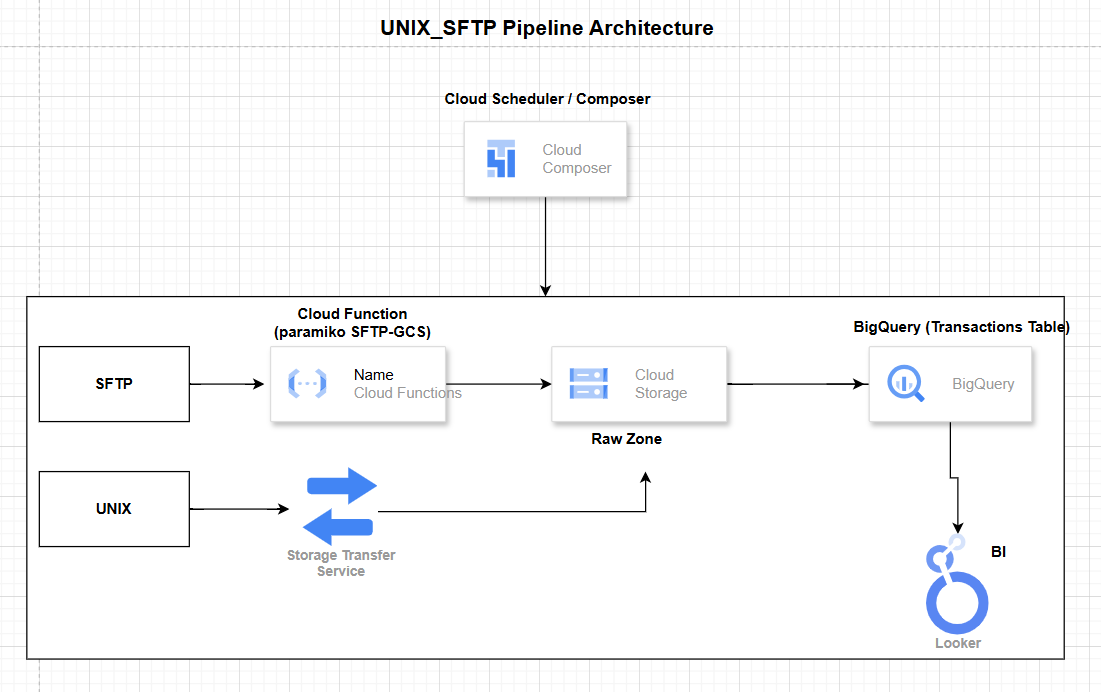
**Architecture**



* **Extract Files from SFTP Servers using cloud function**   
  Securely connect to one or more SFTP servers to retrieve daily data files from on-premise or external partners using the cloud function
* **Extract Files from UNIX Servers using storge transfer service**

Since we're trying to move a 25 GB file from a Unix server, using **Storage Transfer Service (STS)** is a suitable option. It’s designed for handling large-scale data transfers efficiently and securely."

* **Upload to Cloud Storage (GCS)**  
  Store each file in a designated **Cloud Storage bucket** using a logical folder structure (/raw/sftp/YYYYMMDD/) to support data traceability and lifecycle management.
* **Expose Files as External Tables in BigQuery**  
  Register each file or folder in GCS as an **external table** in BigQuery for immediate querying, without requiring ingestion or duplication.
* **BigQuery External Table + MERGE**

