Building
Modern Data Stacks

Project Brief

Adventure Works

Adventure works is a bicycle manufacturing company. This project demonstrated how to build data pipelines for an e-commerce, implement machine learning models, and develop business intelligence reporting solutions.

BRIAN GWAYI

Building Modern Data Stacks

Foundation First !!!

Four Key Questions

I. Where do we consolidate our data ? > Storage
II. How will we get it there ? > Ingestion
III. How will we clean it up? > Transformation
IV. How will we analyze it? > Reporting

BRIAN GWAYI

The
Big Choice

Data Stack

Popular Options

Storage > Snowflake, BigQuery, s3, Redshift
Ingestion > Airbyte, Airflow, Fivetran
Transformation > dbt
Reporting > Tableau, Power BI, Looker, Superset

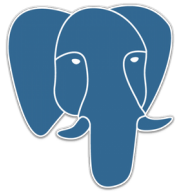
N/B This is not an exhaustive list.

BRIAN GWAYI

Data Stack Architecture Design

Source

PostgreSQL



Ingestion

Airflow

+

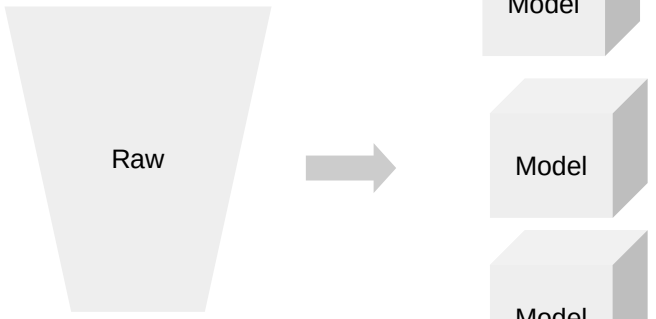
Python

Storage
BigQuery

Amazon **Redshift**

Transformation
dbt

dbt



Reporting

Looker

looker

End Goal

Put data to use

“Data is like garbage. You’d better know what you are going to do with it before you collect it.”

~ Mark Twain

PROJECTS

01

Storage/Database/Data Warehouse
Google BigQuery
Snowflake
AWS Redshift

02

Ingestion
Apache Airflow
Airbyte
Dagster

03

Transformation
Setting up dbt
Building Models

04

Reporting
Looker
Tableau
Power BI

02

INGESTION

Setting up Apache Airflow - Documentation

Phase I: Development

-Writing python scripts

```
# importing libraries

from airflow.decorators import dag, task
from datetime import datetime, timedelta
import requests
from google.cloud import bigquery
import pandas as pd
import psycopg2
from io import StringIO
```

02

INGESTION

Setting up Apache Airflow

Defining a DAG - Directed Acyclic Graph

```
args{
    ”owner”:”gwayi”,
    "retries": 1,
    "retry_delay":timedelta(minutes=5)
    }


@dag(
    default_arguments = args
    schedule=timedelta(minutes=30),
    start_date=datetime(2024, 7, 29),
    catchup=False,
    tags=['Team B']
    )
```

02

INGESTION

Setting up Apache Airflow

Extract Task Group – Source PostgreSQL Database

```
@task()
def extract():
  try:
     src_cursor.execute(sql)
     tables = cursor.fetchall()

     output = {}

     for table in tables:
       cursor.execute(f"SELECT *
              FROM {table[0]}")
rows = cursor.fetchall()

output.update({table[0]: rows})
       return output

    except Exception as e:
        print("extract error:" +
str(e))

    finally:
        connection.close()
output = extract()
```

