TP Cloud : Pipeline Moderne

Modern Data Stack avec AWS, Snowflake et dbt

Étudiant : Aymane EL MKADMI

Filière : Data Analytics

Date : 6 janvier 2026

1. Introduction au Modern Data Stack

1.1 L'évolution vers le Cloud

Les organisations migrent massivement leurs infrastructures vers le cloud, bénéficiant ainsi d'une scalabilité élastique, d'une optimisation des coûts (pay-as-you-go), d'une maintenance réduite et d'un accès rapide aux innovations technologiques.

1.2 Architecture du Modern Data Stack

Le Modern Data Stack combine des outils cloud-natifs modulaires : PostgreSQL (source OLTP), Amazon S3 (Data Lake), Snowflake (Data Warehouse), dbt (transformations), Airflow (orchestration) et Power BI (visualisation).

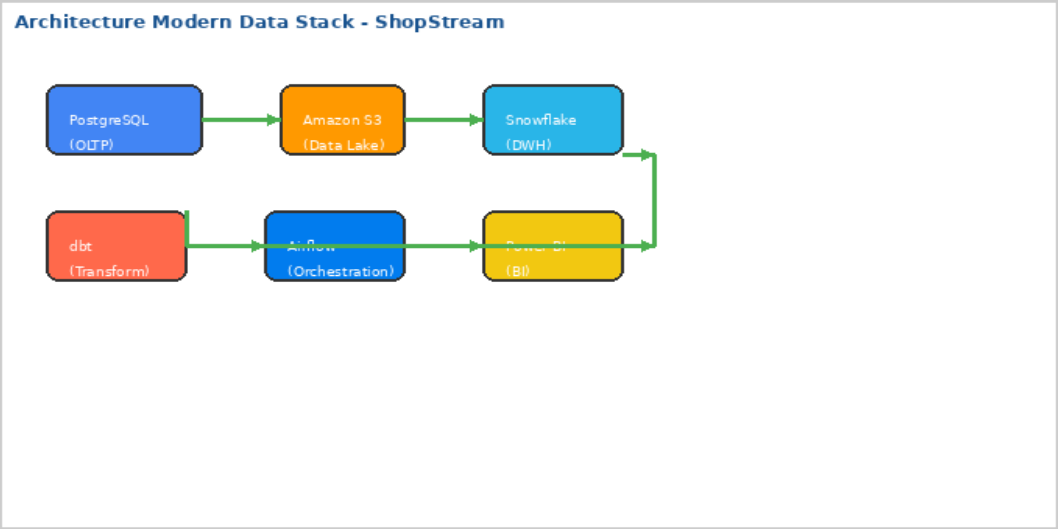

Figure 1 – Architecture du Modern Data Stack

1.3 Configuration de PostgreSQL

PostgreSQL sert de base de données transactionnelle source. La base shopstream est créée avec les tables : users, products, orders, order_items, events et crm_contacts.

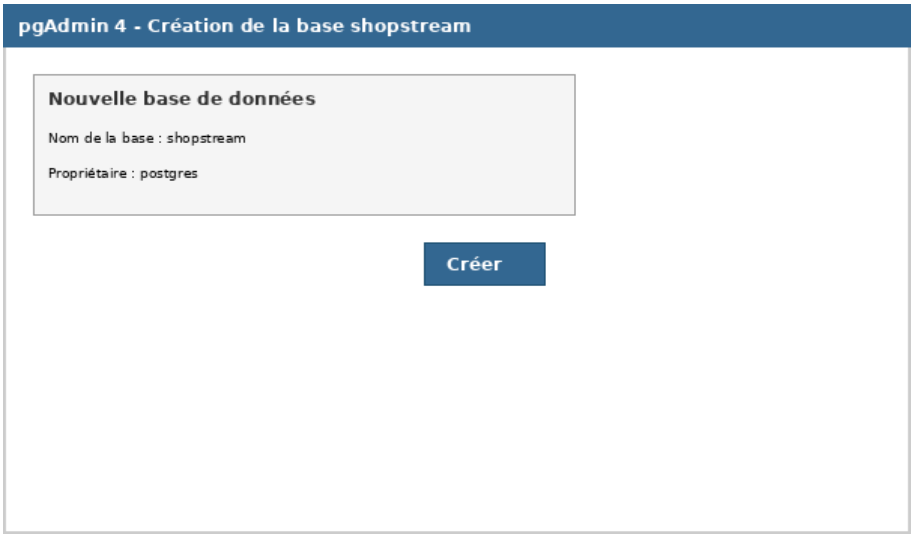

Figure 2 – Création de la base shopstream

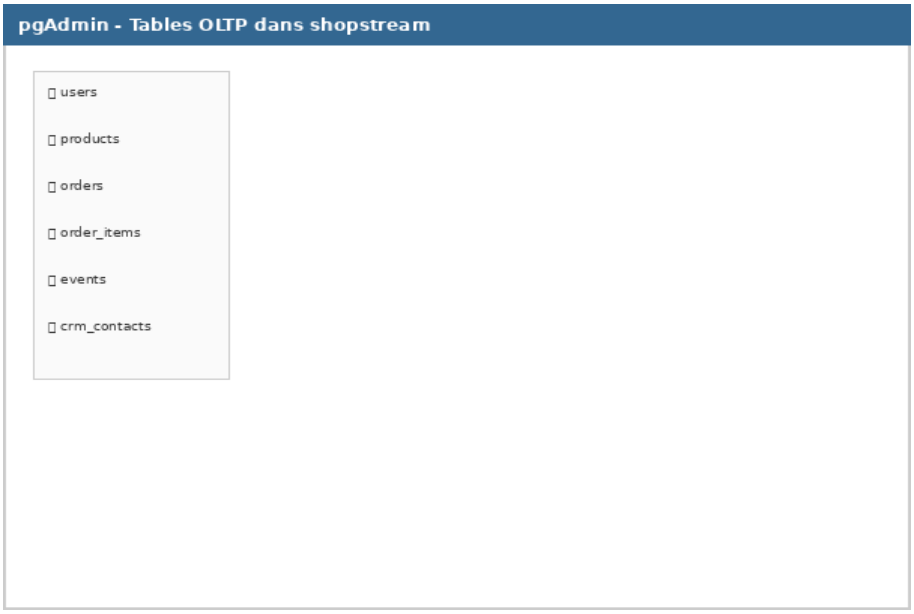

Figure 3 – Tables OLTP créées

1.4 Génération et vérification des données

Le script generate_data.py génère automatiquement des données de test (utilisateurs, produits, commandes, événements). La vérification dans pgAdmin confirme l'insertion correcte des enregistrements.

