

E-COMMERCE DATA PIPELINE & DATA INSIGHTS

Data Engineering Team:

- Bandekar Anjali Ranjit
- Irene Lua
- M Logaraj
- Yee Soon Tuck

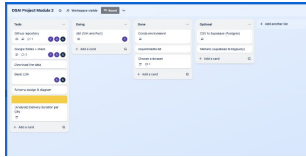


Agenda

1. Project Management Approach
2. Logical Data Pipeline Architecture
3. Platform Data Pipeline Architecture (2 Variations)
4. Data Warehouse Design
5. ELT Pipeline
6. Data Quality Testing
7. Data Analysis & Insights
8. Pipeline Orchestration



Trello



Project Management Strategy – Trello

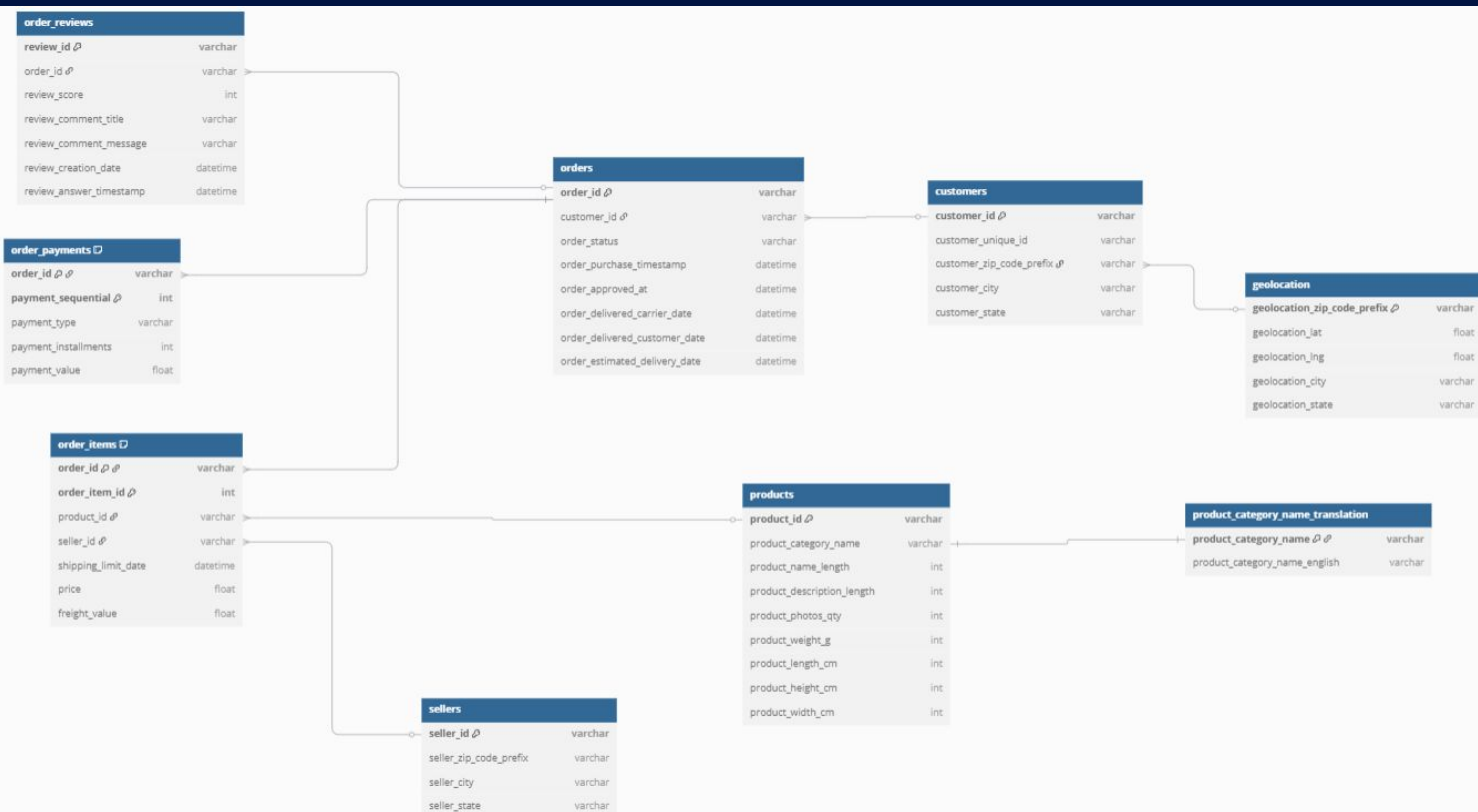
- ☐ Trello was utilized for tracking project tasks and progress.
- ☐ Each team member has designated cards for their responsibilities.
- ☐ The board is organized into columns for 'To Do', 'In Progress', and 'Done'.
- ☐ Guide for daily meetings
- ☐ Able to give comments ensure clear communication and accountability

