DP600

Intégration des Données Olist avec Microsoft Fabric

Rapporteur : **Jiayin CHEN**

Bilgenur OZDEMIR

Boris Yves NZEUYO DJOKO

Xuan Thu NGUYEN

Tuteur : **Alexandre BERGÈRE**

Sommaire

I.Présentation Générale du Projet

Ce projet consiste à intégrer les données d'e-commerce provenant de la plateforme brésilienne **Olist** dans un environnement **Microsoft Fabric**, en suivant une architecture moderne de type **Lakehouse**. L'objectif est de démontrer :

- La mise en place d'un **pipeline de traitement de données** depuis des fichiers CSV vers un modèle analytique utilisable.
- La création d'un entrepôt de données (**Data Warehouse**) structuré en **modèle en étoile**.
- L'implémentation de plusieurs types de **SCD (Slowly Changing Dimensions)** pour suivre l'évolution des données dans le temps.

1. Description des Données Sources :

- Nom du dataset :

  Brazilian E-Commerce Public Dataset by Olist

- Source :

  Kaggle - Brazilian E-Commerce Public Dataset by Olist
  https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce?resource=download&select=olist_customers_dataset.csv

2.Description des données :

- **Le jeu de données:**

  Le jeu de données provient de la plateforme de commerce électronique Olist, un site de vente en ligne brésilien qui vend des produits dans diverses catégories. Il s'agit de véritables données commerciales, elles ont été anonymisées et les références aux entreprises et partenaires dans le texte de la critique ont été remplacées par les noms des grandes maisons de Game of Thrones.

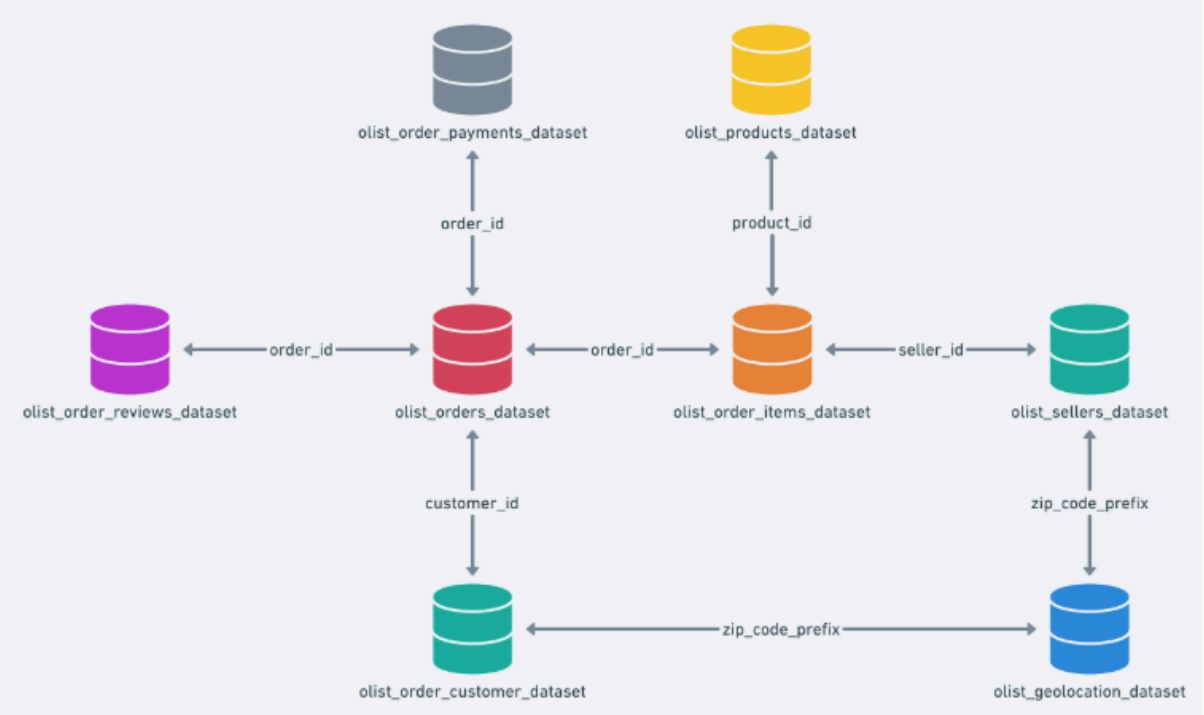
- **Ensemble de données disponible** :

