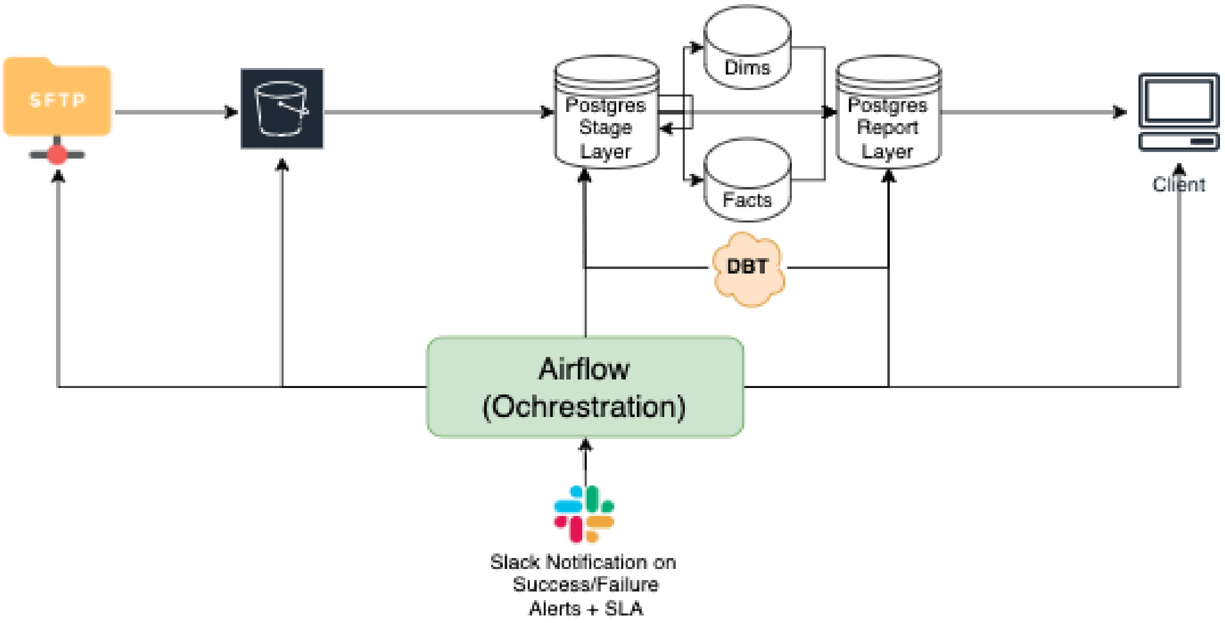

Using **Airflow** is one of the options**:**

- We write an Airflow DAG to pull data from SFTP to an S3 Bucket/ADLS Container.
- We can schedule DAG to copy from data from Storage location to Database.
- We can have Airflow DAG's which can have both SQL and Python to apply business rules/transformations in Tasks in DAG and write results into Reporting Database.
- Another DAG can query Reporting DB and generate results and copy to Client Server
- A QA DAG which generates metrics and writes to QA Database.
- Slack can be integrated in all DAG’s.

Note: For transformations, we can use Rivery/DBT, and for databases we can also use Cloud Data Warehouse like Redshift/Snowflake(Semi Structured data too), but I am assuming the data is not huge and complex.

- Inclined to openSource & cost effectiveness.
- Retention Period can be configured on S3(1 year) and Postgres to have latest data(1 week)

1. Given the lack of historical institutional knowledge of the system how would you approach a complex task to gather requirements from end users?

- Data Profiling to understand the structure and quality of the existing data
- Understand the Data and End User Access Pattern
- Check Existing Documentation, Design docs to get an idea
- Stakeholder meetings including end-users, analysts, and decision-makers to understand pain points, challenges and critical features.
- Gather Data Patterns,Anomalies,Compliance requirements, SLA's etc.

2. What are some important factors that you would consider when selecting which product would replace their Vertica architecture?

- Scalability, Concurrency and Performance
- Cost effectiveness
- Ease of Migration
- Data Security and compliance
- Integration with the existing ecosystem.
- Ease of development and integration with third party tools and Softwares.
- Documentation and customer/community support

3. It is critical the existing system remains online during the entire process (a restart can take up to 12 hours to complete). How would you ensure a smooth cut over experience with minimal downtime?

- Make sure that data is replicated in the new system and it’s up to date before the cutoff.
- Update the data in parallel in the new system and old system. Come with a process that updates both systems concurrently.
- Come up with a Roll black plan incase if the new system fails for any unknown foreseen error.
- Communicate the Migration plan to all stakeholders and keep them informed on every action.
- Continuously monitor data consistency & performance.Collect Key Metrics.
- Collect continuous feedback from your end customers.