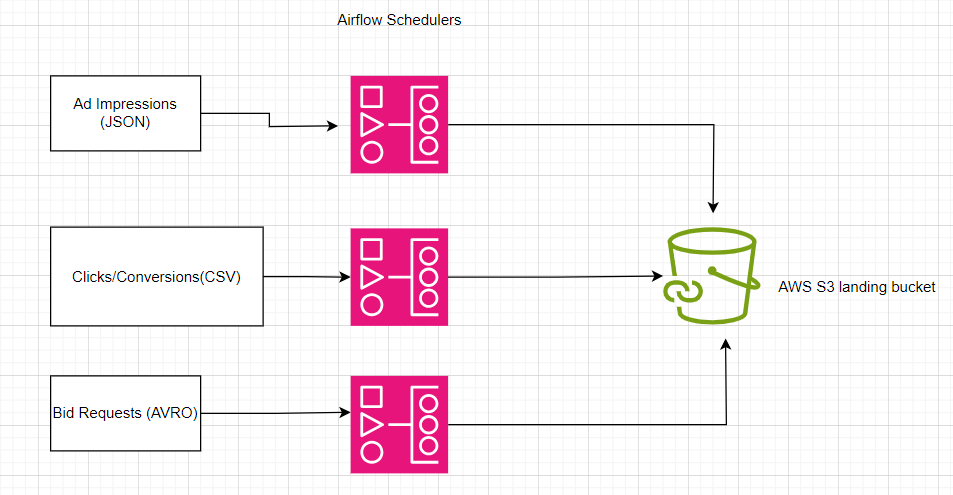
Data Ingestion

For the data ingestion system which is scalable and can handle a wide variety of data types, we would go with AWS S3 buckets as our data lake store.

The reason why this cloud vendor is picked is due to the generally good performance associated with S3 buckets (11 9's of availability).

One another reason is the general cost effectiveness associated with AWS s3 solutions. There are many storage tiers available like standard, infrequent access, glacier to name a few.

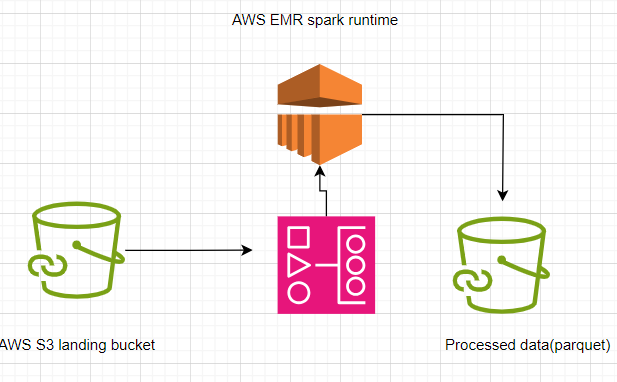


Data Processing

The raw data after landing to the s3 bucket can now be processed by AWS EMR clusters

Having Pyspark runtime. It is very useful for performing Transformation/Processing for huge datasets. The scalability and fault tolerance of the EMR service is guaranteed by AWS.

We will extract the data in the three formats and place them in a separate curated bucket.After being processed by Pyspark(EMR), the consistent data would be placed in parquet format using snappy compression.



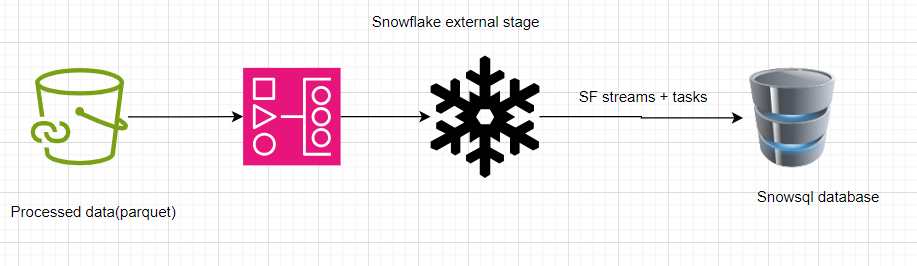
Querying and storage of the Data

The processed data, which resides in our curated bucket would then be loaded onto a snowflake temporary table using the concept of data loading via external stages. This step would be triggerred by an airflow job using cron scheduler.

The tabular data would then be cleaned to remove all irrelevant records(null values,inconsistent values etc).