Benefits of Using Trello

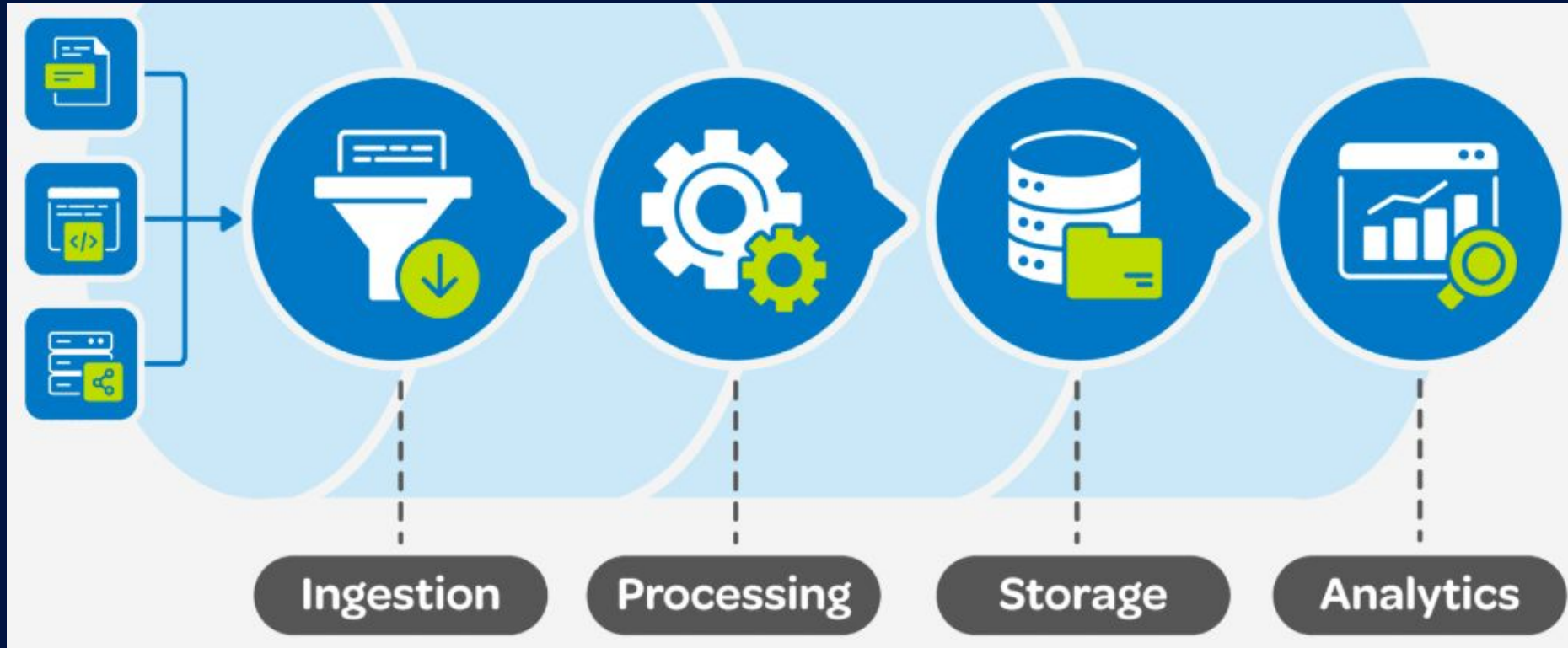
- ☐ Visual organization enhances team collaboration and transparency.
- ☐ Easy to assign tasks and set deadlines within the platform.
- ☐ New ideas/links could be easily updated and read by team members
- ☐ Trello's mobile app allows team members to update tasks on the go.

