Objective

This document serves as a guide to understanding the Annie's Magic Numbers use case. It outlines the initial approaches, exploratory data analysis, and diagrams, providing a structured overview of the process.

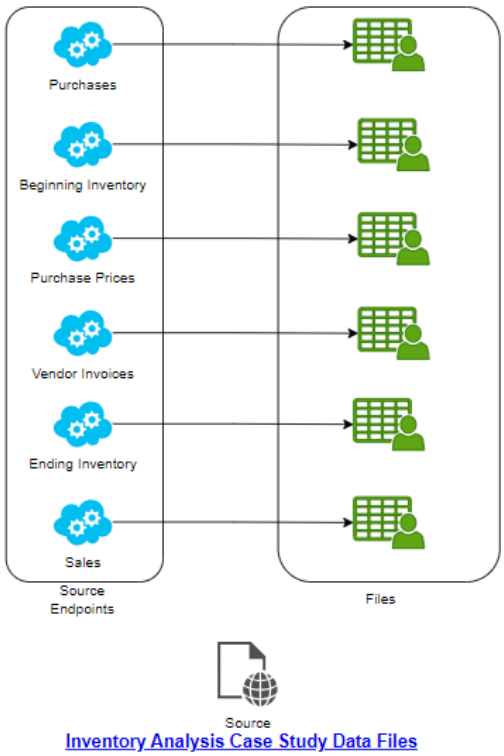
Data Sources

Location of the data

The data sources come from an endpoint available on the [Data analytics case study data files] page. Each endpoint provides CSV files that must be ingested to extract the information required by stakeholders.

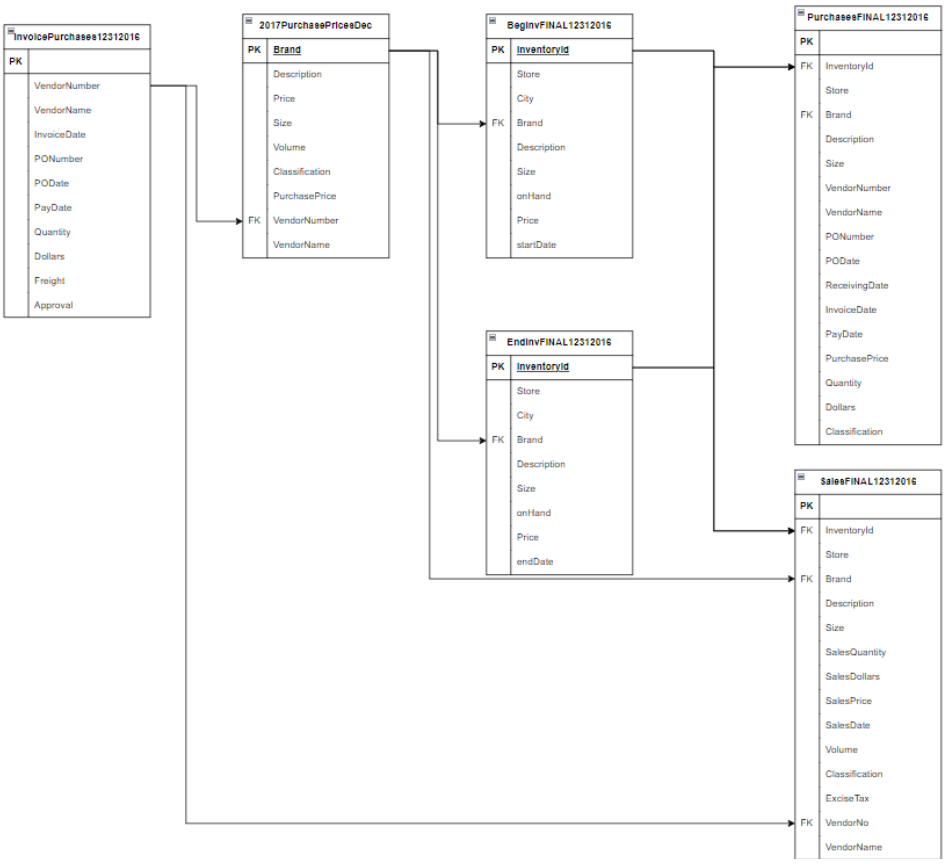
Data Relationships

In the initial analysis of the source, we identified data extracted from a transactional system. Some of these datasets appear to be related, as illustrated in the following diagram:



Some tables do not have a primary key, so it will be necessary to create one in the Data Warehouse.