  Les fichiers CSV suivants ont été utilisés (stockés dans le répertoire **Files/** de Fabric):

- **olist_customers_dataset.csv** : Informations sur les clients.
- **olist_geolocalisation_dataset.csv** : Données géographiques des clients.
- **olist_order_items_dataset.csv** : Détails sur les articles commandés (produits, prix, seller).
- **olist_products_dataset.csv** : Informations détaillées sur les produits.
- **olist_sellers_dataset.csv** : Informations sur les vendeurs.
- **olist_order_payments_dataset.csv** : Détails sur les paiements des commandes.
- **olist_order_reviews_dataset.csv** : Avis des clients sur les commandes.
- **olist_orders_dataset.csv** : Il s'agit de l'ensemble de données de base. Pour chaque commande, vous pouvez trouver toutes les autres informations.
- **product_category_name_translation.csv** : Traduit le nom de la catégorie de produit en anglais.

3.Schéma de données

:



Pour la réalisation du tableau de bord, nous avons utilisés **olist_order_payments_dataset, olist_order_reviews_dataset.csv, olist_order_items_dataset.csv, olist_orders_dataset.csv et product_category_name_translation.csv**.

II.Microsoft Fabric

1.Zone de Staging (Raw Layer)

Dans cette étape, les fichiers CSV sont lus via **PySpark** puis enregistrés dans des tables Delta dans Fabric. Voici le code exécuté :

```
# Import necessary modules
from pyspark.sql.functions import *
from datetime import datetime
# Define file paths (files uploaded to /lakehouse/default/Files/)
base_path = "Files/"
# Load CSV files into DataFrames
df_customers = spark.read.option("header", True).csv(base_path +
"olist_customers_dataset.csv")
df_geolocation = spark.read.option("header", True).csv(base_path +
"olist_geolocation_dataset.csv")
df_order_items = spark.read.option("header", True).csv(base_path +
"olist_order_items_dataset.csv")
df_order_payments = spark.read.option("header", True).csv(base_path +
"olist_order_payments_dataset.csv")
df_orders = spark.read.option("header", True).csv(base_path +
"olist_orders_dataset.csv")
df_products = spark.read.option("header", True).csv(base_path +
"olist_products_dataset.csv")
# Save as staging Delta tables
df_customers.write.mode("overwrite").format("delta").saveAsTable("stg_cu
stomers")
df_geolocation.write.mode("overwrite").format("delta").saveAsTable("stg_
geolocation")
df_order_items.write.mode("overwrite").format("delta").saveAsTable("stg_
order_items")
df_order_payments.write.mode("overwrite").format("delta").saveAsTable("s
tg_order_payments")
df_orders.write.mode("overwrite").format("delta").saveAsTable("stg_order
s")
df_products.write.mode("overwrite").format("delta").saveAsTable("stg_pro
ducts")
```

Tables créées dans la zone de staging :

- stg_customers
- stg_geolocation
- stg_order_items
- stg_order_payments
- stg_orders
- stg_products

2.Zone Data Warehouse – Modèle en Étoile

Nous avons structuré les données en modèle en étoile avec :

Fait :

- `fact_orders` (faits de commande)

Dimensions :

- `dim_customer` (Type SCD 2)
- `dim_product` (Type SCD 1)

Script PySpark utilisé :