Source Data

Data set: Brazilian E-Commerce Public Dataset by Olist



Logical Data Pipeline Architecture



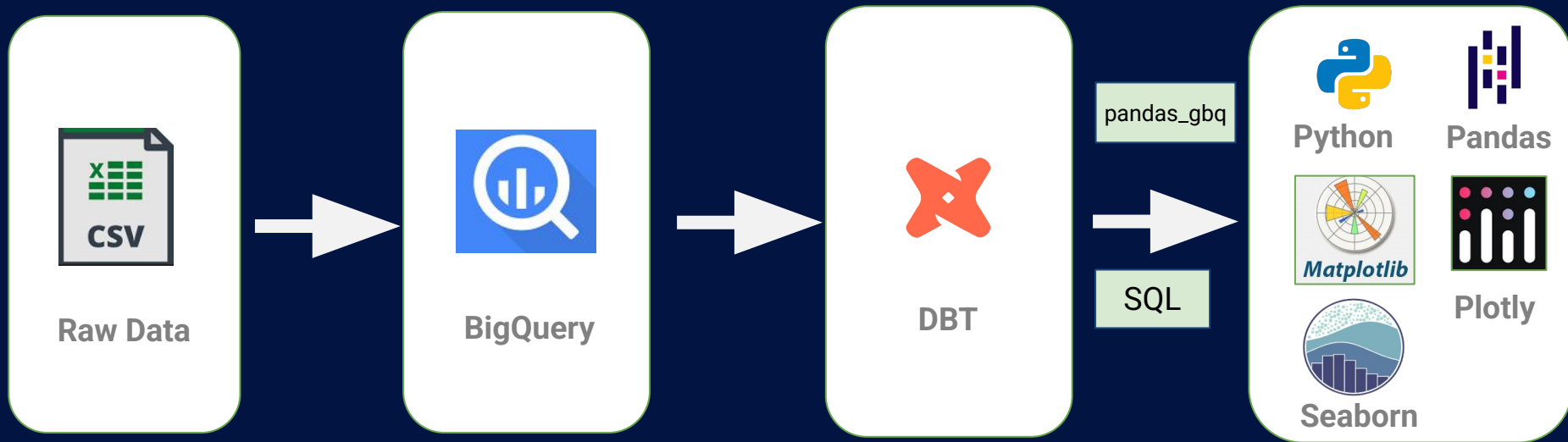
Data Pipeline Architecture v 1.0

Data Source

Data Storage

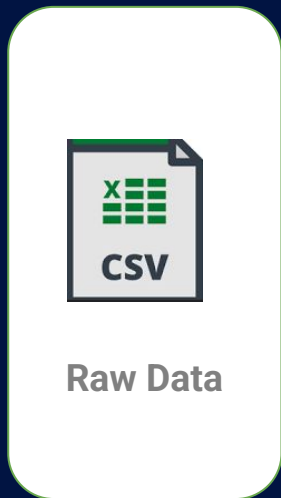
Data Preparation

BI & Analytics



Data Pipeline Architecture v 2.0

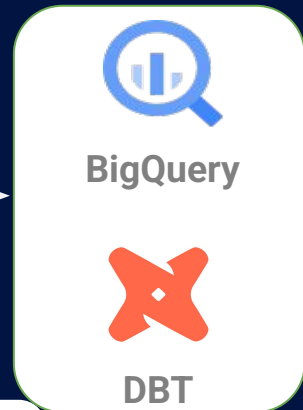
Data Source



Data Storage



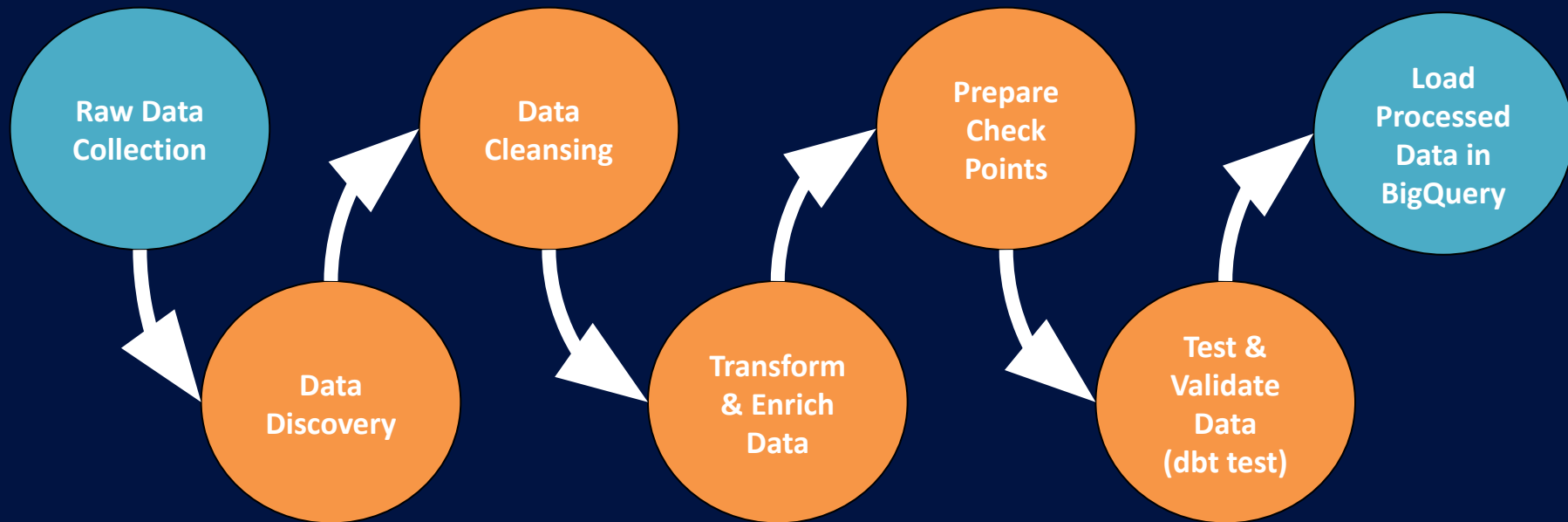
Data Preparation



BI & Analytics



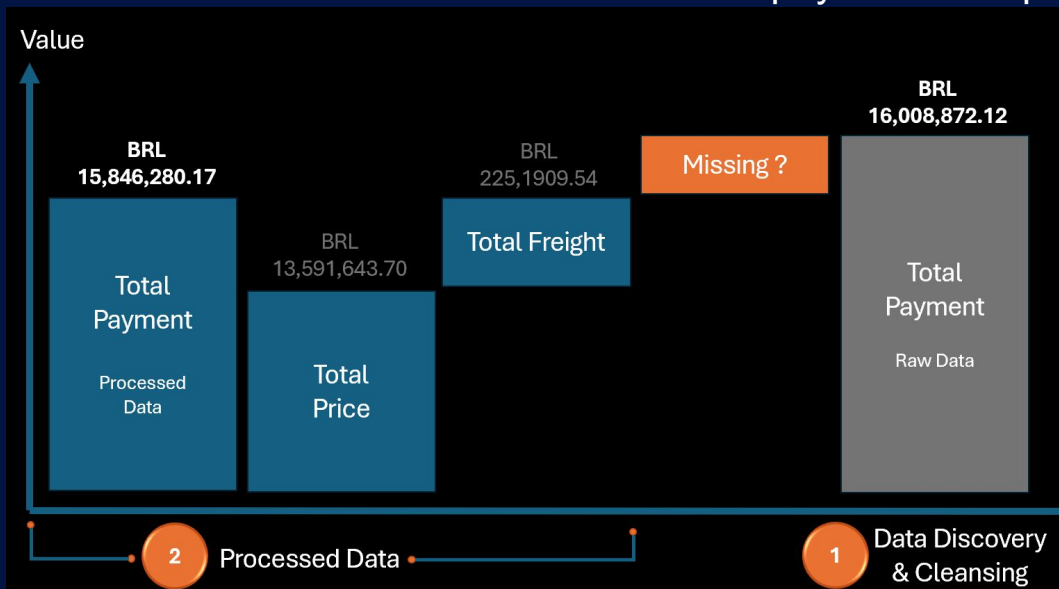
Data Preparation



Data Exploration & Validation

Test Case:

Does the delivered order value match our payment receipts?



