Design Choices and Performance Considerations

1. Table Design Choices

Raw Tables

- **raw_user_logs** is partitioned by year, month, day to enable efficient filtering and querying of data by date. This reduces the amount of data scanned during queries.
- **raw_content_metadata** is stored as a flat external table because metadata changes infrequently and does not require partitioning.

Star Schema Tables

- **dim_content** is stored in **Parquet format**, which provides better compression and faster read performance compared to CSV or TEXTFILE.
- **fact_user_actions** is also stored in **Parquet format** and is **partitioned by year, month, day**, enabling optimized queries for date-based filtering.

2. Performance Considerations

- **Partitioning** helps reduce query execution time by scanning only the necessary partitions instead of the whole dataset.
- **Using Parquet** for storage improves read performance due to columnar storage, compression, and better data encoding.
- **Setting hive.exec.dynamic.partition=true** allows efficient partitioning during inserts without explicitly defining static values.

3. Execution Time for the Whole Pipeline

- **Data Ingestion (HDFS Upload)**: The time to upload CSVs into HDFS depends on network speed and file sizes (typically **1-2 minutes** for small datasets).
- **Creating and Populating dim_content**: Since this involves a simple transformation from raw_content_metadata, it runs quickly (**~1-3 seconds**).
- **Inserting into fact_user_actions**: Since this step involves **partitioning and timestamp conversion**, it takes longer (usually **~1-3 minutes**, depending on data volume and cluster performance).
- **Execution of sample queries took ~ 5**-10 seconds