2.1 `dim_customer` avec SCD Type 2

Ajout de colonnes pour la gestion temporelle :

```
df_cust = spark.table("stg_customers")
df_cust = df_cust.withColumn("hash_id", sha2(concat_ws("||",
*df_cust.columns), 256))
df_cust = df_cust.withColumn("start_date", current_date())
df_cust = df_cust.withColumn("end_date", lit("9999-12-31"))
df_cust = df_cust.withColumn("is_current", lit(True))
```

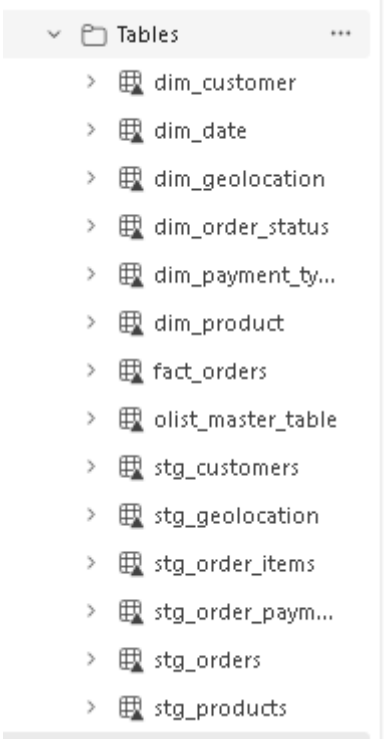
2.2 `dim_product` avec SCD Type 1 (simple remplacement)

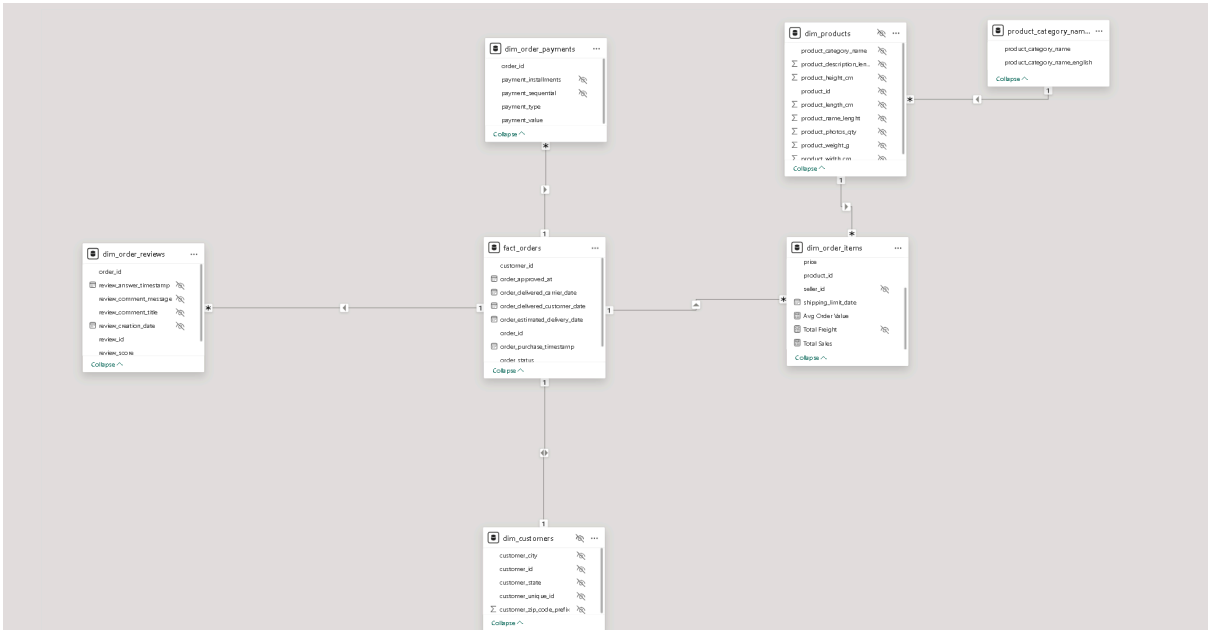
```
df_products = spark.table("stg_products")
df_products.write.mode("overwrite").format("delta").saveAsTable("dim_product
")
```