During the initial analysis, we identified that some data is extracted from a transactional source. Based on this, we can infer potential relationships between datasets, as illustrated in the following diagram:



Calculations required

The calculation requested by the stakeholders are the following: **profits** and **margins**

Profit

$$\text{Profit} = \text{Total Revenue} - \text{Total Cost}$$

Margin

$$\text{Profit Margin} = \left(\frac{\text{Profit}}{\text{Total Revenue}} \right) \times 100$$

Exploratory Data Analysis
To perform the initial exploratory data analysis (EDA), we manually loaded the information into the Databricks Unity Catalog (Filestore). This allowed us to implement the first version of the profit and margin calculations.

The primary tables used for this analysis are:

- Sales
- Purchase

The SQL implementation is as follows:

```sql
%sql
SELECT
    s.InventoryId,
    s.Store,
    s.SalesDescription,
    s.SalesPrice,
    p.PurchasePrice,
    (s.SalesPrice - p.PurchasePrice) AS Profit,
    ((s.SalesPrice - p.PurchasePrice) / s.SalesPrice) * 100 AS ProfitMargin
FROM
    Sales s
LEFT JOIN
    PurchasePrices p
    ON s.Brand = p.Brand
LIMIT 5;
```

The complete exploratory data analysis is available in the following Databricks notebook: [AMN - Databricks Notebook]

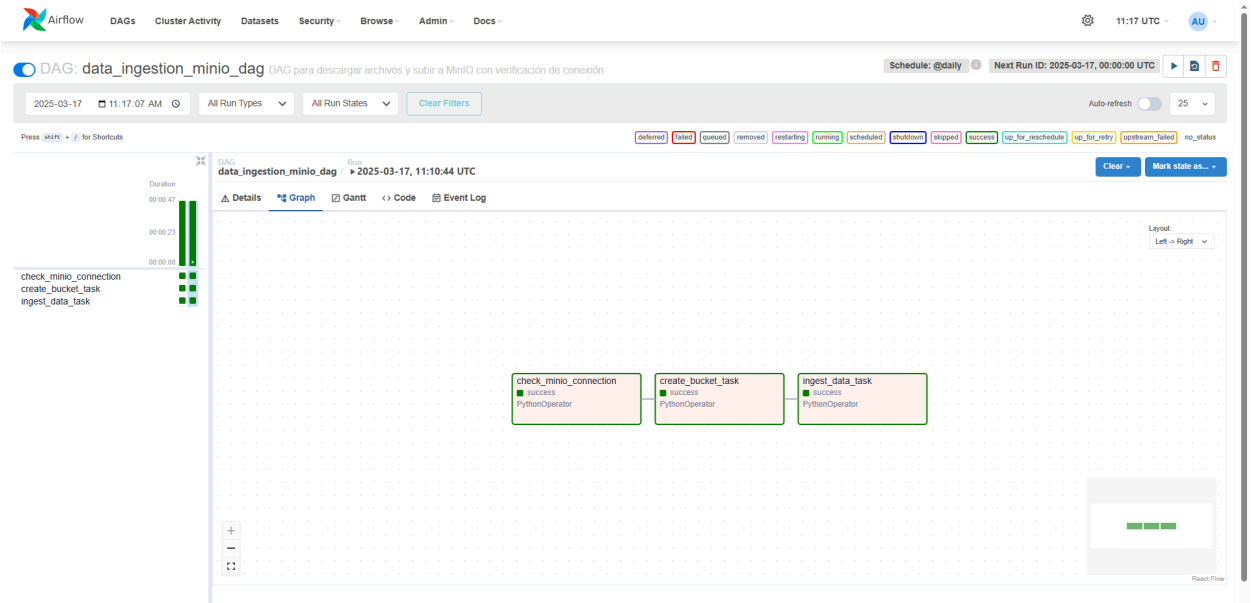
Brands not profitable

```sql
SELECT
    s.Brand,
    s.SalesDescription,
    (s.SalesPrice - p.PurchasePrice) AS Profit
FROM
    Sales s
LEFT JOIN
    PurchasePrices p
    ON s.Brand = p.Brand
WHERE
    (s.SalesPrice - p.PurchasePrice) < 0
```