Test Case working files link

New Query : Checking **Missing Data**

Total : 833 payments transactions

776 distinct orders with BRL162,591.95

	order_status	order_status_count	price_sum
0	canceled	179	0.00
1	created	5	0.00
2	delivered	3	134.97
3	invoiced	2	0.00
4	shipped	1	0.00
5	unavailable	643	0.00

Check Point:

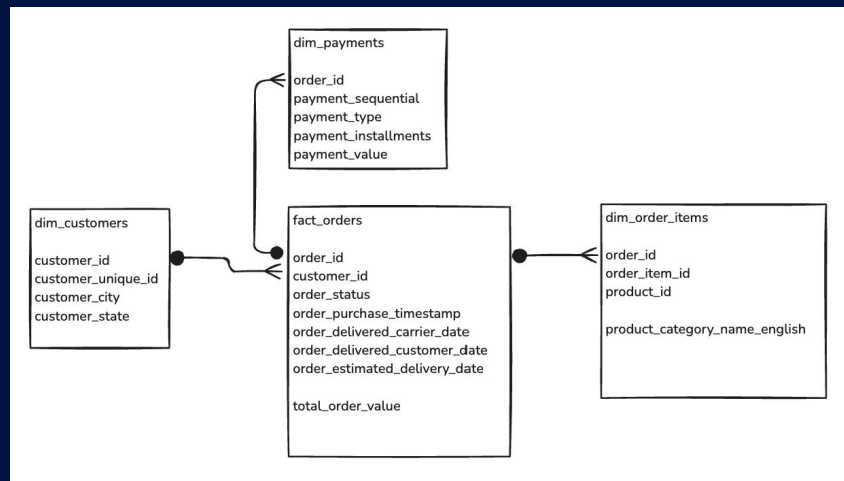
We accept 'canceled' status for dirty data. But the rest of the transactions require to rectify and clean up up.

```
models:
  - name: check_order_item_payment
    description: 'Model description'
    columns:
      - name: order_status
        description: Only 'canceled' status is acceptable here.
        tests:
          - dbt_utils.expression_is_true:
              expression: "{{ order_status }}" = 'canceled'
```

Data Warehouse Design (Star Schema)

Business Question: Providing Insights on Delivery Performance of Olist

1. Central Table — **fact_orders**
 - a. Tracks every order placed, including key timestamps, delivery status, and total value.
 - It's the heart of the schema where all delivery metrics are measured.
2. Customer Dimension — **dim_customers**
 - a. Adds geographic context (city, state) to each order, enabling analysis of regional delivery trends and customer behavior.
3. Item Dimension — **dim_order_items**
 - a. Breaks down orders into individual items with price, freight cost, and product category — crucial for understanding delivery cost drivers and product-based delays.
4. Payment Dimension — **dim_order_payments**
 - a. Links payment methods and values to orders, allowing insights into how payment types affect delivery performance (e.g., delays with boleto payments).



ELT Pipeline Features

1. Transformation of Raw Data into a Star Schema Using dbt:

- Utilized dbt to structure the raw data into a star schema, enhancing query performance and simplifying data analysis.
- Created SQL models to represent different dimensions and fact tables within the schema.

2. Implementation of Data Cleaning and Validation Steps:

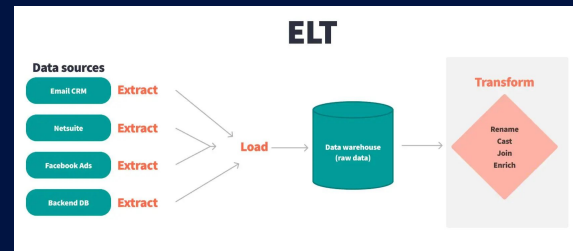
- Employed dbt's testing capabilities to ensure data quality by defining tests within the project, such as checking for uniqueness, non-null constraints, dbt_utils functions

3. Creation of Derived Columns:

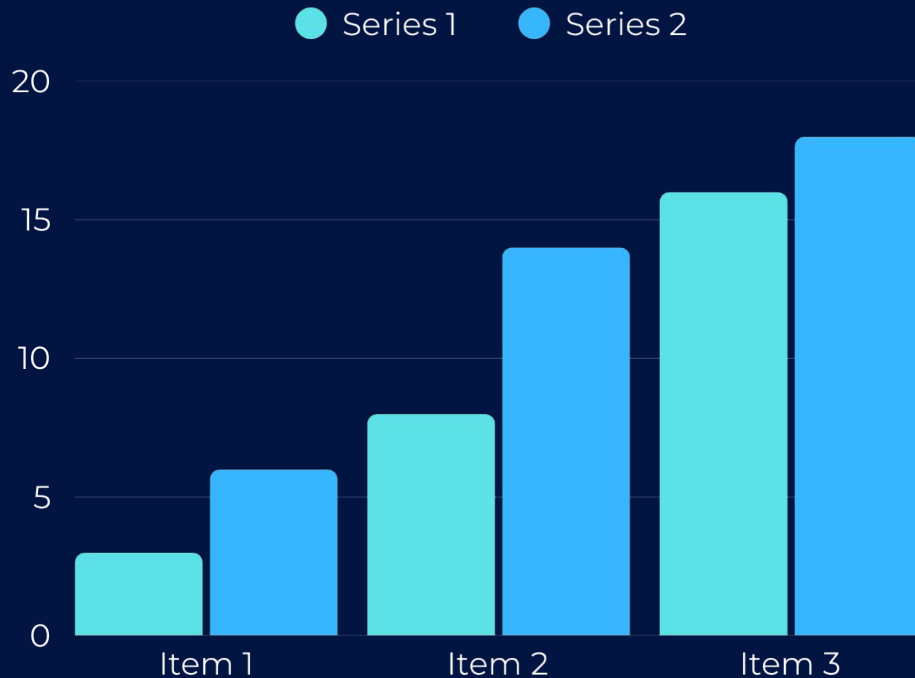
- Developed additional metrics like **total_order_amount** within dbt models to enrich the dataset and provide deeper analytical insights.