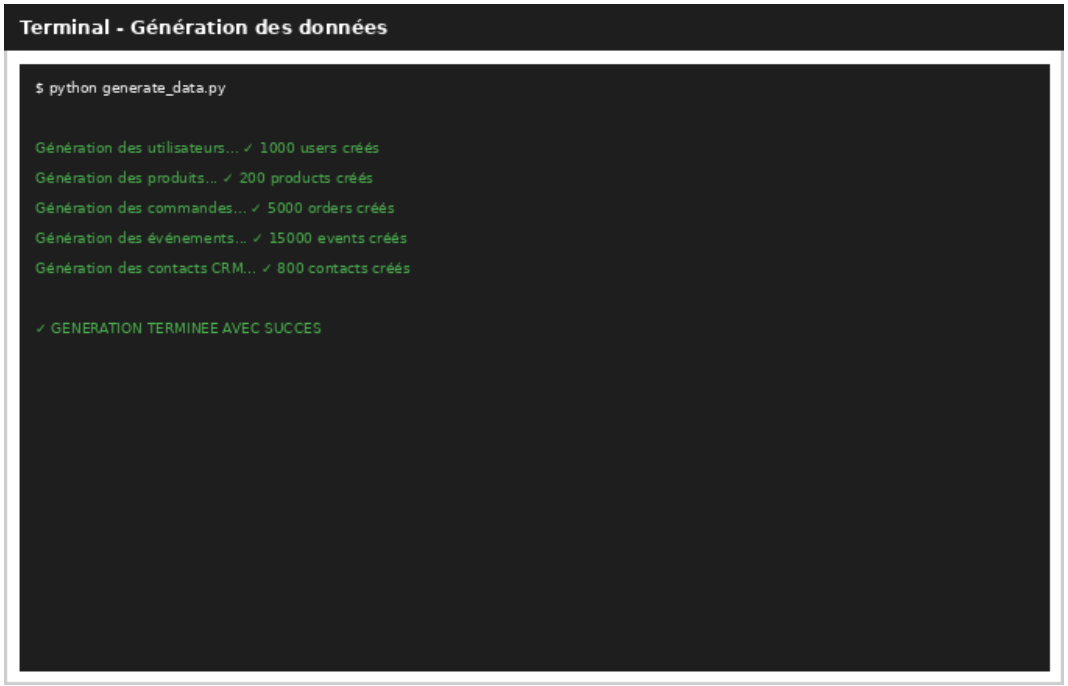

Figure 4 – Génération des données

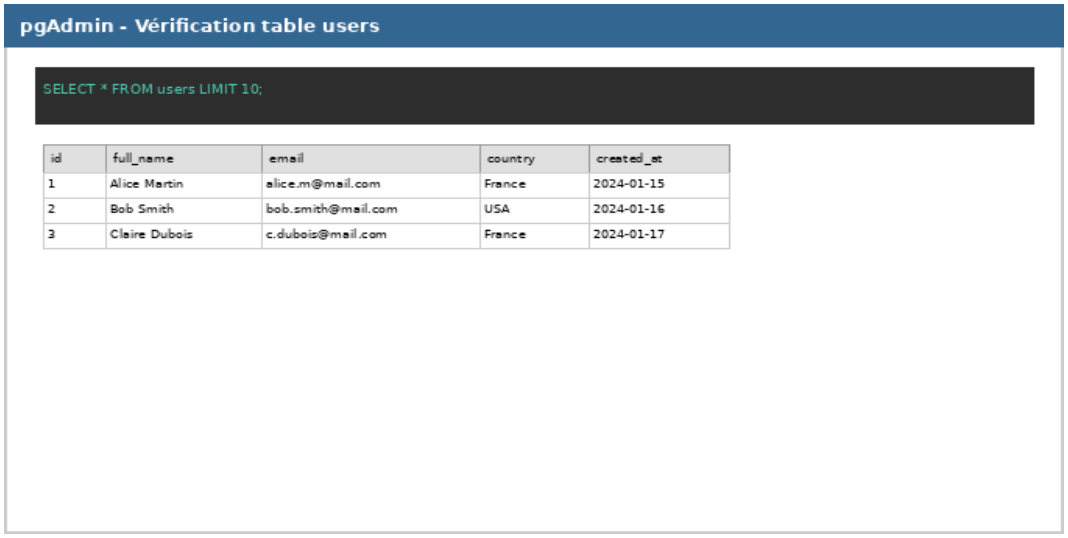

Figure 5 – Vérification table users

1.5 Création du Data Lake sur AWS S3

Un bucket S3 (shopstream-datalake-aymane) est créé dans la région Europe (Paris) avec chiffrement activé. Une structure de dossiers hiérarchique organise les données brutes par source.

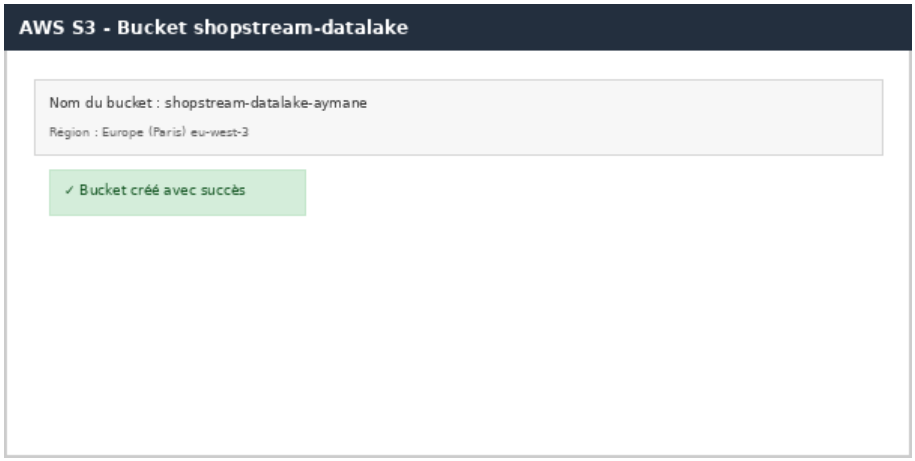

Figure 6 – Création du bucket S3

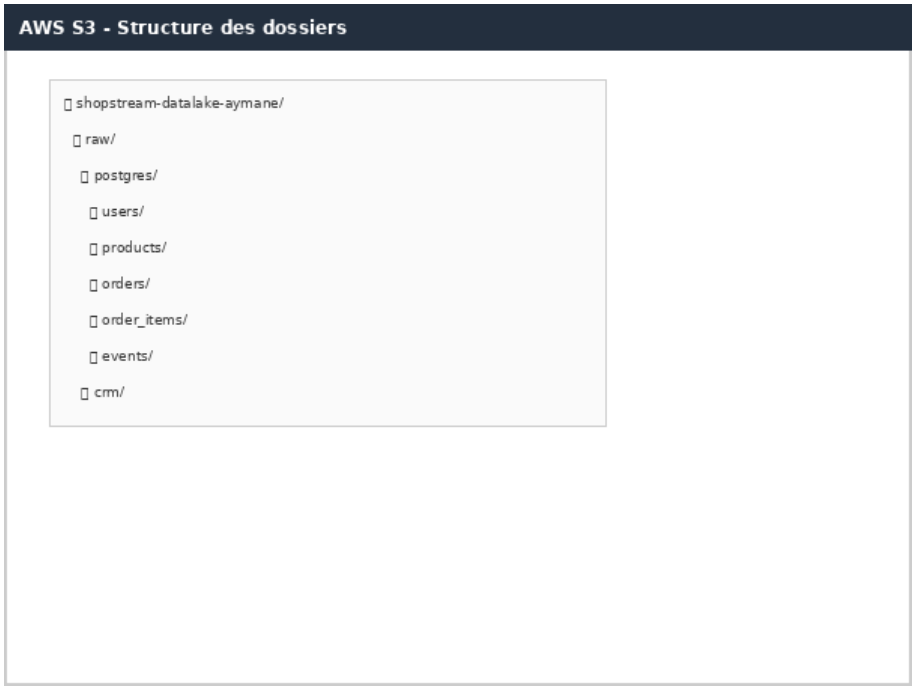

Figure 7 – Structure des dossiers S3

1.6 Configuration IAM et AWS CLI

Un utilisateur IAM (shopstream-s3-user) avec la stratégie AmazonS3FullAccess est créé. Les clés d'accès sont générées puis configurées via AWS CLI pour permettre l'accès programmatique au bucket.