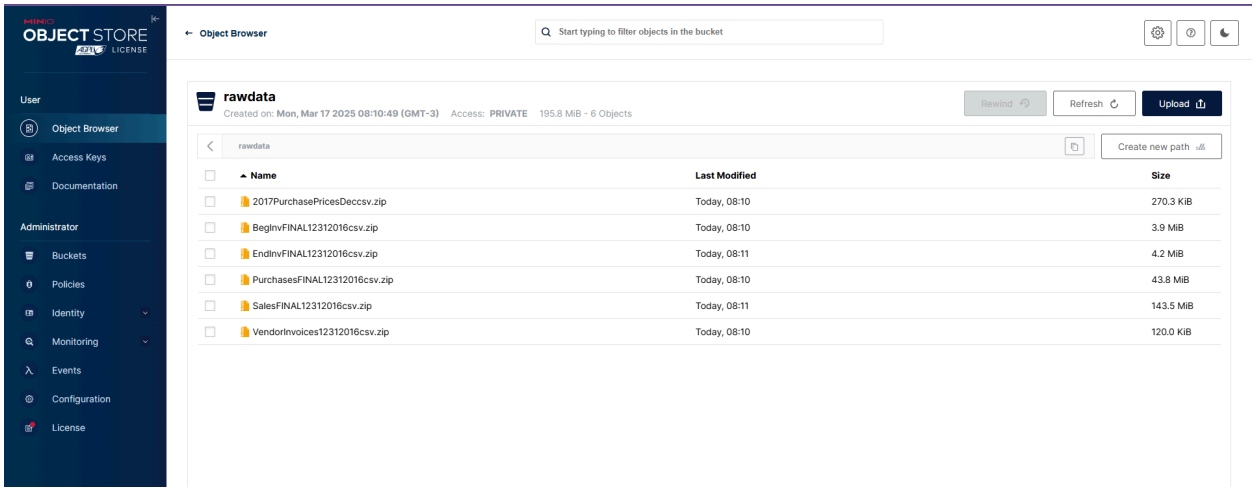
Steps

Ingest Data from Sources to S3 (Minio)

The movement of the data is made in a dag which moves from the endpoints to the data lake as .zip