4. Exploratory Data Analysis Using Jupyter Notebooks:

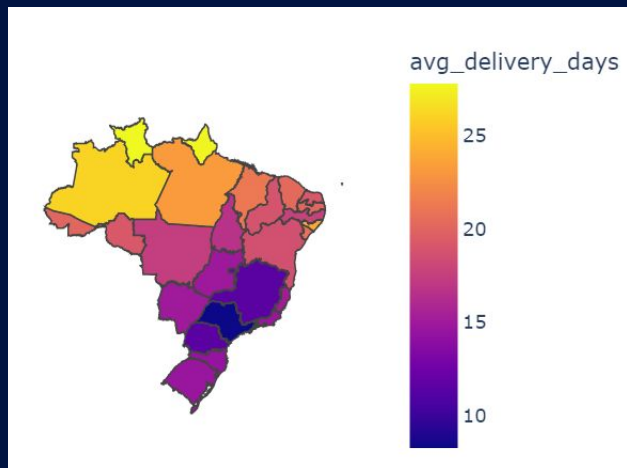
- Utilized Jupyter Notebooks to perform exploratory data analysis on the transformed data, allowing for interactive examination and visualization of trends and patterns.



Data Analysis Insights



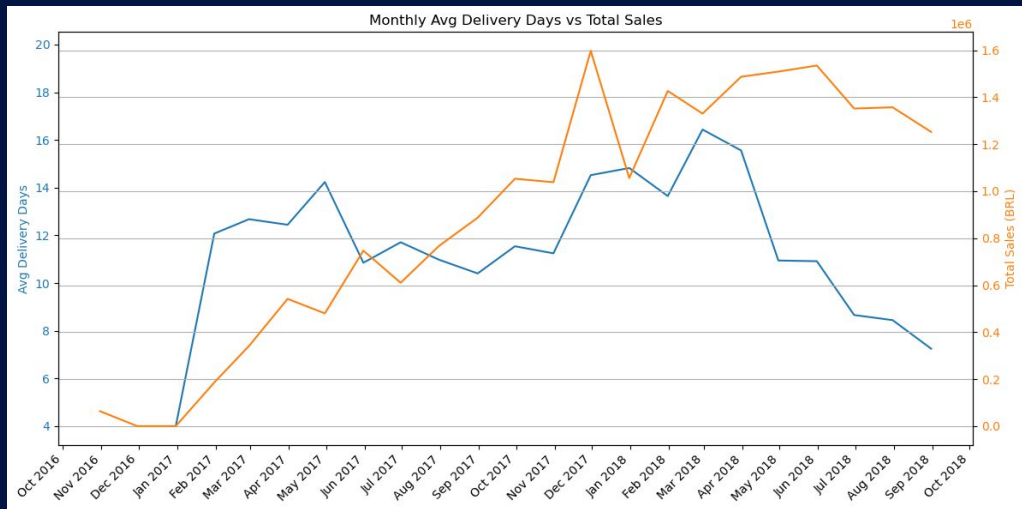
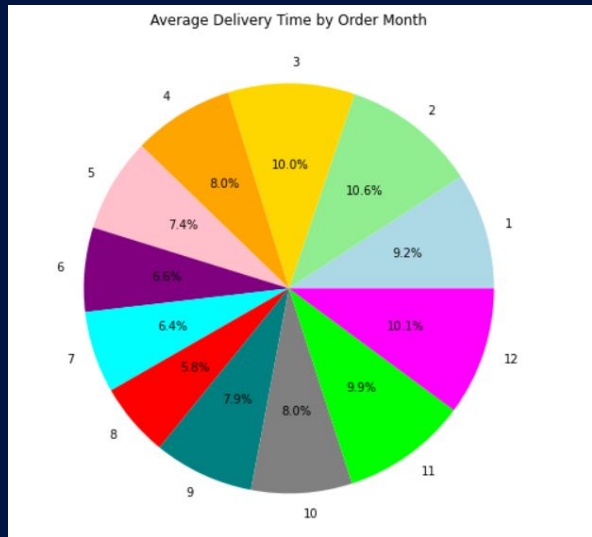
1) Northern States in Brazil had the highest average delivery dates