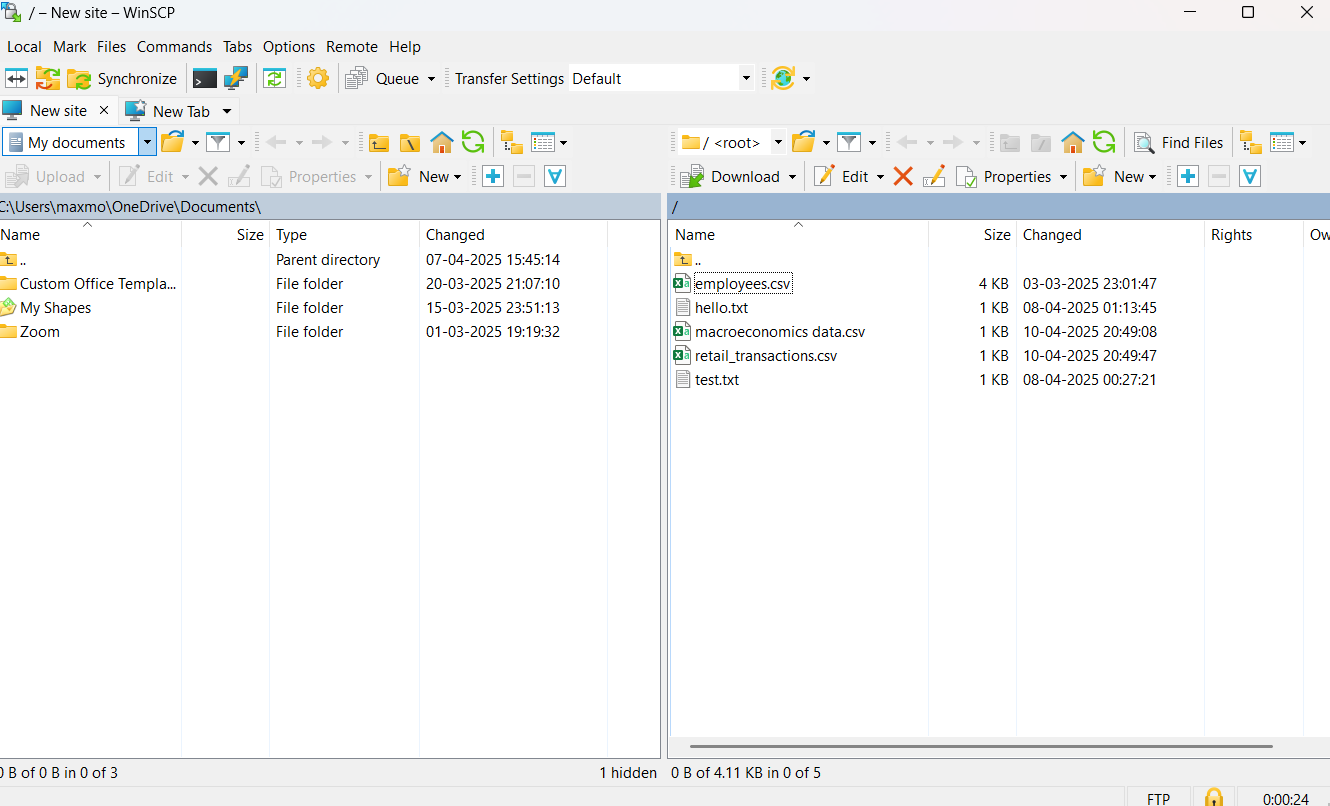
Store the data in a BigQuery external table, then use a scheduled query to transform and load the data into the master table