Figure 8 – Utilisateur IAM et clés d'accès

Figure 9 – Configuration AWS CLI

1.7 Export PostgreSQL vers S3

Le script export_to_s3.py extrait les données de PostgreSQL et les charge au format CSV dans le bucket S3, préparant ainsi les données pour l'ingestion dans Snowflake.

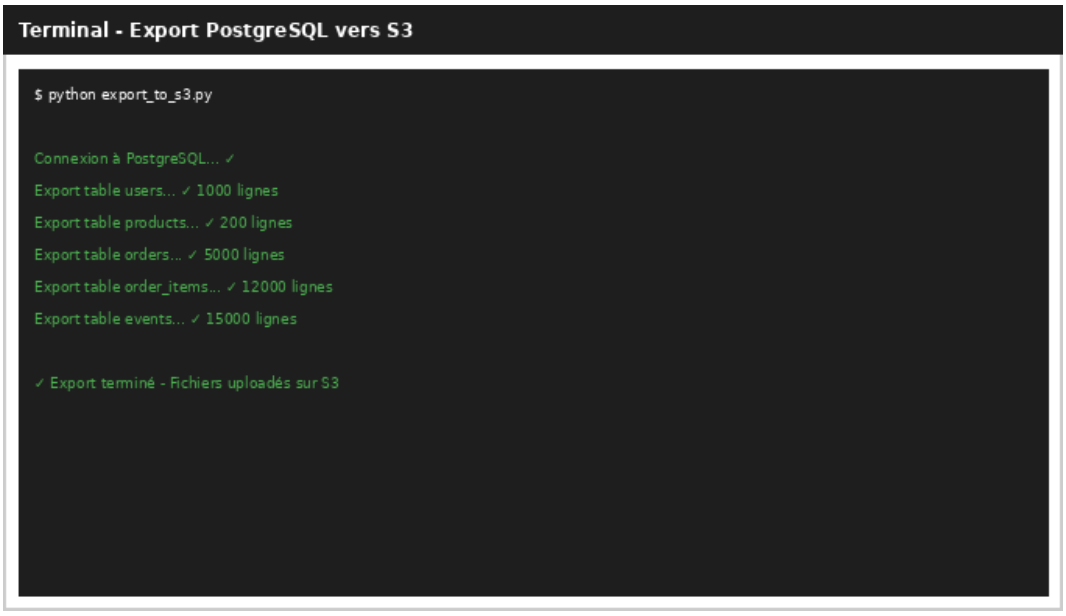

Figure 10 – Export vers S3

1.8 Configuration de Snowflake

Dans Snowflake, quatre schémas sont créés (RAW, STAGING, CORE, MARTS) ainsi que trois Virtual Warehouses (LOADING_WH, TRANSFORM_WH, BI_WH) pour gérer les différentes charges de travail.

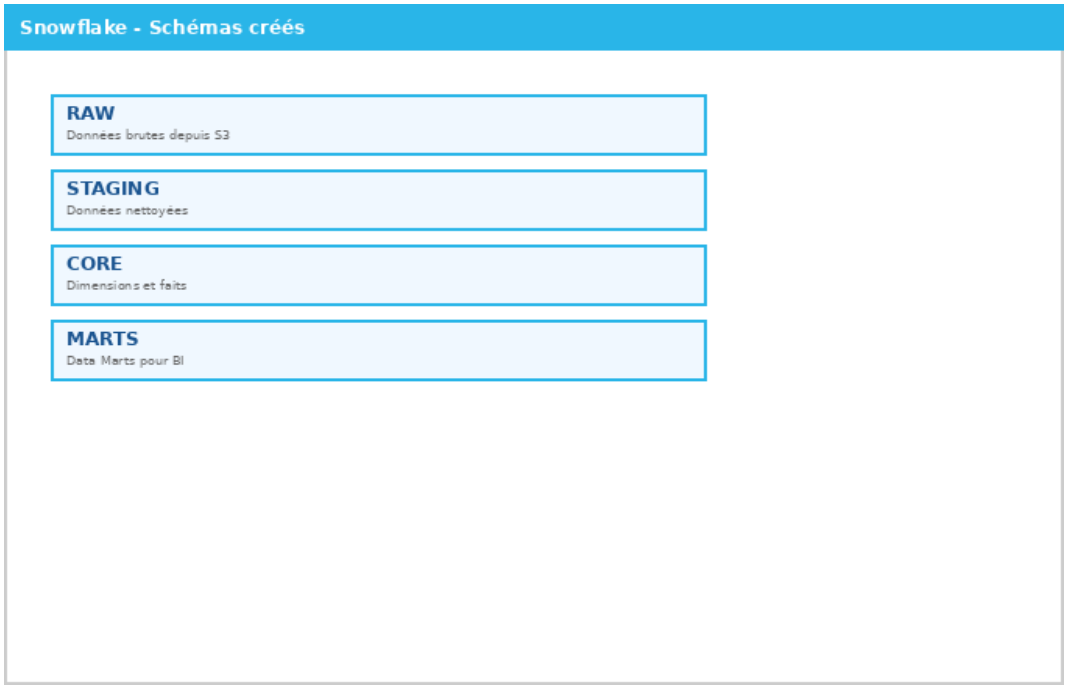

Figure 11 – Schémas Snowflake

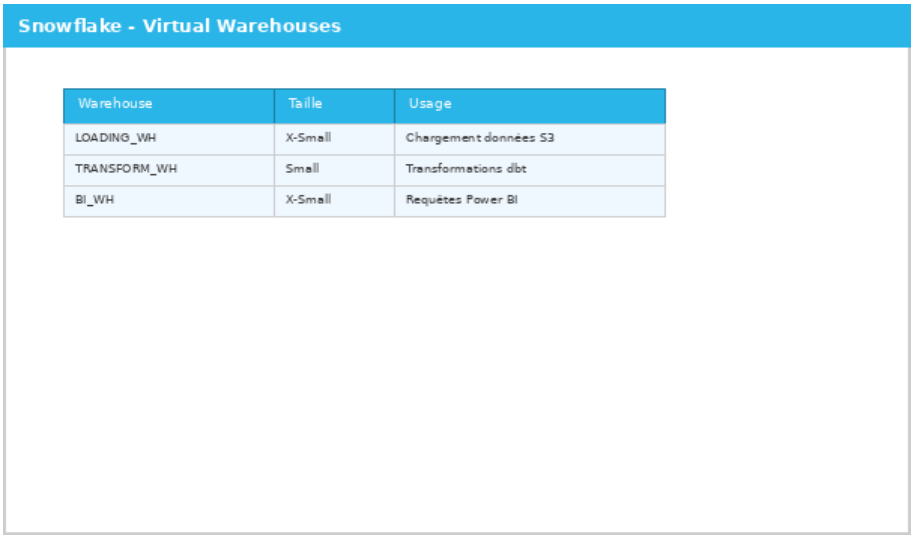

Figure 12 – Virtual Warehouses

1.8.1 Storage Integration et chargement des données

Une Storage Integration permet à Snowflake de se connecter au bucket S3. Un Stage externe pointe vers S3 et les tables STAGING sont créées puis chargées via la commande COPY INTO.