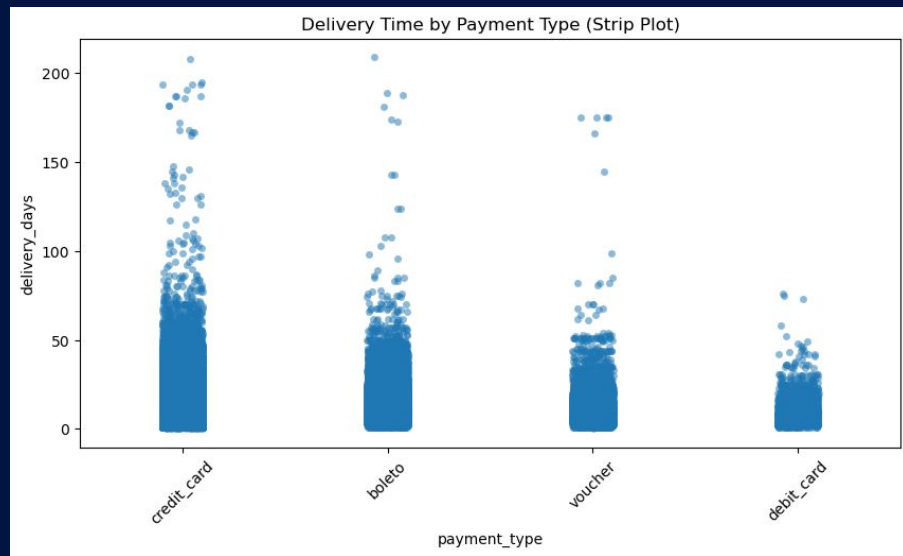
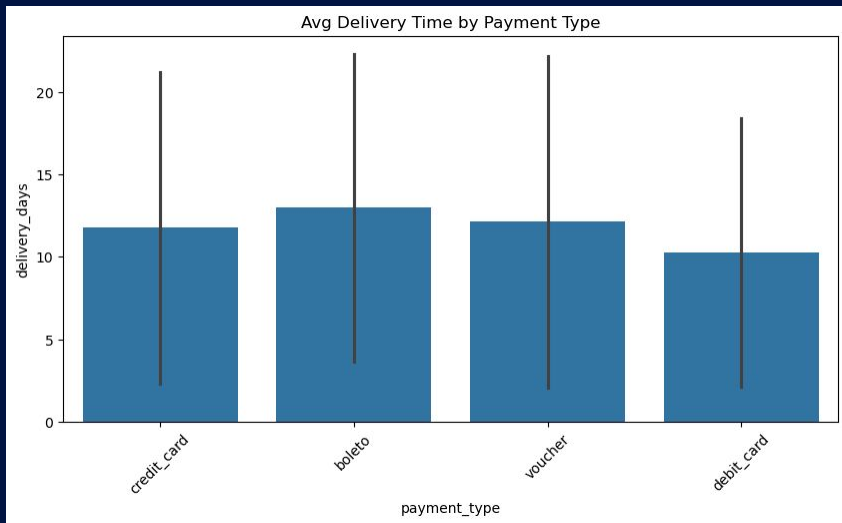
2) Bulkier Items show a stronger correlation to higher delivery times

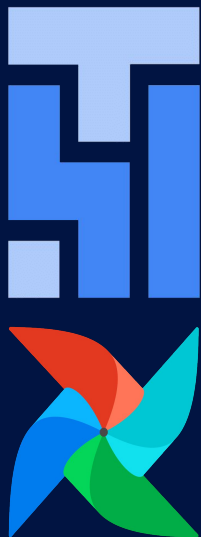


3) Delivery times peak in February, March, November, and December due to seasonal demand, holidays



4) Boleto & Credit Card Transactions have higher delivery times





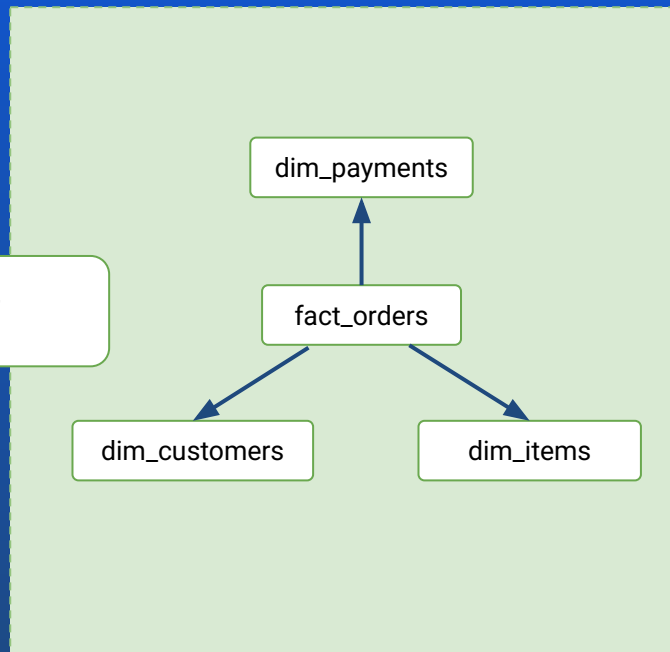
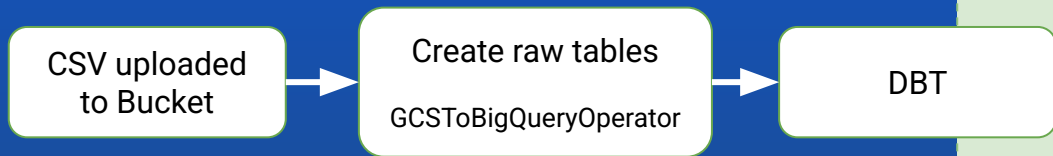
Pipeline Orchestration - Google Cloud Composer

Managed service, using Apache Airflow.

Note: this orchestration pipeline is connected to a different dataset in BigQuery (composer_test) for the purpose of presentation.

The other parts of this Project uses the “brazil_e_commerce” dataset.