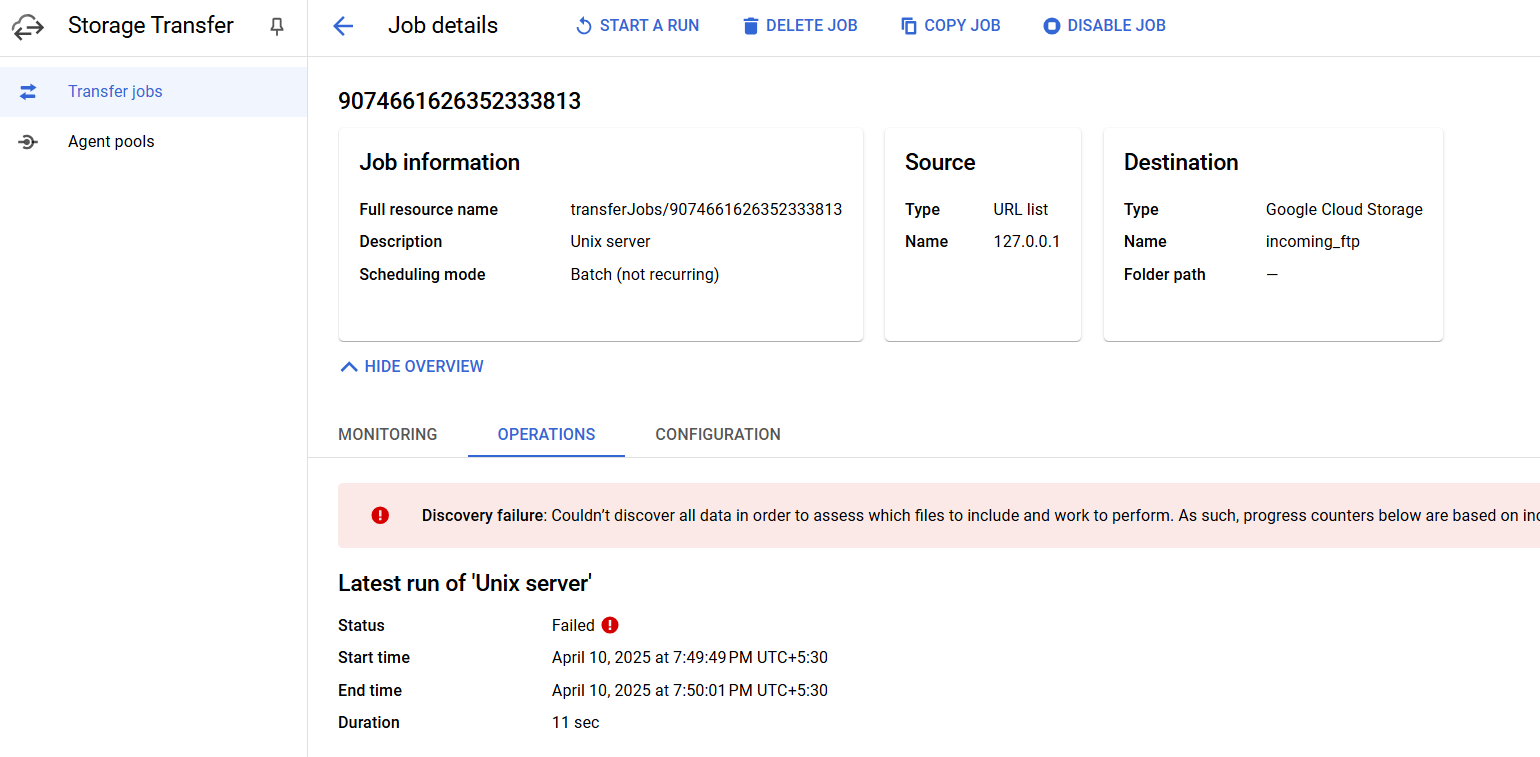
* **Orchestrate with Cloud Composer (Airflow)**  
  Use **Cloud Composer** to schedule and manage the end-to-end pipeline — including:

Unix server files / SFTP :-

**Get file from Unix server: -**

Create the job as outlined below to retrieve data from the Unix server.  
**Note:** The job is currently failing due to the use of localhost (127.0.0.1) as the host and some authentication issues."