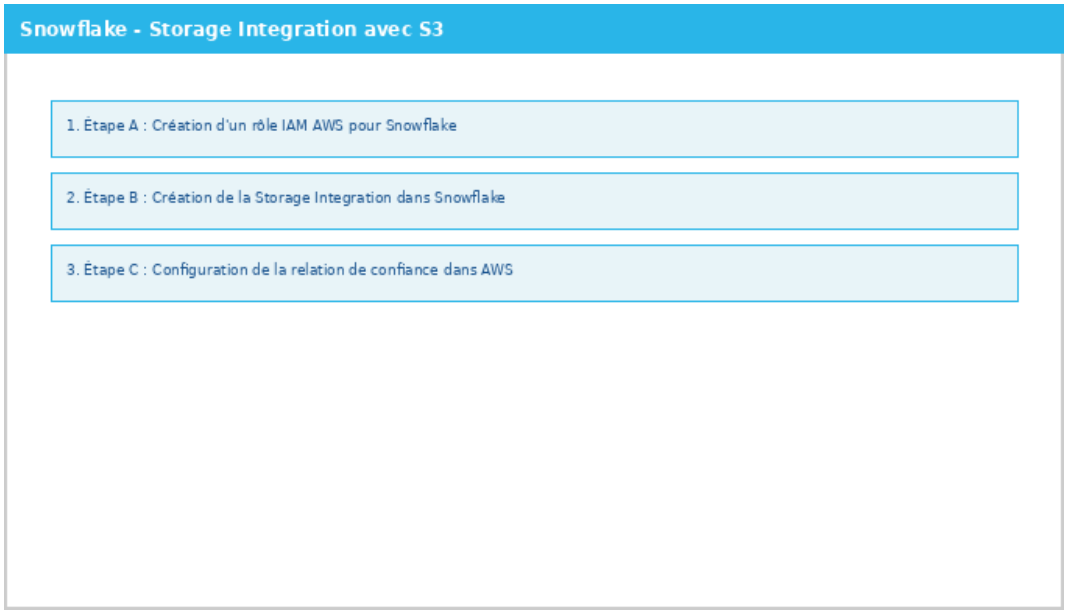

Figure 13 – Storage Integration S3

Figure 14 – Stage externe S3

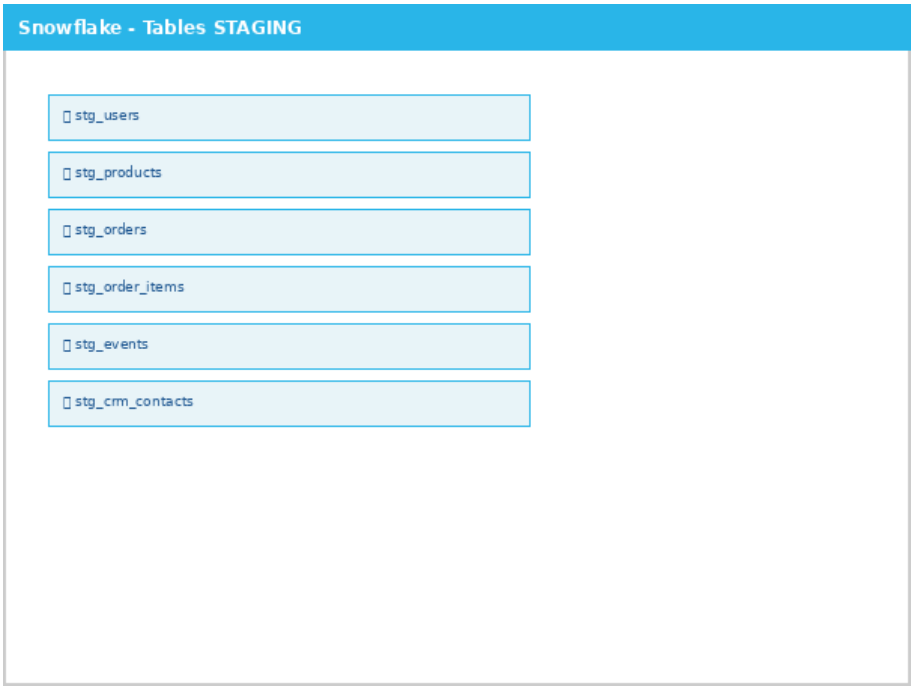

Figure 15 – Tables STAGING

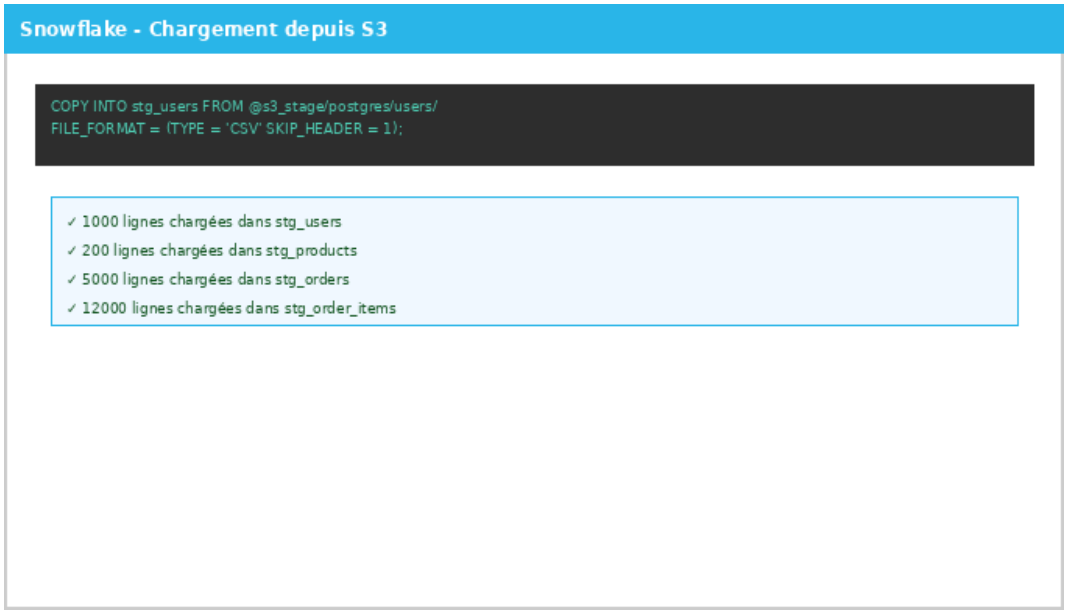

Figure 16 – Chargement données S3

1.9 dbt : Transformations de données

dbt (data build tool) transforme les données STAGING en dimensions, faits et data marts optimisés pour l'analyse. Installation via pip, puis initialisation du projet et configuration de la connexion Snowflake.