Data Pipeline in Google Cloud Composer



DAGs folder in Bucket

The screenshot shows the Google Cloud Storage interface for a bucket named 'us-central1-project2-compos-5c62fdca-bucket'. The bucket is located in 'us-central1 (Iowa)' with 'Standard' storage class, 'Public access' set to 'Subject to object ACLs', and 'Protection' set to 'Soft delete'.

The 'OBJECTS' tab is selected, showing a 'Folder browser' on the left and a list of objects on the right. The folder browser shows the following structure:

- us-central1-project2-compos-5c62fdca-bucket
 - dags/
 - models/
 - data/
 - logs/
 - plugins/

The main panel shows the 'dags' folder contents, filtered by name prefix only. The table below lists the objects:

<input type="checkbox"/>	Name	Size	Type	Created	Storage class	Last modified
<input type="checkbox"/>	airflow_monitoring.py	798 B	text/x-python	25 Mar 2025, 10:45:01	Standard	25 Mar 2025
<input type="checkbox"/>	dbt_dag2.py	1.7 KB	text/x-python-script	27 Mar 2025, 20:32:21	Standard	27 Mar 2025
<input type="checkbox"/>	dbt_project.yml	1.2 KB	application/x-yaml	26 Mar 2025, 18:02:57	Standard	26 Mar 2025
<input type="checkbox"/>	load_csv.py	5.7 KB	text/x-python-script	27 Mar 2025, 20:07:10	Standard	27 Mar 2025
<input type="checkbox"/>	load_items.py	2.4 KB	text/x-python-script	27 Mar 2025, 20:07:11	Standard	27 Mar 2025
<input type="checkbox"/>	load_orders.py	2.4 KB	text/x-python-script	27 Mar 2025, 20:01:48	Standard	27 Mar 2025
<input type="checkbox"/>	models/	—	Folder	—	—	—

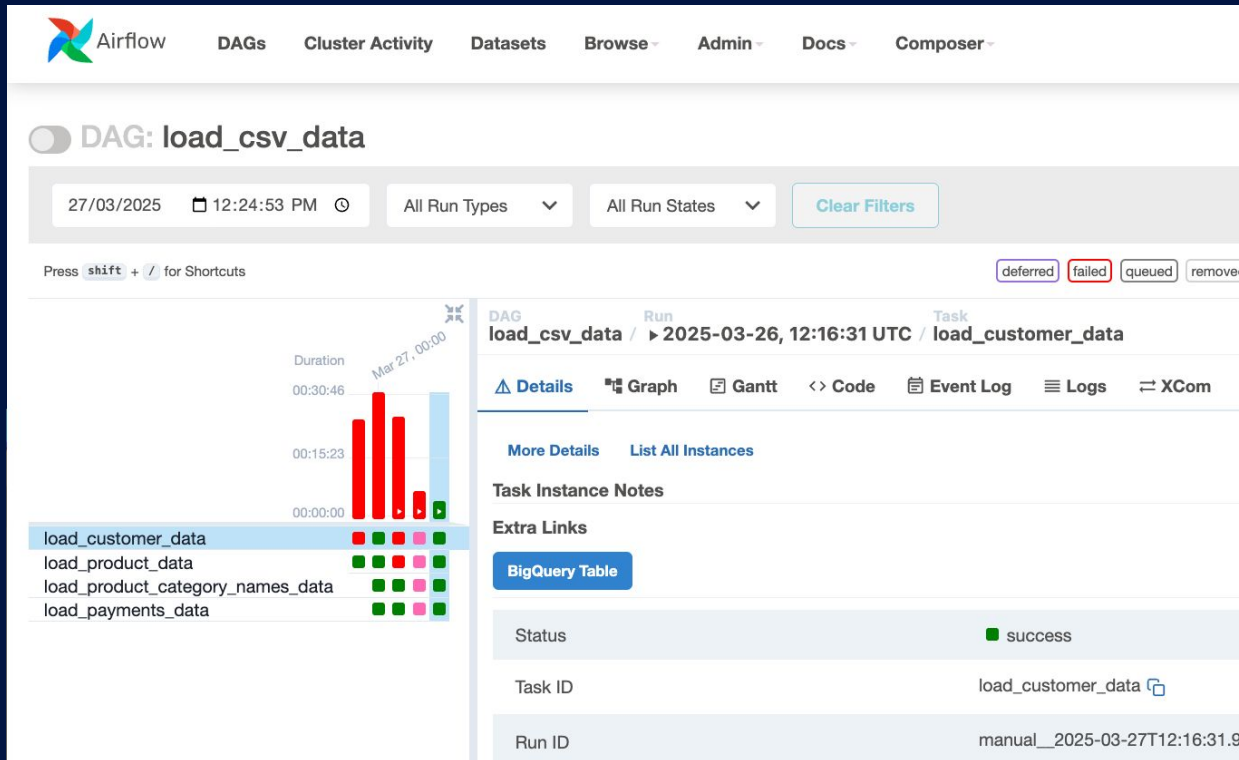
Airflow DAGs in Google Cloud Composer

The screenshot displays the Airflow web interface for a project named 'project2-composer'. The top navigation bar includes links for DAGs, Cluster Activity, Datasets, Browse, Admin, Docs, and Composer. Below the navigation bar, the project name 'project2-composer' is shown. A filter section indicates 5 All DAGs, 2 Active, and 3 Paused. There are also buttons for Running (0) and Failed (0) DAGs, and a search bar for DAGs. The main table lists five DAGs: airflow_monitoring, dbt_olist, load_csv_data, load_items_data, and load_orders_data. Each row shows the DAG name, owner (airflow), a status bar with counts (green for success, red for failure), the schedule, the last run time, and the next run time.