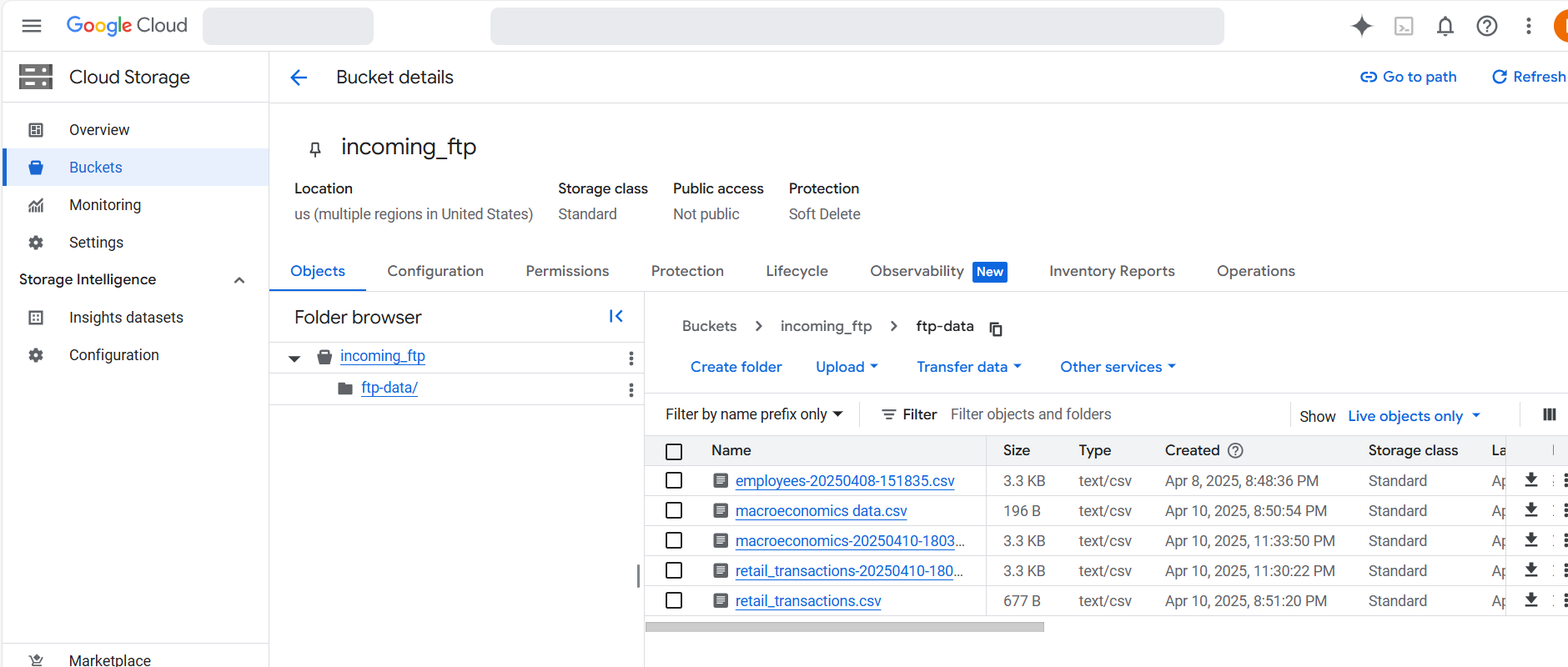




**Using the cloud function to get the SFTP file: -**

To get a file from SFTP using a Cloud Function, here's a basic working example using Python and paramiko (an SSH library).

