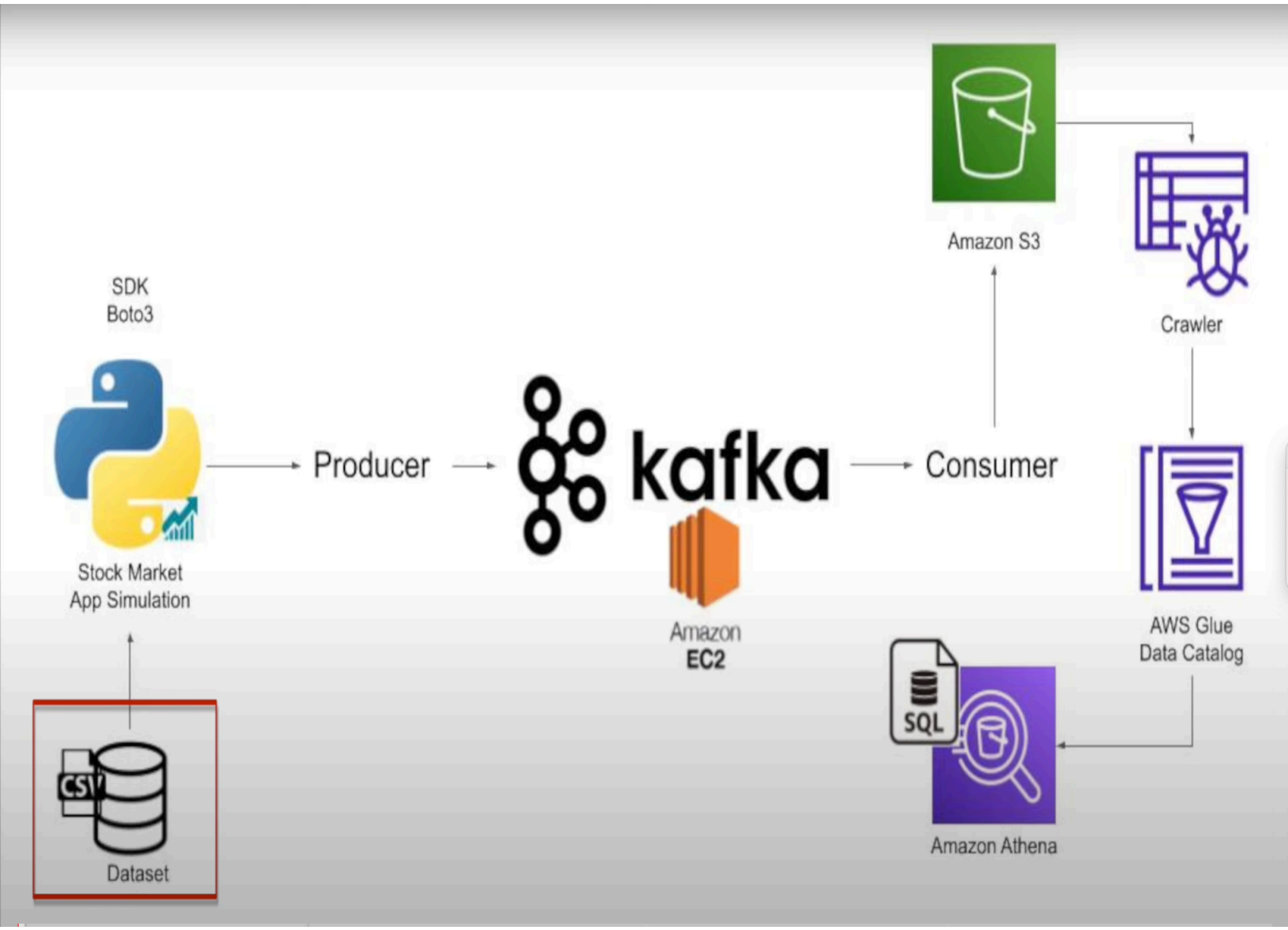


Design Document

SDE

Ayush Mishra (M22AIE251)
Abhishek Tamrakar (M22AIE228)
Mohit Chandra Saxena (M22AIE240)

Real-Time Stock Market Data Pipeline using Kafka and AWS



Streaming Stock Market Data Pipeline using

Kafka & AWS Integration

1. Overview:

This design document outlines the architecture of a real-time stock market data pipeline using Kafka and AWS services. The pipeline consists of a stock market simulation that produces data in real-time, a Kafka message broker that facilitates the data streaming, and several AWS components like S3, Glue, and Athena to store, catalog, and query the data.

1.1 Purpose:

The main objective of this pipeline is to stream stock market data from a simulated stock trading application into an Amazon S3 bucket, catalog it using AWS Glue, and enable querying and analysis using Amazon Athena.

2. Components of the System

The system is composed of the following components, each performing a critical role in the data flow:

2.1 Stock Market Simulation Application (Producer)

- **Technology**: Python SDK (Boto3)
- **Description**: The stock market simulation app generates real-time stock data, mimicking the behavior of a stock market. The data comes from a CSV dataset which contains historical or pre-generated stock prices.
- **Functionality**:
    - Reads data from the CSV dataset.
    - Sends data to the Kafka producer, simulating stock price updates in real-time.