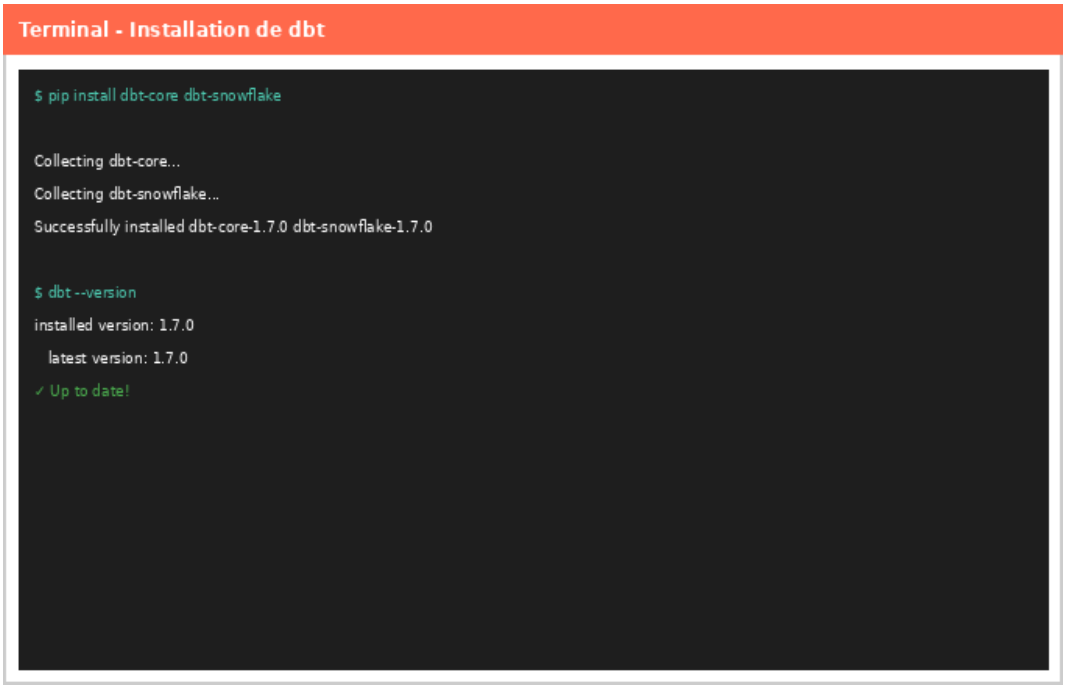

Figure 17 – Installation dbt

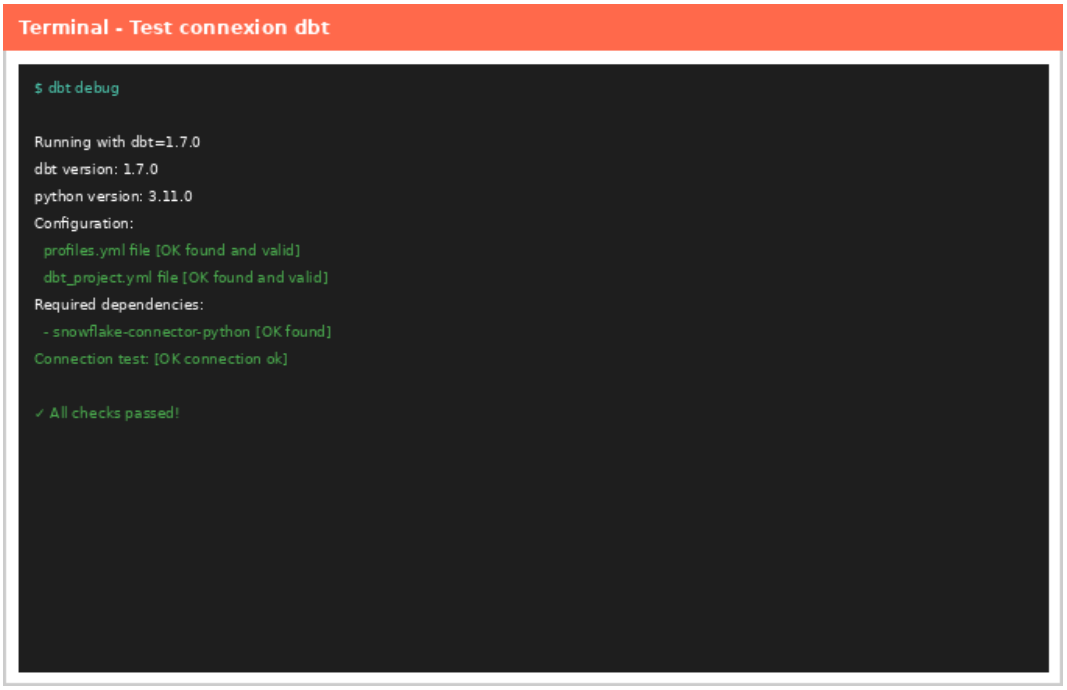

Figure 18 – Test connexion dbt

1.9.1 Exécution des modèles dbt

La commande dbt run compile et exécute tous les modèles dans l'ordre des dépendances, créant les tables stg_*, dim_customers, dim_products, fact_orders et mart_sales_overview dans Snowflake.

