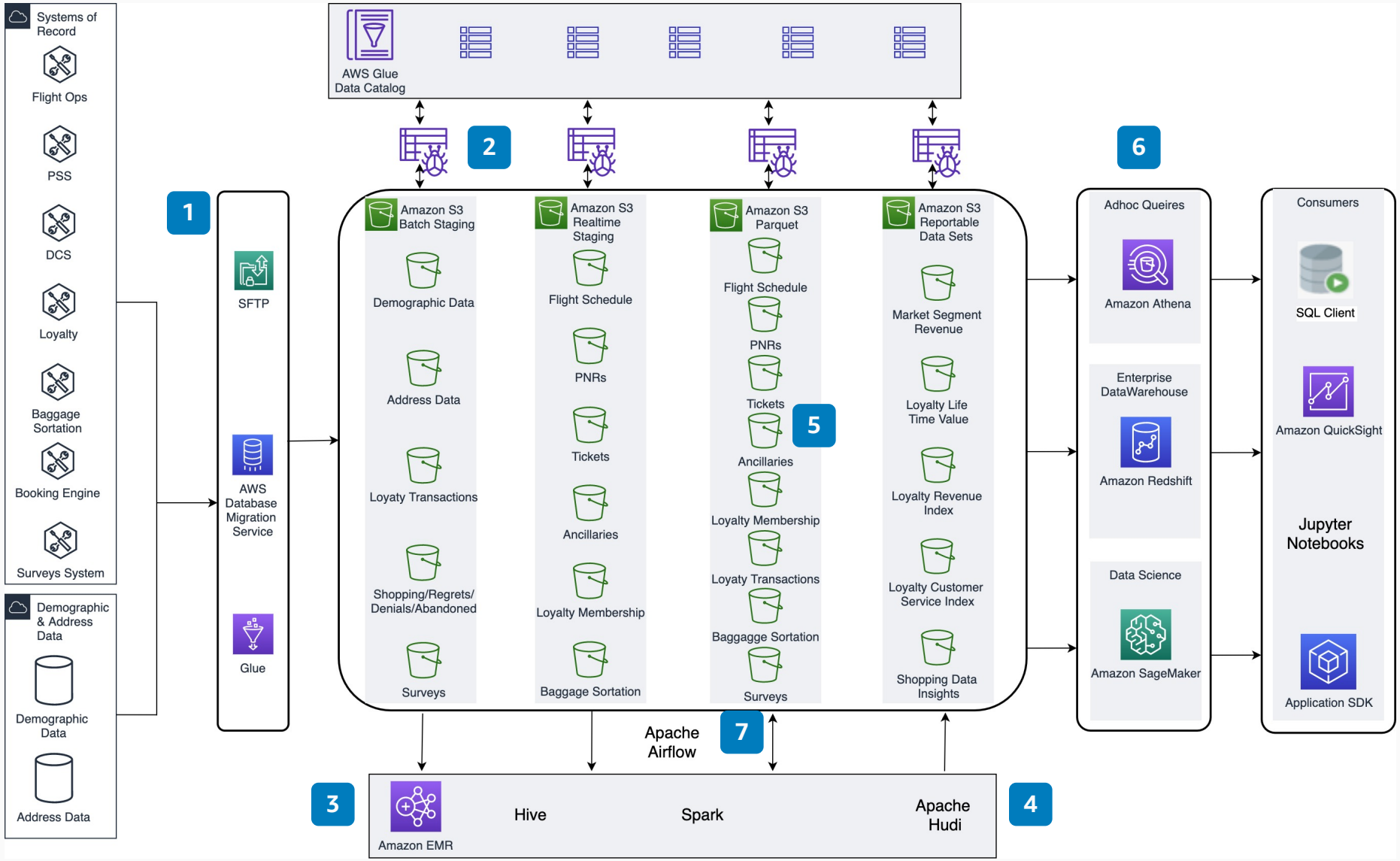
Managing Inserts and Upserts in a Serverless Data Lake

Leverage Apache Hudi running on Amazon EMR to process inserts and updates to data sets in Amazon S3 to build a cost effective and scalable data lake. This enables the provisioning of on-demand analytics, helping data scientists, and creating persistent data marts as necessary to help manage business agility with a lower total cost of ownership.



1. Data is ingested from the source systems using either batch, CDC, Streaming, etc. into RAW layer in **Amazon S3.**
2. Once the data persisted in the RAW data lake on S3, the data will be crawled and will be populated in the **AWS Glue Data Catalog** using Crawler.
3. The RAW data is pulled into an **Amazon EMR** cluster and read using Hive and Spark, for cleaning and transformation.
4. Apache Hudi running on **Amazon EMR**, will read the data using Spark APIs and perform inserts and upserts on the required data sets.
5. The cleaned and transformed data is persisted back into the **Amazon S3** processed and reportable buckets.
6. The reportable data is consumed on demand using **Amazon Athena** or loaded into **Amazon Redshift**, and can be consumed by different users, tools and resources.
7. The complete data movement, spinning on-demand **Amazon EMR** clusters (in case of batch data) and loading the data is handled by a workflow orchestration using **Amazon Managed Workflows for Apache Airflow** (MWAA).

AWS Reference Architecture