02

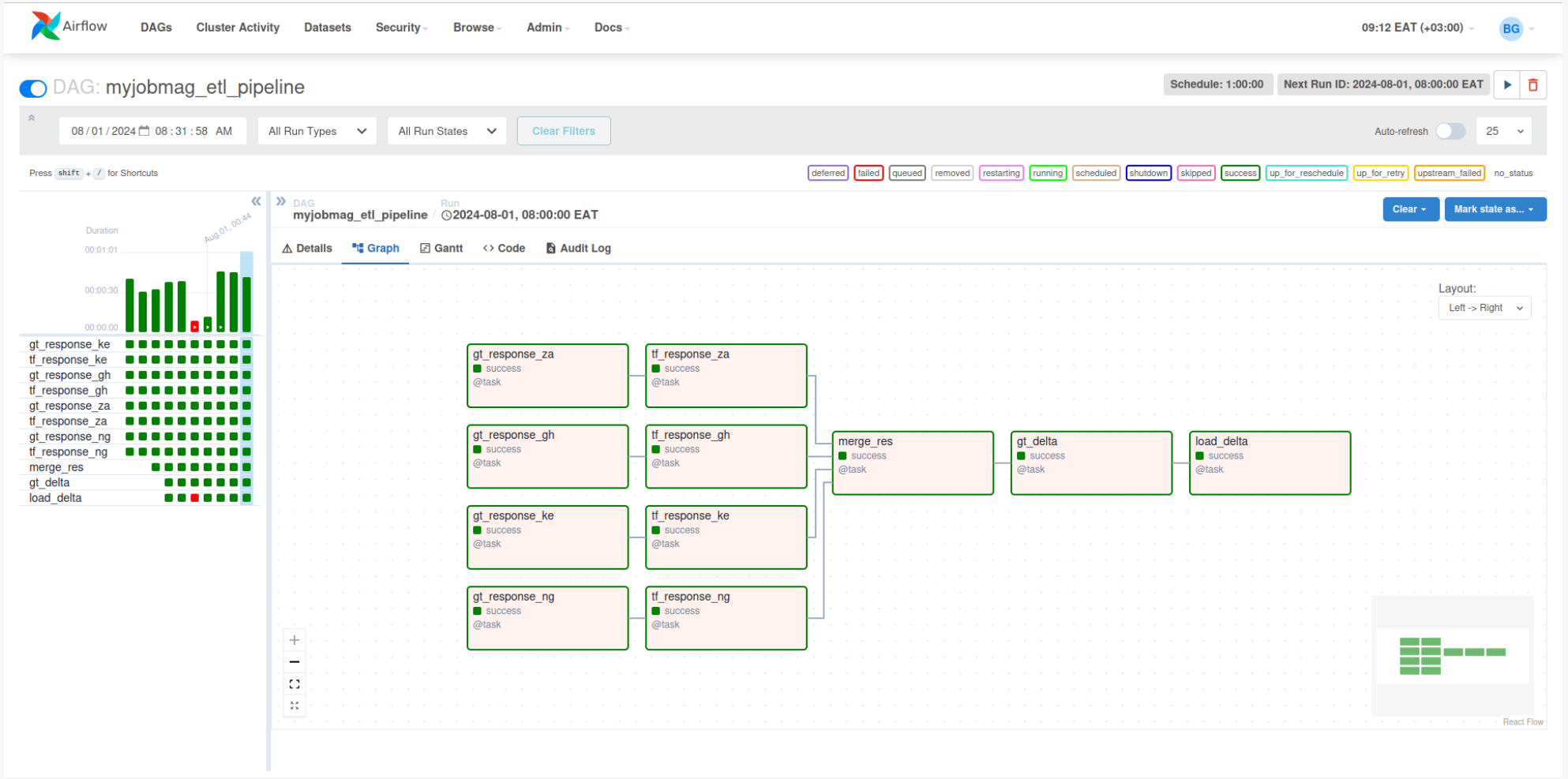
INGESTION

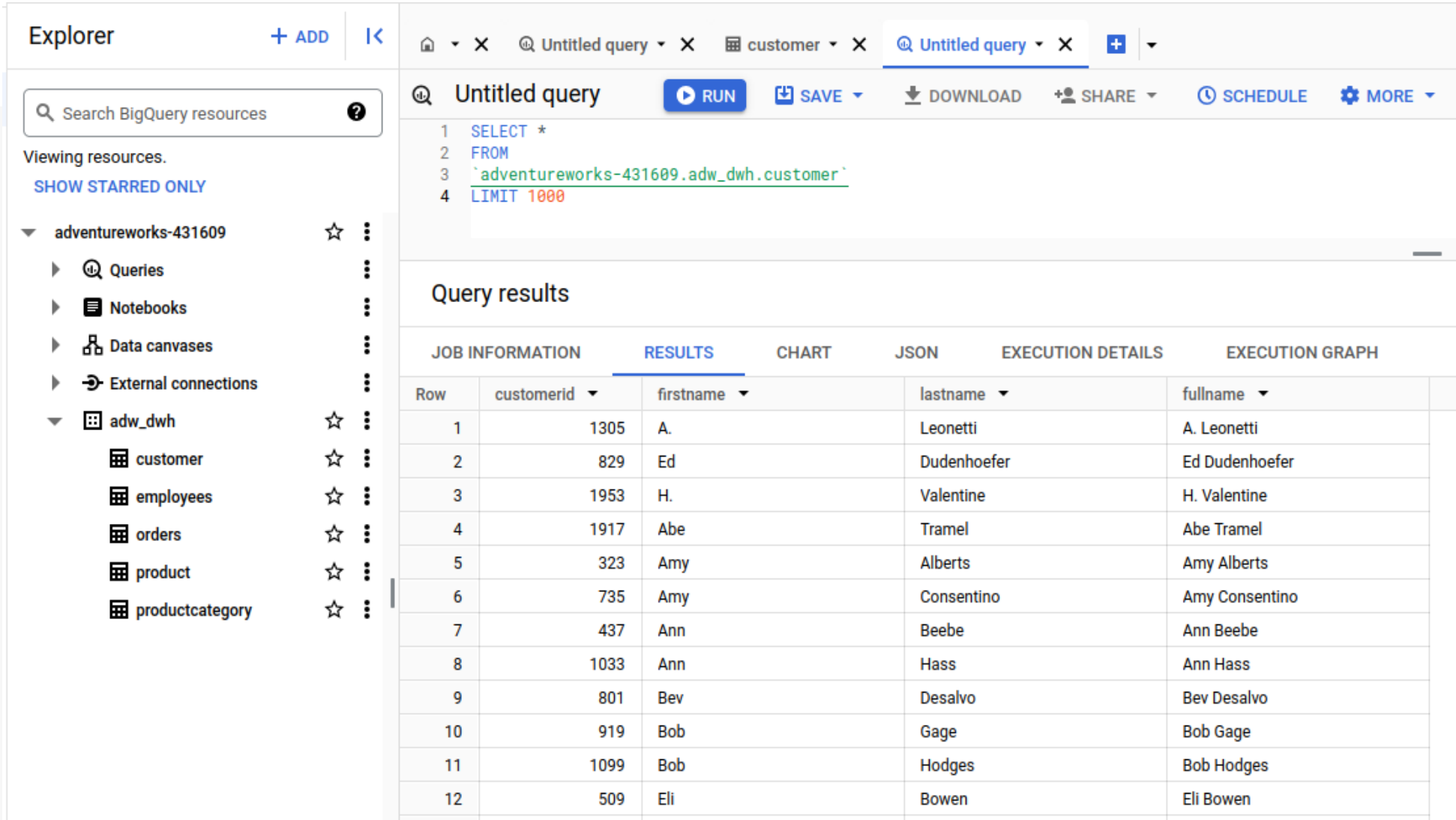
Setting up Apache Airflow

Load Task Group – Destination BigQuery

```
task()
@def load(dict):
    pandas_gbq.to_gbq(
        df,

        project_id=project_id,
        if_exists=append,
load(data)
```

```
Set dependencies

extract = extract()
load = load(extract)

extract >> load
```


Airflow
DAGs
Cluster Activity
Datasets
Security
Browse
Admin
Docs
09:12 EAT (+03:00)
BG
DAG: myjobmag_etl_pipeline
Schedule: 1:00:00
Next Run ID: 2024-08-01, 08:00:00 EAT
08 / 01 / 2024 08 : 31 : 58 AM
All Run Types
All Run States
Clear Filters
Auto-refresh
25
Press shift + / for Shortcuts
deferred
failed
queued
removed
restarting
running
scheduled
shutdown
skipped
success
up_for_reschedule
up_for_retry
upstream_failed
no_status
Duration
Aug 01, 00:44
00:01:01
00:00:30
00:00:00
gt_response_ke
tf_response_ke
gt_response_gh
tf_response_gh
gt_response_za
tf_response_za
gt_response_ng
tf_response_ng
merge_res
gt_delta
load_delta
DAG
myjobmag_etl_pipeline
Run
2024-08-01, 08:00:00 EAT
Clear
Mark state as...
Details
Graph
Gantt
Code
Audit Log
Layout:
Left -> Right
gt_response_za
