High-Level Design Document (HLD)

Project Name: Real-Time Data Streaming Pipeline – Kafka - NiFi -HDFS-Airflow-Hive
Prepared By: Marwa Mansour
Date:11-Feb-2026

Project Overview

This project implements a real-time data streaming pipeline integrated with a modern Data Lake architecture.
User events are generated using a Java-based Kafka Producer and serialized using Avro schema. The events are streamed into Apache Kafka.
Apache NiFi consumes the streaming data, transforms it into Parquet format, and stores it in Hadoop HDFS.
Apache Airflow orchestrates the batch processing layer by scheduling data loading jobs, managing file archiving, and ensuring reliable pipeline execution.
Finally, the data is made available for analytical querying through Apache Hive.
The solution ensures:

- Reliable real-time streaming
- Centralized workflow orchestration
- Schema validation and evolution support
- Optimized columnar storage (Parquet + Snappy)
- Analytics-ready structured datasets

---

2. Business Logic (High-Level)

2.1 Real-Time Event Generation

- Generate 1,000 structured UserEvent records.
- Events are produced in real time using a Java Kafka Producer.

2.2 Serialization & Schema Governance

- Messages are serialized using Avro format.
- Schema is registered and validated using Confluent Schema Registry.
- Schema compatibility is enforced for future evolution.

2.3 Kafka Streaming Layer

- Events are published to Kafka topic: user-events.
- Partitioning and offset management are handled automatically.
- Provides scalable and fault-tolerant streaming.

2.4 Stream Processing (NiFi)

- Messages are consumed using ConsumeKafkaRecord_2_6.
- Data is validated and transformed.
- Records are written to HDFS in Parquet format.

2.5 Orchestration Layer (Airflow)

- Schedule and manage pipeline execution.
- Load only new files into Hive managed table.
- Move processed files to archive directory.
- Handle retries, logging, and monitoring.

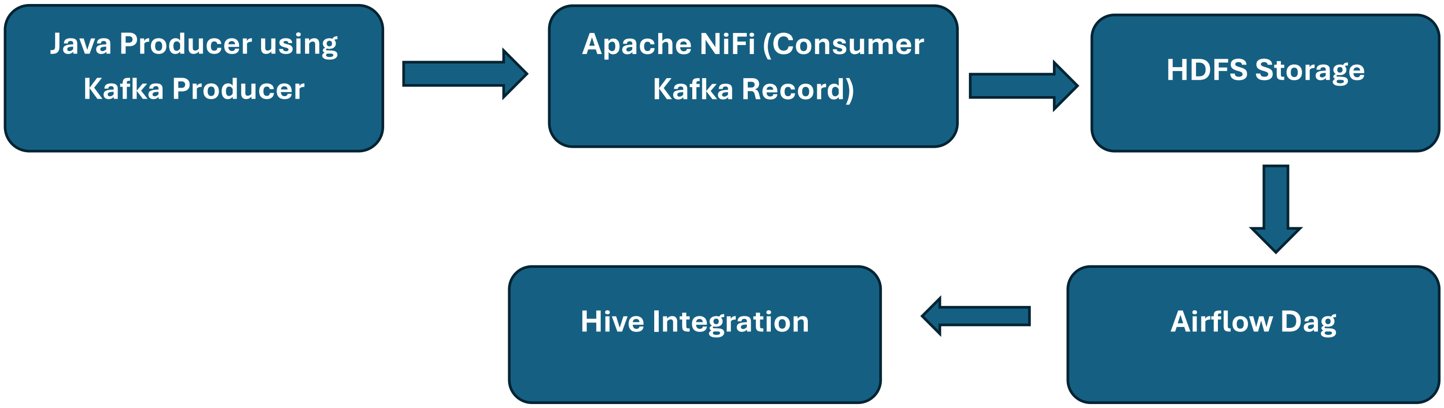
2.6 Storage & Analytics Layer

- Data is stored in HDFS (Raw Layer).
- Airflow loads processed data into Hive Managed Table.
- Hive enables structured querying for analytics and BI reporting.

2.7 Supporting Technologies

- **Parquet** → Columnar storage format
- **Snappy** → Compression codec
- **Bash** → HDFS file operations

3. Data Flow


Java Producer using Kafka Producer
Apache NiFi (Consumer Kafka Record)
HDFS Storage
Airflow Dag
Hive Integration

4. Tools Used

Tool	Purpose
Apache Kafka	Distributed real-time streaming platform
Apache NiFi	Data flow orchestration & monitoring
HDFS	Store processed data in HDFS in Parquet format
Airflow	Schedules and manages pipeline execution, loads new data into Hive,
Hive	Create Hive table for Parquet files in HDFS and store data

5. Expected Data Volume

- Per 1000 Records: ~200–300 KB
- Daily (single execution): ~300 KB
- Monthly: ~9 MB
- Yearly: ~108 MB

6. Future Enhancements

- Add stream analytics layer
- Integrate with BI tools (Power BI / Superset)