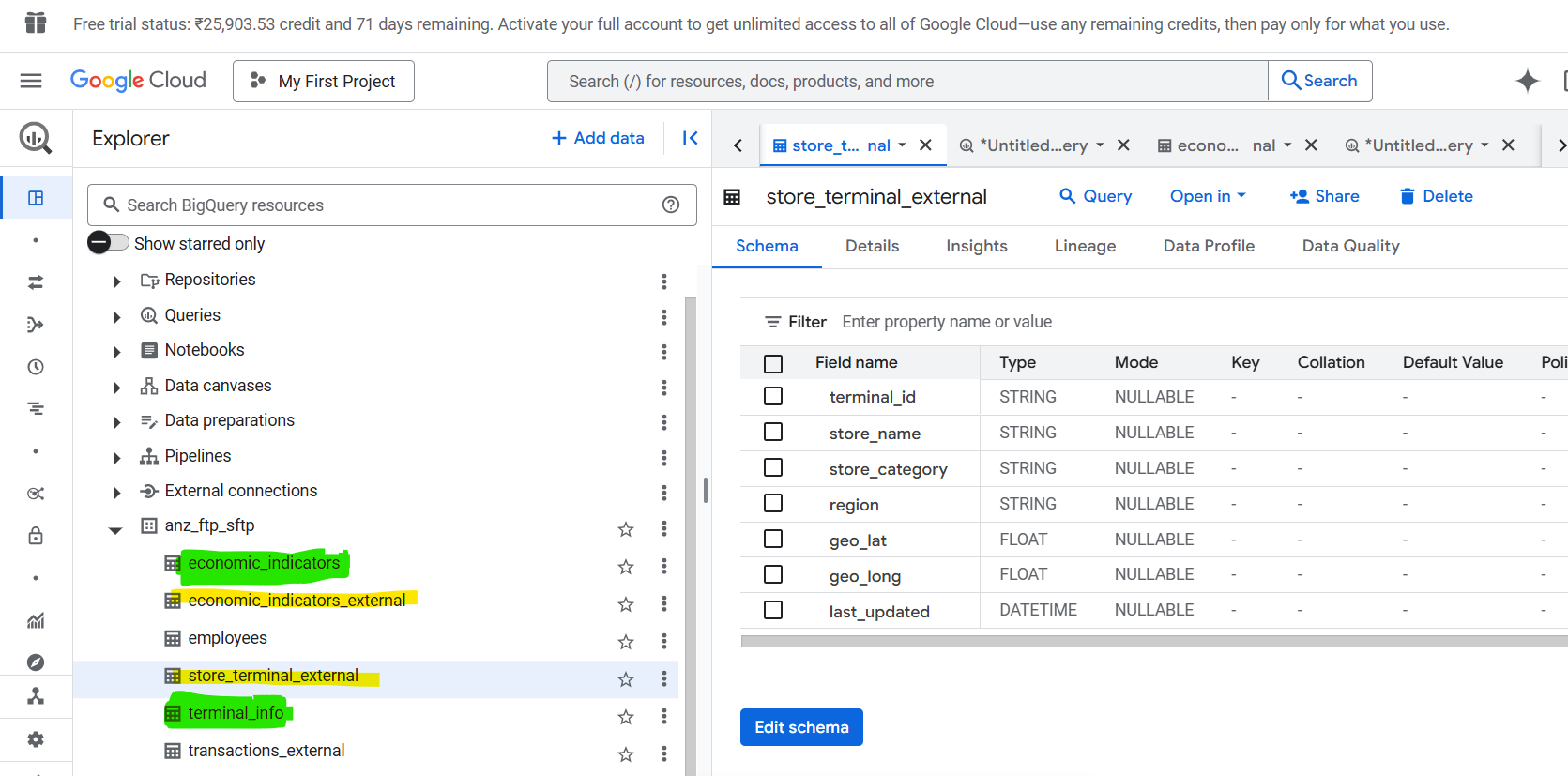
This Cloud Function connects to an SFTP server, downloads a file, and saves it to Google Cloud Storage (GCS).