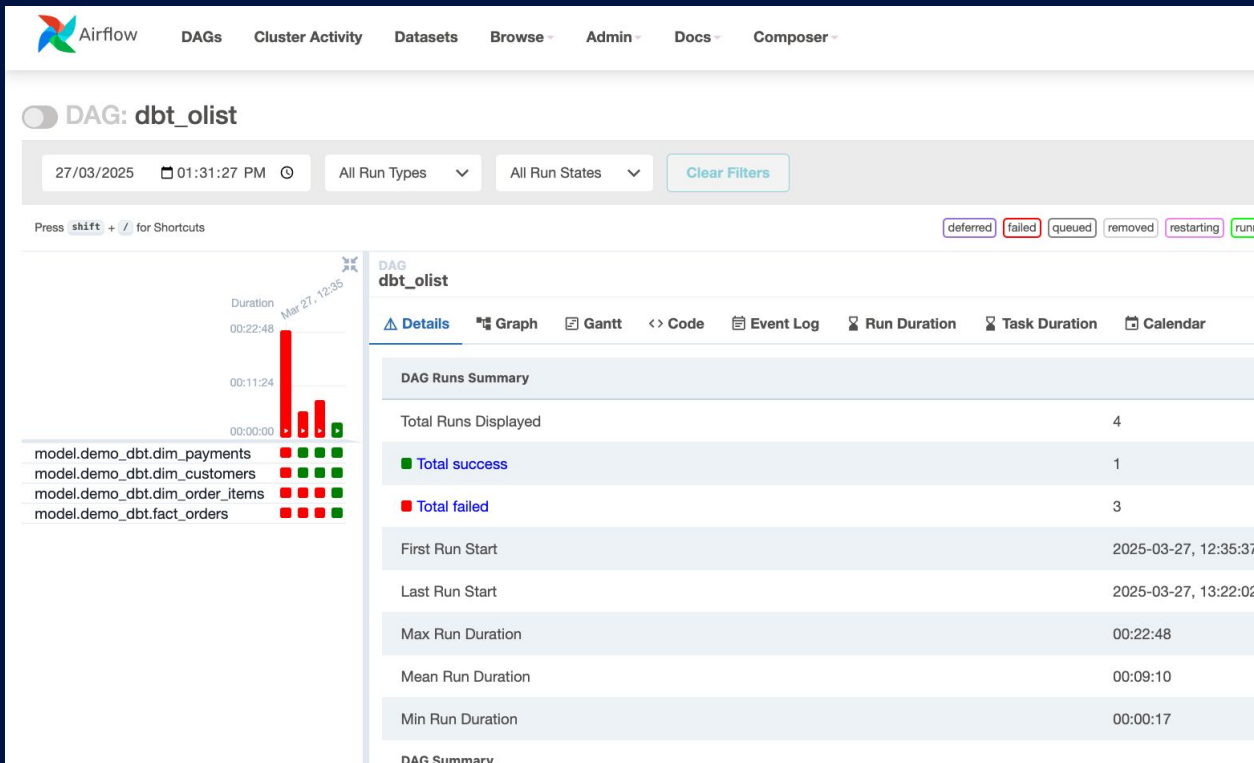
DAG	Owner	Runs	Schedule	Last Run	Next Run
airflow_monitoring	airflow	350 (green), 3 (red)	* / 10 * * * *	2025-03-27, 13:10:00	2025-03-27, 13:20:00
dbt_olist	airflow	1 (green), 3 (red)	None	2025-03-27, 13:22:02	
load_csv_data	airflow	1 (green), 4 (red)	1 day, 0:00:00	2025-03-27, 12:16:31	2025-03-27, 00:00:00
load_items_data	airflow	1 (green)	1 day, 0:00:00	2025-03-26, 00:00:00	2025-03-27, 00:00:00
load_orders_data	airflow	1 (green)	1 day, 0:00:00	2025-03-26, 00:00:00	2025-03-27, 00:00:00

At the bottom, there is a pagination control showing page 1 of 1.

load_csv_data DAG run logs



DBT running as DAG



Tables and Views in BigQuery

▼	composer_test	☆	⋮	10	a9fc3ae13
	dim_customers	☆	⋮	11	1b122a7b
	dim_order_items	☆	⋮	12	4b038a7f8
	dim_payments	☆	⋮	13	f175d6758
	fact_orders	☆	⋮	14	aa830d74
	olist_customers_raw	☆	⋮	15	c13f61a17
	olist_order_items_raw	☆	⋮	16	0e660617
	olist_order_payments_raw	☆	⋮	17	f2dd5f151
	olist_order_product_category_name_translation_raw	☆	⋮	18	1c47e787
	olist_orders_raw	☆	⋮	19	92580e70
	olist_products_raw	☆	⋮	20	3d24c492
				21	c504b1d6
				22	46c22ce0
				23	3eceaaf1
				24	47279b46

References

<https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce/data>

<https://cloud.google.com/bigquery/docs/samples/bigquery-pandas-gbq-read-gbq-simple>

<https://stripe.com/en-sg/resources/more/boleto-an-in-depth-guide>

<https://www.olist.com/>

<https://docs.getdbt.com/guides/bigquery>

https://schemas.getdbt.com/dbt/manifest/v12/index.html#nodes_additionalProperties_anyOf_i4

THANK YOU



+123-456-7890



WWW.OLIST-DATAELAB.COM



HELLO@OLIST-DATAELAB.COM



123 MONSERATT ST., SR, ST 12345