2.3 `fact_orders`

Créée par jointure de : `stg_orders`, `stg_order_items`, `stg_order_payments`

```
fact_orders = spark.sql("""
SELECT
  o.order_id,
  o.customer_id,
  i.product_id,
  i.seller_id,
  i.price,
  p.payment_type,
  p.payment_value,
  o.order_purchase_timestamp
FROM stg_orders o
JOIN stg_order_items i ON o.order_id = i.order_id
JOIN stg_order_payments p ON o.order_id = p.order_id
""")
```





3. Gestion des SCD – Détail

Table	Type SCD	Description
dim_customer	SCD Type 2	Historisation complète des changements via colonnes `start_date`, `end_date`, `is_current`
dim_product	SCD Type 1	Remplacement complet des données à chaque chargement

III.Power BI

1. Thème d'analyse choisi

Analyse des ventes sur la plateforme de commerce électronique Olist (09/2016-08/2018)

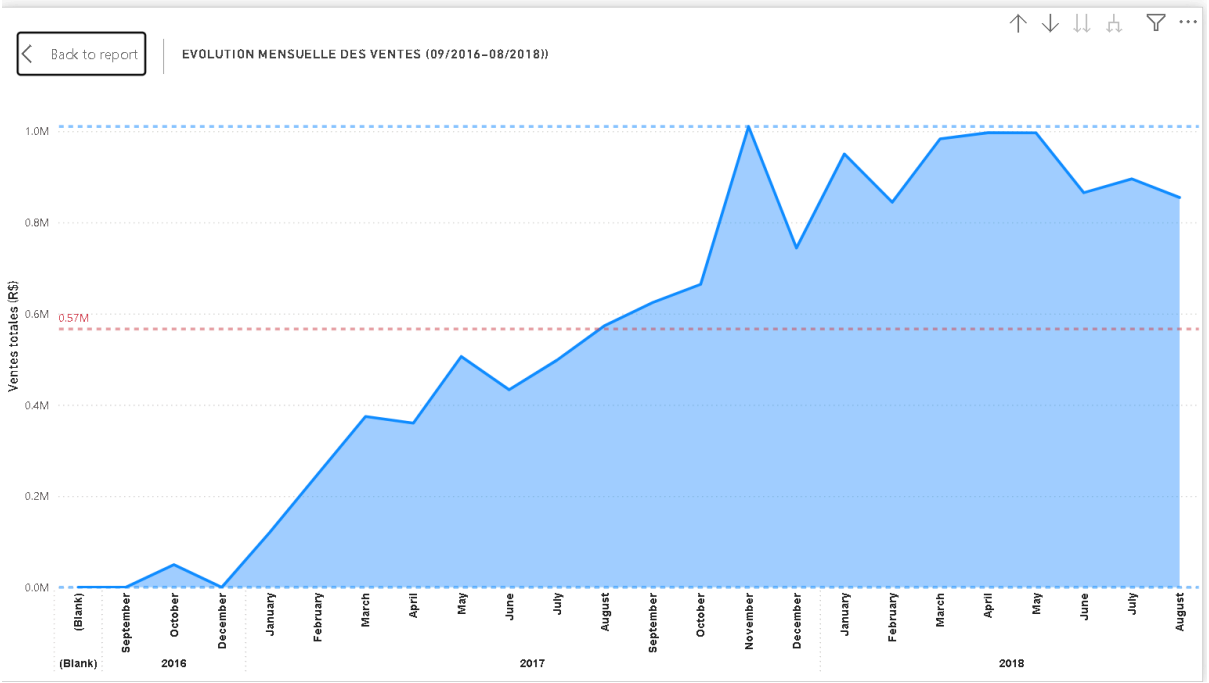
L'objectif de cette analyse est d'examiner les tendances de vente, les préférences des consommateurs et les modes de paiement utilisés sur la plateforme de commerce en ligne Olist au Brésil, sur une période de 2 ans, de septembre 2016 à août 2018. Les principales questions auxquelles cette analyse répondent sont les suivantes :

1. Évolution des ventes mensuelles :

Nous avons créé ici un graphique en courbes, avec les mois en abscisse (axe X) et un indicateur personnalisé intitulé **Ventes Totales** en ordonnée (axe Y). Cette mesure représente la valeur totale des transactions, calculée comme suit :

Total Sales = SUM(olist_order_items_dataset[price]).