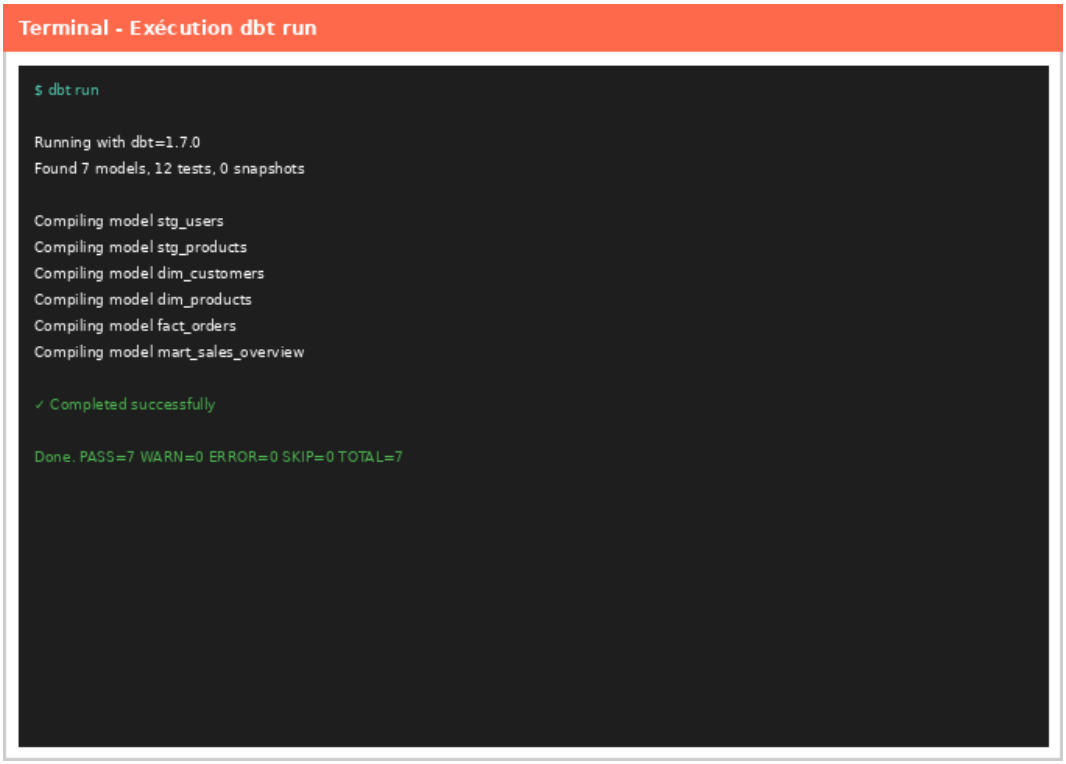

Figure 19 – Exécution dbt run

1.9.2 Tests de qualité et documentation

dbt test vérifie la qualité des données (not_null, unique, relations). dbt docs génère automatiquement la documentation complète du projet avec le lineage graph visualisant les dépendances entre modèles.

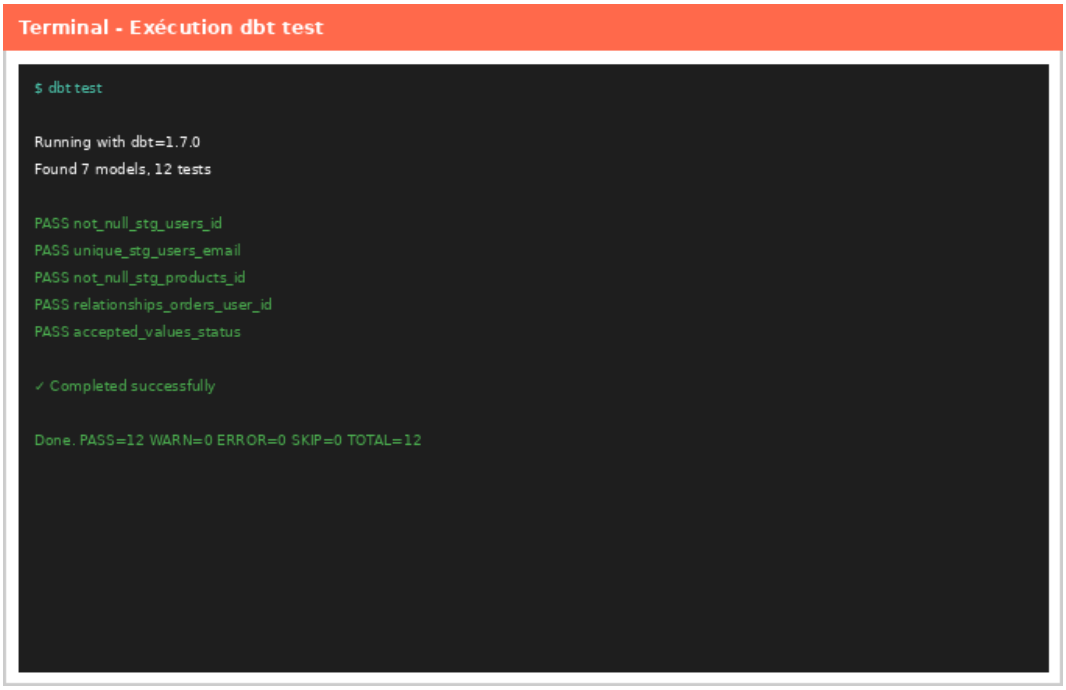

Figure 20 – Tests dbt

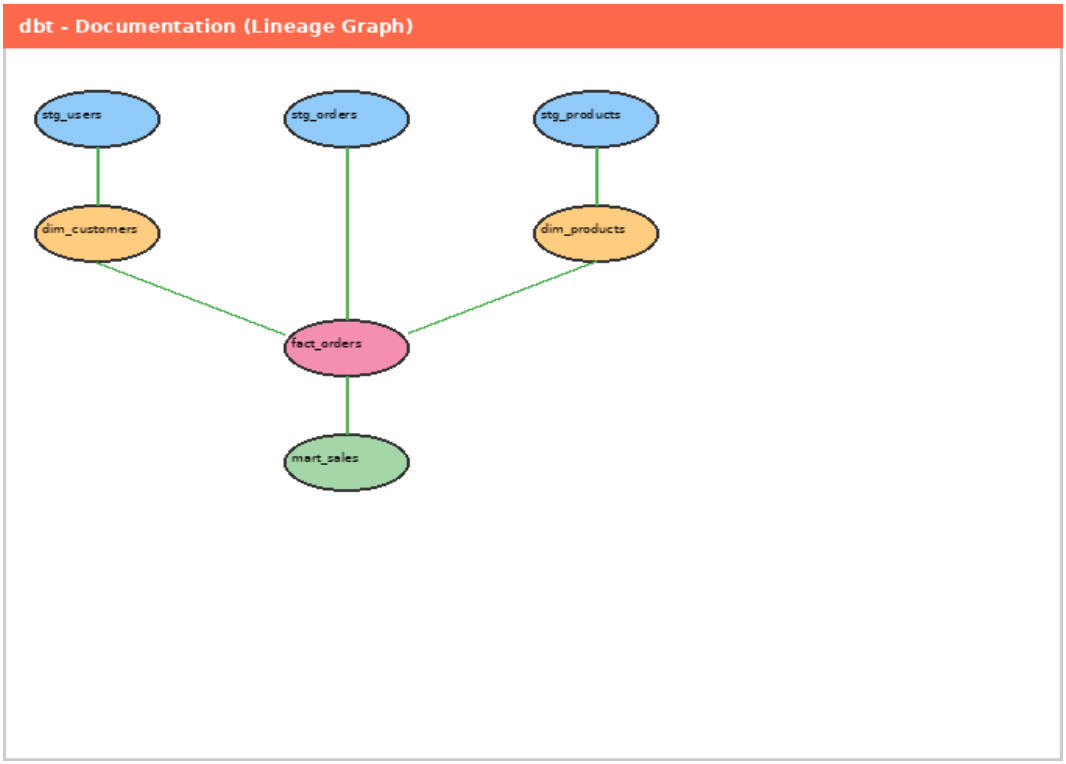

Figure 21 – Documentation dbt (lineage)