2.2 Kafka (Message Broker)

- **Technology**: Kafka on Amazon EC2
- **Description**: Kafka acts as the intermediary between the producer (the stock market simulation) and the consumer.
    - **Producer Side**: The stock market simulation app sends data to Kafka topics.
    - **Consumer Side**: The consumer subscribes to the relevant Kafka topics to retrieve the streamed data.
    - **Hosted on**: Amazon EC2 instance(s), providing flexibility and scalability for high-throughput data streams.

2.3 Consumer

- **Description**: The consumer application retrieves the streamed data from Kafka.
    - It stores the consumed data into Amazon S3 for long-term storage and further processing.

2.4 Amazon S3 (Data Lake Storage)

- **Description**: The consumer stores all stock market data into an Amazon S3 bucket. This data is stored in its raw format (likely as CSV or JSON), ready for cataloging and querying.
- **Functionality**:
    - Stores raw stock market data coming from the Kafka consumer.
    - Acts as the central data lake for all historical stock market data.

2.5 AWS Glue (Crawler and Data Catalog)

- **AWS Glue Crawler**:
    - Scans the data stored in S3, detects its schema, and generates metadata (such as table definitions) automatically.
    - The crawler runs at regular intervals to capture updates and new data in S3.
- **AWS Glue Data Catalog**:
    - Stores metadata generated by the crawler, organizing it into a structured format (tables) that can be queried.
    - Provides a central repository to manage and search through the metadata associated with the stock data.

2.6 Amazon Athena (Data Querying)

- **Description**: Amazon Athena is a serverless query service that allows running SQL queries on the stock data stored in S3. It integrates with the Glue Data Catalog, enabling users to run SQL-like queries on the raw data stored in the S3 bucket.
- **Functionality**:
    - Performs SQL queries on stock data without the need for complex ETL pipelines.
    - Uses the metadata created by AWS Glue to run ad-hoc or scheduled queries, providing insights into the stock market data.

3. Workflow

The overall workflow of the data pipeline can be described as follows:

1. **Data Generation:** The stock market app simulates stock prices and sends real-time data from a CSV dataset to the Kafka producer.
2. **Message Broker (Kafka):**
    - Kafka, running on EC2, receives the data from the stock market app via the producer.
    - Kafka brokers forward the data to the Kafka consumer.
3. **Data Ingestion (Consumer):**
    - The consumer retrieves stock market data from Kafka and writes it to an Amazon S3 bucket in a specified format (CSV or JSON).

4. **Data Crawling (AWS Glue):**
    - AWS Glue crawler scans the S3 bucket to detect new stock market data.
    - The crawler updates the metadata in the Glue Data Catalog, organizing it for future queries.
5. **Querying and Analysis (Athena):**
    - Amazon Athena uses the Glue Data Catalog to run SQL queries on the data stored in S3, allowing users to retrieve stock data insights in real-time or from historical data.

4. Data Flow Diagram

The following flow illustrates how data moves through the pipeline:

1. **Stock Market Simulation App** (Producer) → **Kafka** (EC2 Broker) → **Consumer** → **Amazon S3**
2. **Amazon S3** → **AWS Glue Crawler** → **AWS Glue Data Catalog**
3. **AWS Glue Data Catalog** → **Amazon Athena** (for querying)

5. Technologies Used

- **Python SDK (Boto3)**: Used to interface with AWS services for the stock market simulation.
- **Apache Kafka**: Message broker for real-time data streaming, hosted on Amazon EC2.
- **Amazon EC2**: Host for Kafka services.
- **Amazon S3**: Data lake to store the streamed stock market data.
- **AWS Glue**: For automatic metadata generation and data cataloging.
- **Amazon Athena**: For querying and analyzing data using SQL.

6. Advantages

- **Real-time Data Streaming**: The use of Kafka ensures that stock data is streamed in real-time with minimal latency.
- **Scalable**: AWS services like EC2, S3, Glue, and Athena are highly scalable, ensuring that the system can handle large amounts of stock data.
- **Serverless Querying**: Amazon Athena provides a cost-effective, serverless way to run queries on the data without needing a dedicated database infrastructure.
- **Automated Data Management**: AWS Glue automatically crawls and catalogs the data, removing the need for manual schema management.

7. Use Cases

- **Real-time Stock Market Monitoring**: This pipeline can be used by stock analysts or traders who need real-time data to make decisions.
- **Historical Data Analysis**: The stored data can be queried using Athena to perform backtesting, forecasting, or other analytical tasks.

8. Potential Enhancements

- **Data Transformation with AWS Glue ETL**: Future versions of the pipeline could incorporate AWS Glue ETL jobs to preprocess the data before it's queried by Athena.
- **Real-Time Alerts**: Integration with Lambda or SNS to trigger real-time alerts based on specific stock market conditions detected in the data stream.