Comment les ventes ont-elles évolué au fil des mois, et existe-t-il des tendances saisonnières ?

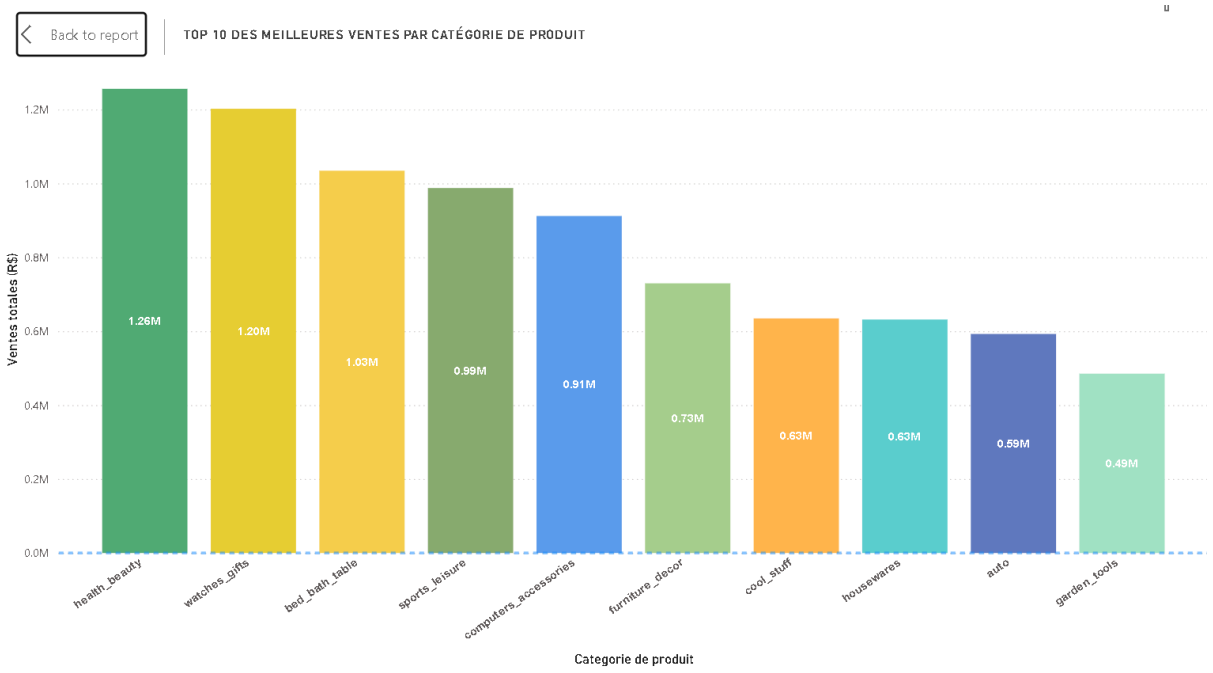


Les ventes mensuelles montrent une forte progression, avec des pics lors des périodes de fêtes, indiquant l'importance de planifier les campagnes et la gestion des stocks en fonction de la saisonnalité.

2. Top 10 des meilleures ventes par catégorie de produit :

Nous avons créé ici un graphique en barre, avec la catégorie de produit en abscisse (axe X) et un indicateur personnalisé intitulé **Ventes Totales** en ordonnée (axe Y). Un filtre a été appliqué pour ne conserver que les 10 premières catégories en termes de chiffre d'affaires.

Quelles sont les catégories de produits les plus populaires et comment se répartissent-elles en termes de ventes ?