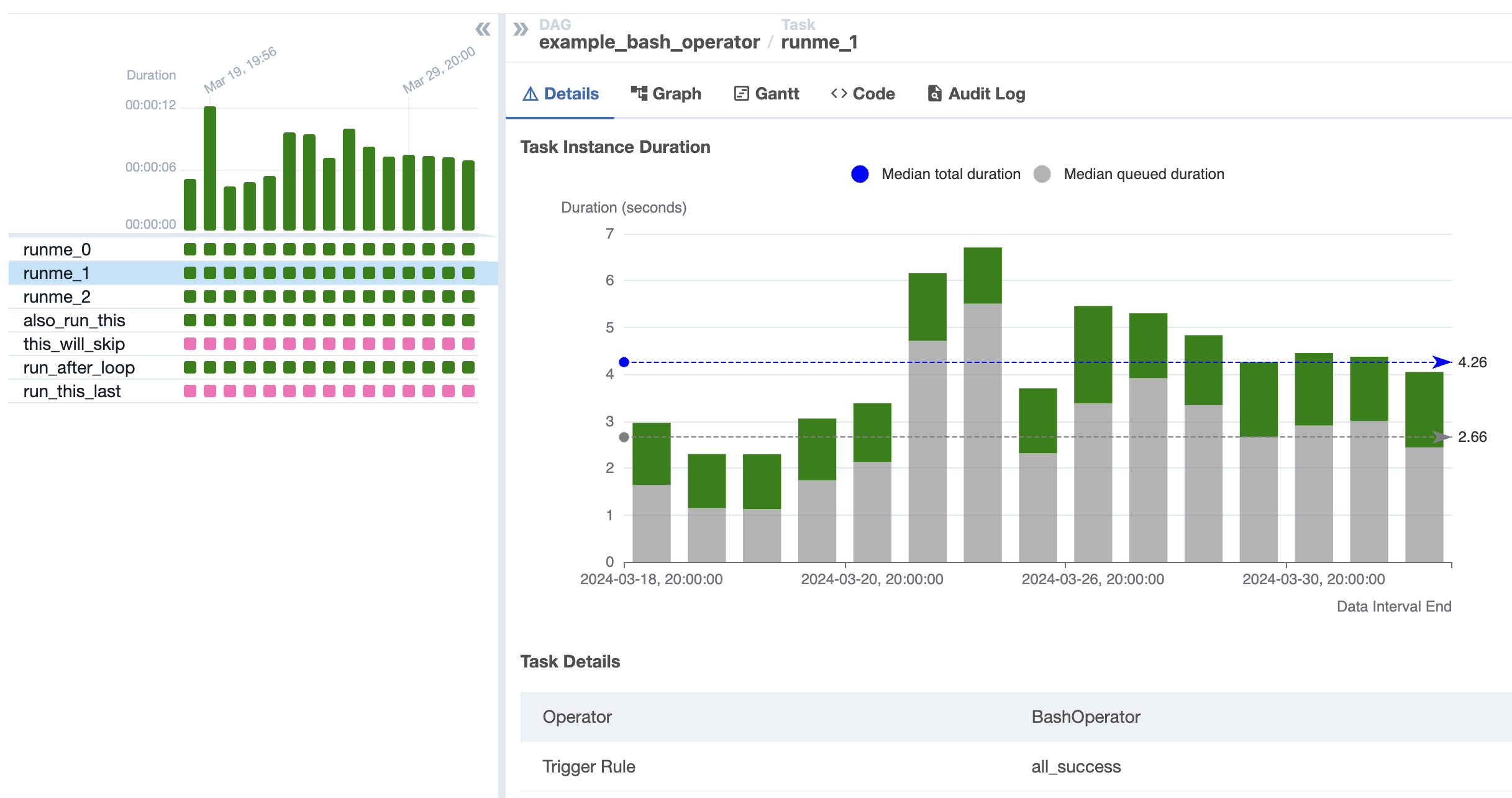
The cleaned data can then be moved to our cleaned tables. The data would be moved to the clean layer using a combination of Snowflake Tasks and Streams objects.

The cleaned tables would be further used to create materialized views and they can be shared to the downstream BI systems for analysis. This makes up the consumption layer for our data pipeline.



Handling Errors and Notification

The entire pipeline is orchestrated through Apache Airflow. The tool provides a very intuitive interface for viewing our job/task runs using a graph view.We can read through the task status using several logs which are created in the logs folder.



In addition, we have the AWS cloudwatch logs to troubleshoot any errors/exceptions during Spark processing.