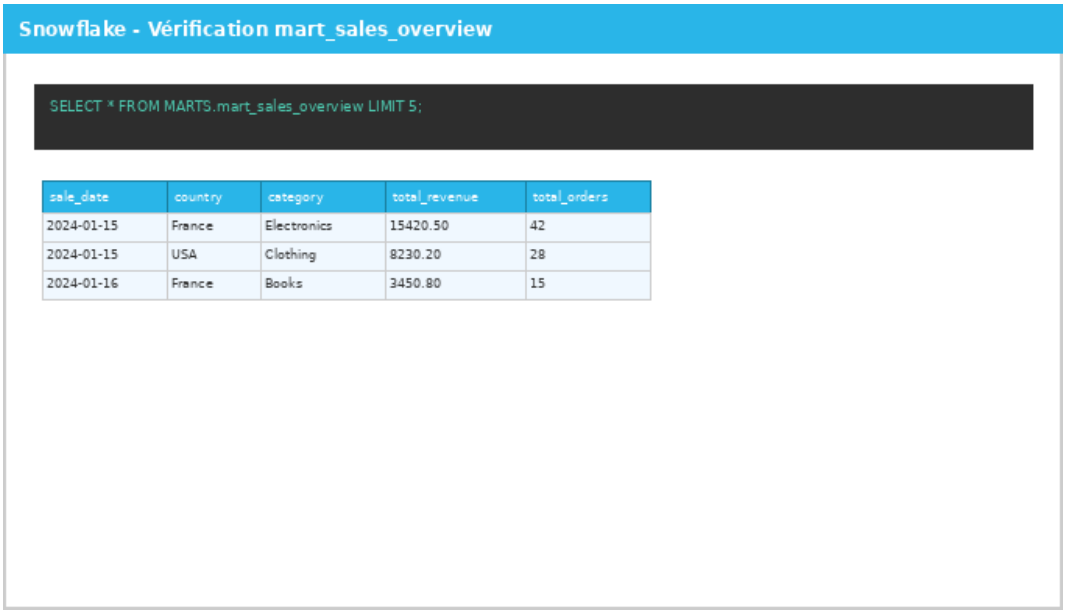

Figure 22 – Vérification dans Snowflake

1.10 Création des Data Marts

Deux data marts supplémentaires sont créés : mart_product_performance (analyse ABC des produits) et mart_customer_ltv (Customer Lifetime Value avec segmentation RFM).

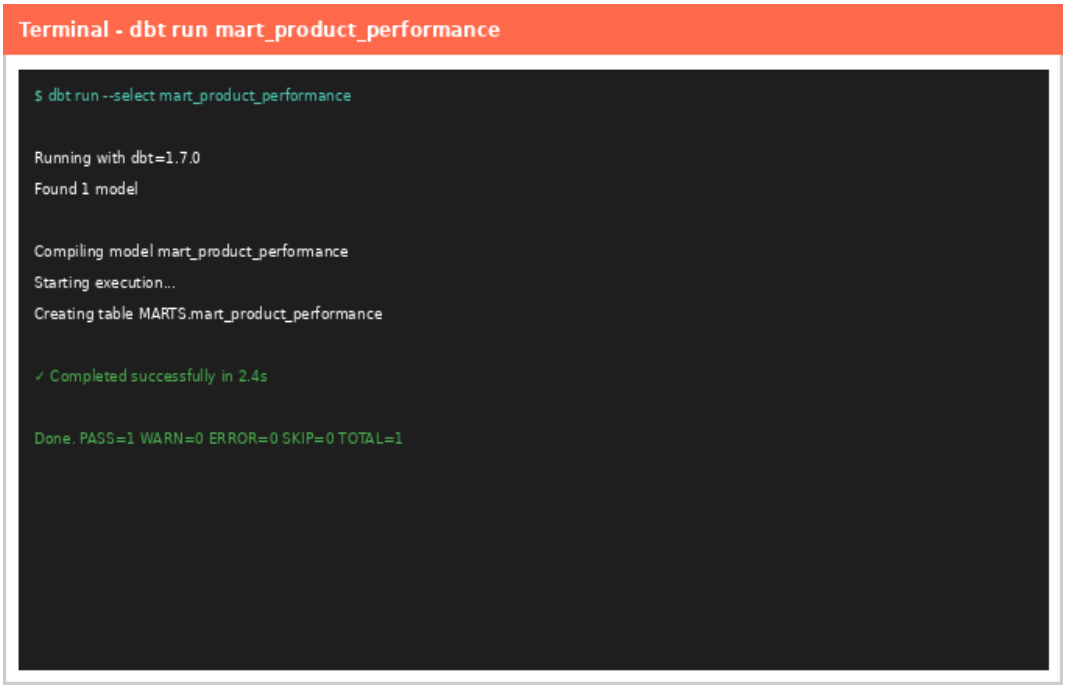

Figure 23 – Data Mart Performance Produits

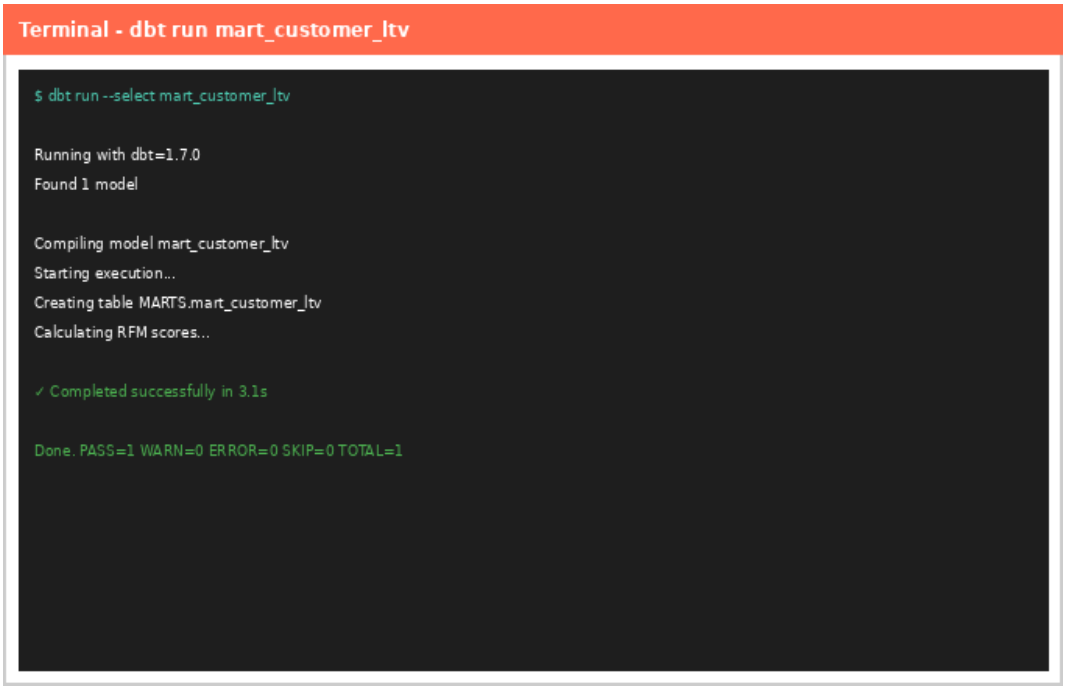

Figure 24 – Data Mart Customer LTV

1.11 Power BI : Visualisation et Dashboards

Power BI se connecte directement à Snowflake via le warehouse BI_WH. Trois dashboards interactifs sont créés : Vue d'ensemble des ventes, Performance produits et Analyse clients.

