Multiple Kafka Brokers, multiple topics, different types of data
MySQL -> Kafka
DynamoDB -> Kafka
PB scale
Dump the Lakehouse

SLA:
Transaction: 1hr
Clickstream: 6hrs (topic to topic)

Analytics, ds, BE,
Looker, DB, notebooks, SQL warehouse
(RBAC)

RPS: 3k RPS -> 64.8M records (618GB)
Transactional: 500 QPS
Max records: 2TB (1B records)
Time: 5yrs
—----------

Clickstream and Transactional Topics:
—----------

High events data
Low events data

Tech Stack:

Kafka
Connectors/ Service
AWS
S3 (Datalake)
Iceberg (Lakehouse)
Microservices
Glue (Catalog)

AllEvents data (T-1D) -> JSON to parquet format -> Spark Job (EMR) -> Athena tables
Each event topic

Spark Job:

Kafka topics (JSON) -> Flink Job -> S3 buckets (JSON) (eventTime) -> Spark Job (JSON -> parquet) -> S3 buckets -> Athena/ iceberg tables

Ingestion:
Schema evolution, transformations (user-specific), partitioning, mode (append only, upsert)

Requirements
- Add/Remove Columns and change data type
- Encrypt data, extract field information
- append/upsert
- Testing framework
- Add Failure scenarios (incompatible data type)
- Flink job
- S3 (localstack)
- Scalable, extend transformation Framework (custom built in functions)
- Create DDL using YAML
-

Kafka -> Flink job -> Iceberg tables

Event records in cache which will have (eventName, Schema)
Flink Job:
- Validating the schema for whitelisted events
- Process the data (windowing, aggregations, define watermarks)
- Dumps data into the Kafka topic
-

YAML Config:
- ingestionUnit
    - Kafka
        - Topic
        - Retention
        - incomingData
        - numberOfConsumers
        - Brokers
        - R53
  - outsourcingUnit
    - Table
        - Database
        - TableName
        - S3 Location
        - Format
        - Mode (Upsert/Append/TruncateLoad)
        - id Columns

- Timestamp Columns
- PIIColumns
- Jsoncolumn
-

Schema Registry:
schemas for all kafka topics

Fetch the schema of source topics

DDL of the destination table

Validate and evolve the schema of the destination
- Change in data Type
- Additon/Removal of columns

Dump data based on Mode in yaml

Storage level optimizations:
- Compressions:
- Size of columns
- Compaction, Removal of orphan files

Read heavy - Write Heavy

- COW/MOR
-