success
@task
tf_response_za
success
@task
gt_response_gh
success
@task
tf_response_gh
success
@task
merge_res
success
@task
gt_delta
success
@task
load_delta
success
@task
gt_response_ke
success
@task
tf_response_ke
success
@task
gt_response_ng
success
@task
tf_response_ng
success
@task
React Flow


Explorer
+ ADD
Search BigQuery resources
Viewing resources.
SHOW STARRED ONLY
adventureworks-431609
Queries
Notebooks
Data canvases
External connections
adw_dwh
customer
employees
orders
product
productcategory
Untitled query
customer
Untitled query
Untitled query
RUN
SAVE
DOWNLOAD
SHARE
SCHEDULE
MORE
1 SELECT *
2 FROM
3 `adventureworks-431609.adw_dwh.customer`
4 LIMIT 1000
Query results
JOB INFORMATION
RESULTS
CHART
JSON
EXECUTION DETAILS
EXECUTION GRAPH
Row customerid firstname lastname fullname
1 1305 A. Leonetti A. Leonetti
2 829 Ed Dudenhoefer Ed Dudenhoefer
3 1953 H. Valentine H. Valentine
4 1917 Abe Tramel Abe Tramel
5 323 Amy Alberts Amy Alberts
6 735 Amy Consentino Amy Consentino
7 437 Ann Beebe Ann Beebe
8 1033 Ann Hass Ann Hass
9 801 Bev Desalvo Bev Desalvo
10 919 Bob Gage Bob Gage
11 1099 Bob Hodges Bob Hodges
12 509 Eli Bowen Eli Bowen