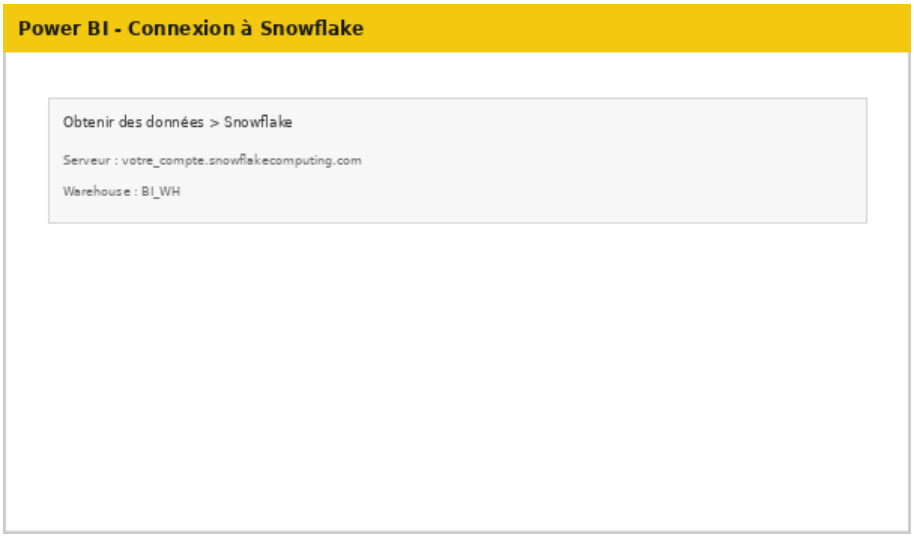

Figure 25 – Connexion Power BI à Snowflake

1.11.1 Dashboard Vue d'ensemble

Cartes KPI (chiffre d'affaires total, nombre de commandes), graphiques d'évolution temporelle du CA, répartition par pays et par catégorie de produits. Mesures DAX créées pour calculer le panier moyen et la croissance mensuelle.

Figure 26 – Carte KPI Chiffre d'affaires

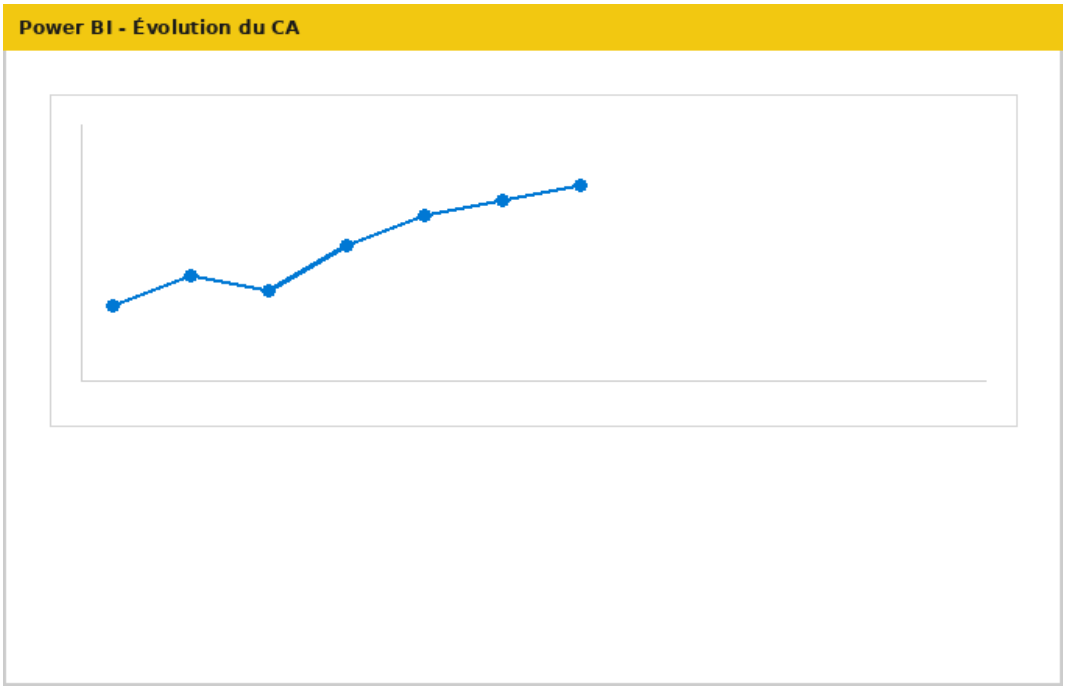

Figure 27 – Graphique évolution CA

1.11.2 Dashboards Produits et Clients

Le dashboard Performance Produits affiche un tableau détaillé avec classification ABC et tiers de performance. Le dashboard Analyse Clients présente la segmentation RFM, la valeur à vie par segment et un tableau des top clients.

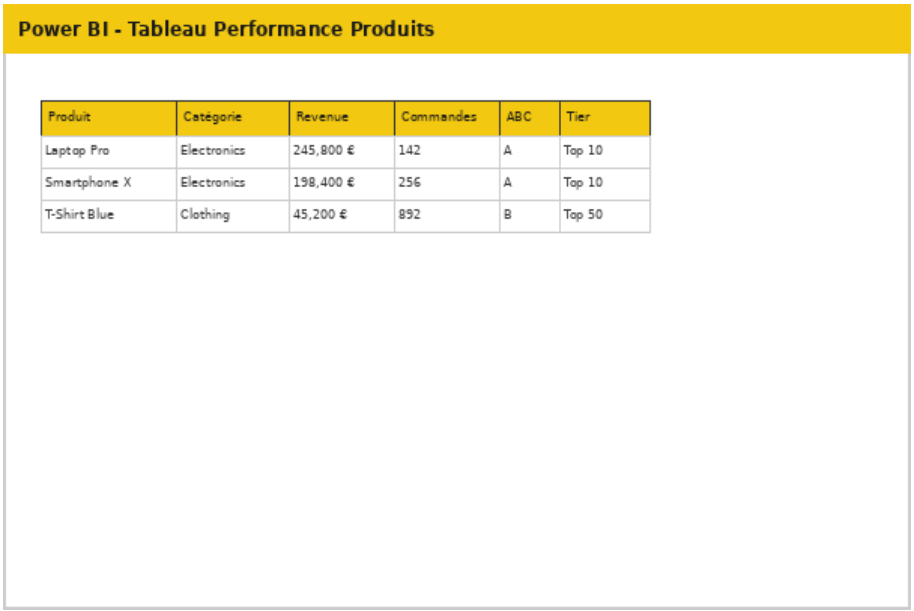

Figure 28 – Tableau Performance Produits

2. Synthèse et conclusions

Ce travail pratique a permis de mettre en œuvre un pipeline de données moderne de bout en bout, intégrant les meilleures pratiques de l'écosystème cloud actuel. Le projet démontre l'orchestration réussie de multiples technologies (PostgreSQL, AWS S3, Snowflake, dbt, Power BI) pour créer une architecture analytique scalable et performante.

Les compétences acquises incluent la création et gestion d'un Data Lake cloud, l'intégration de données depuis diverses sources, la modélisation dimensionnelle dans un Data Warehouse moderne, les transformations SQL déclaratives avec dbt, et la création de dashboards interactifs pour la prise de décision métier.

Cette approche modulaire et cloud-native représente l'état de l'art des architectures de données en entreprise et constitue une base solide pour l'évolution vers des cas d'usage plus complexes (streaming temps réel, Machine Learning, data science à grande échelle).