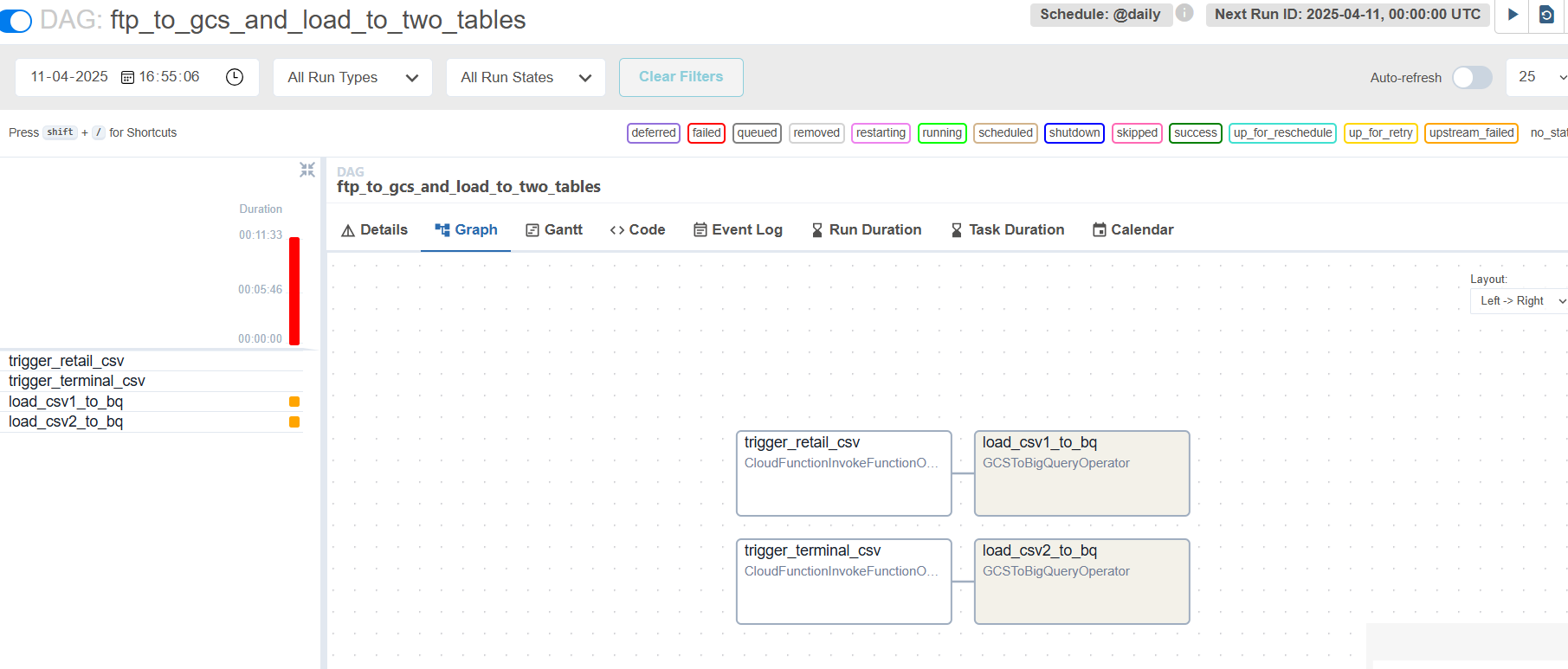
**GCS to big query to move the file to external table in BQ;-**

An external table in BigQuery will reference the source file in GCS. Using scheduled queries,

we'll execute the ELT workflow to transform and load the data into the target master tables in BigQuery



**Orchestrate with Cloud Composer: -**

****

orchestrating a data ingestion and processing pipeline using **Cloud Composer (Apache Airflow)**. The end-to-end workflow involves the following components:

1. **Cloud Function (SFTP Extraction)**
   * A Cloud Function is triggered to securely connect to an SFTP server.
   * It pulls data files and uploads them to Google Cloud Storage (GCS).
2. **Storage Transfer Service (Unix Server to GCS)**
   * For files hosted on a Unix server accessible via SFTP/HTTP, we configure a Storage Transfer Service (STS) job.
   * This job periodically pulls files from the Unix server into GCS, enabling reliable synchronization.
3. **GCS to BigQuery (External Table + ELT)**
   * Files in GCS are referenced via external tables in BigQuery, allowing immediate querying without data loading.
   * Scheduled Queries or downstream tasks in the DAG perform ELT transformations, loading processed data into BigQuery master tables.
4. **Cloud Composer DAG**
   * A Cloud Composer DAG orchestrates all the above steps, ensuring proper execution order, retries, and logging.