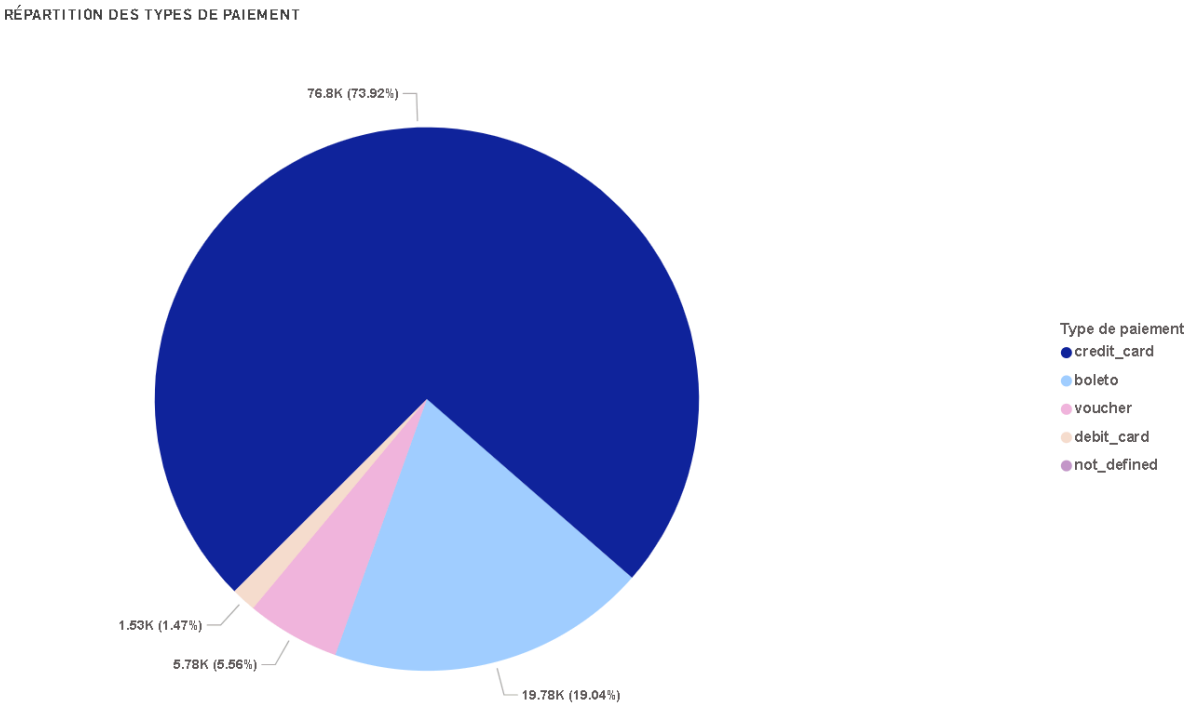
Les produits de la catégorie "health_beauty" arrivent en tête avec un total de plus de 1,26 million de ventes, suivis de près par les catégories "watches_gifts" avec 1,21 million de ventes et "bed_bath_table" avec 1.04 million de ventes.

Ces résultats soulignent l'importance de proposer un large éventail de produits dans ces segments clés.

3. Répartition des types de paiement :

Nous avons créé un graphique en camembert, où la **légende** représente les **types de paiement** et la **valeur** correspond au **nombre total de commandes** (count of order_id), c'est-à-dire le volume total des transactions.

Quels sont les modes de paiement les plus utilisés par les consommateurs sur la plateforme ?



La carte de crédit est le mode le plus utilisé, représentant 73,72 % des transactions avec boleto suit avec 19,04 % des transactions.

IV.Conclusion générale

Ce projet montre l'efficacité de Microsoft Fabric pour :

- Ingestions massives à partir de fichiers plats
- Construction rapide d'un entrepôt de données
- Suivi temporel des dimensions via SCD

La modularité de Fabric (Notebooks, Lakehouse, Delta tables) facilite un déploiement analytique rapide.