Data Source

Source 1: PageView (stream datagen) (topic1_PageView)

This data source provides information about user interactions with a webpage. It helps us understand user behavior on the website, such as which is the most common source that directs users to the current page.
The data is generated by **./producer/02_kafka_producer.py** using confluent_kafka to produce the data into the Kafka topic.

session_id: Unique identifier for user session.
user_id: Identifier for the user who triggered the view event.
source: The origin of the view event. (Homepage, Inbox, External)
event_timestamp: The timestamp when the view event occurred.

Source 2: UserSession (stream datagen) (topic2_UserSession)

This data source logs the login time and session creation for each user.
It's used for tracking user activity and understanding user engagement with the platform.
The data is generated by **./producer/02_kafka_producer.py** using confluent_kafka to produce the data into the Kafka topic.

session_id: Unique identifier for user session.
user_id: Unique identifier for each user.
user_name: The name of the user.
login_timestamp: The timestamp when the user last logged in.
device: The device used by the user for the last login.

Source 3: Users (relational database) (topic3_Users)

This data source stores demographic information about the users. It allows us to understand the user base's composition and can be used to join with the data streams.
It is generated by **./producer/01_database_producer.py** using pymysql to connect to the MySQL instance, then fed into the Kafka topic using jdbc-source-connector.

user_id: Unique identifier for each user.
age: The age of the user.
gender: The gender of the user.
country: The primary location of the user.
subscription: The subscription tier of the user (Platinum, Gold, Silver, Standard).

Kafka System & KSQL DB

Kafka and KSQL DB are used to manage and process the data streams from the sources. Kafka is a distributed streaming platform that allows for real-time data processing and transfer. KSQL DB, on the other hand, is an event streaming database for Apache Kafka that enables stream processing tasks using SQL-like syntax.

In this system, we have three Kafka brokers (kafka1, kafka2, kafka3) configured with 5 partitions and a replication-factor of 3.

The first three topics (topic1, topic2, topic3) are used to create STREAMS and TABLES in KSQL DB. The KSQL commands are executed by **./ksql/01_create_stream.py** using the Python subprocess to send the commands in the KSQL docker environment.

topic1: PageView - Base Stream
topic2: UserSession - Base Stream
topic3: Users - Base Table

The topic4 and topic5 are the cleaned users' table joined with the view events.
The output is the average age and user count by country.

topic4: CleanedUsers - Users table without NULL values
topic5: PageView Country - PageView joined with CleanedUsers

Topics 6-8 use different window functions to get the real-time user count based on various page sources.

topic6: PageView Tumbling - Tumbling Windows
topic7: PageView Hopping - Hopping Windows
topic8: PageView Session - Session Windows

The sample of the output can be obtained by **./ksql/02_testing_correctness.py**, which will output a few rows from each tables.

Apache Pinot & Streamlit

Apache Pinot and Streamlit are used for real-time analytics and interactive visualization of the data streams.

Apache Pinot

Apache Pinot is a real-time distributed OLAP datastore, which is used to deliver scalable real-time analytics with low latency. It can ingest data from batch and stream data sources (such as Kafka).

In this system, real-time tables in Pinot are created based on topic5 to topic8.
The Pinot table configurations and schemas are uploaded using REST API calls:
curl -X POST -H "Content-Type: application/json" -d @./config/pinot-schema-5.json http://localhost:9000/schemas
curl -X POST -H "Content-Type: application/json" -d @./config/pinot-config-5.json http://localhost:9000/tables

The configuration and schema files (pinot-config-5.json, pinot-schema-5.json, and so on for 6-8) define the table structure and settings in Pinot.

The created real-time tables are:

- pageview_country_REALTIME
- pageview_hopping_REALTIME
- pageview_session_REALTIME
- pageview_tumbling_REALTIME

Streamlit

Streamlit is an open-source Python library that makes it easy to create custom web apps for machine learning and data science.

In this system, Streamlit is used to query the real-time tables in Pinot and plot the charts for each table. The charts provide visual insights into user count by country and user count by each window (grouped by source). The real-time data will be fetched every 15 seconds.