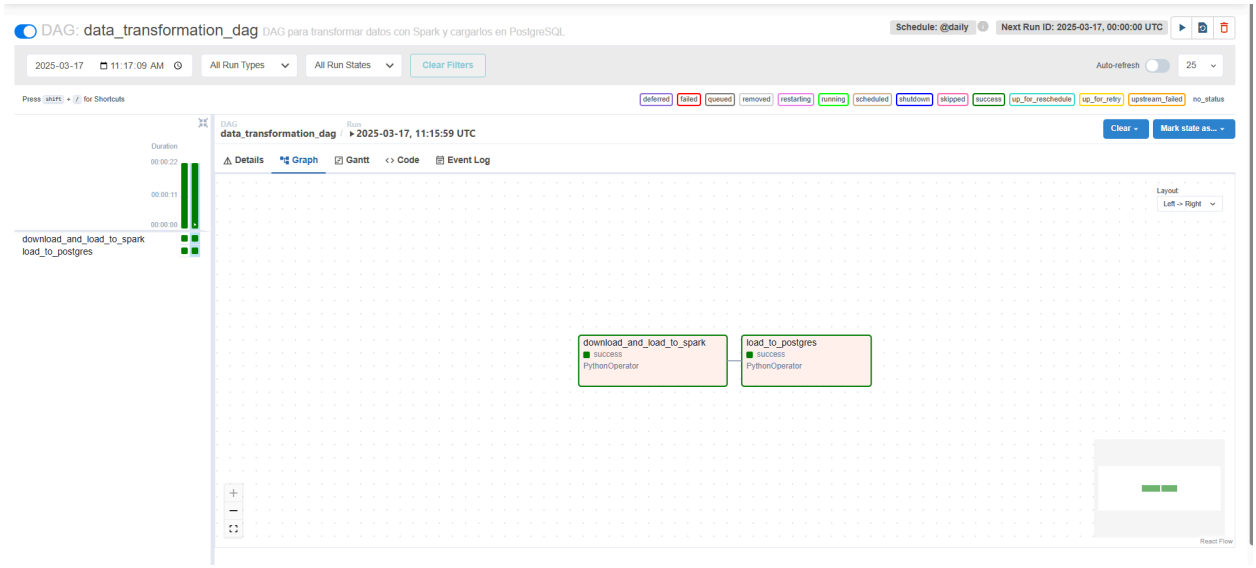
Location of the objects



Transforming the data taking only the information we need

Using Implementing a job in spark the information is taken from the data-lake and load into a relational database in postgres

Loading data



Spark Jobs

Spark Master at spark://f991cdfeac9e:7077

URL: spark://f991cdfeac9e:7077
Alive Workers: 1
Cores in use: 8 Total, 0 Used
Memory in use: 6.6 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 21 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers (1)

Worker Id	Address	State	Cores	Memory	Resources
worker-20250317110900-172.20.0.6-33801	172.20.0.6:33801	ALIVE	8 (0 Used)	6.6 GiB (0.0 B Used)	

Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration

Completed Applications (21)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20250317111909-0020	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:19:09	airflow	FINISHED	1 s
app-20250317111833-0019	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:18:33	airflow	FINISHED	1 s
app-20250317111757-0018	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:17:57	airflow	FINISHED	0.8 s
app-20250317111721-0017	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:17:21	airflow	FINISHED	1 s
app-20250317111645-0016	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:16:45	airflow	FINISHED	0.9 s
app-20250317111619-0015	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:16:19	airflow	FINISHED	3 s
app-20250317111619-0014	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:16:19	airflow	FINISHED	3 s
app-20250317111608-0012	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:16:08	airflow	FINISHED	3 s
app-20250317111608-0011	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:16:08	airflow	FINISHED	3 s
app-20250317111610-0013	Data Transformation with Spark	0	1024.0 MiB		2025/03/17 11:16:10	airflow	FINISHED	1 s
app-20250317111530-0010	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:15:30	airflow	FINISHED	1 s
app-20250317111453-0009	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:14:53	airflow	FINISHED	1 s
app-20250317111417-0008	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:14:17	airflow	FINISHED	1 s
app-20250317111341-0007	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:13:41	airflow	FINISHED	1 s
app-20250317111305-0006	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:13:05	airflow	FINISHED	0.9 s
app-20250317111229-0005	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:12:29	airflow	FINISHED	0.9 s
app-20250317111153-0004	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:11:53	airflow	FINISHED	1 s
app-20250317111116-0003	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:11:16	airflow	FINISHED	3 s
app-20250317110029-0002	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:10:29	airflow	FINISHED	1.0 s
app-20250317110953-0001	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:09:53	airflow	FINISHED	1 s
app-20250317110917-0000	Data Transformation with Spark	8	1024.0 MiB		2025/03/17 11:09:17	airflow	FINISHED	2 s

Destination in the warehouse