Sales Analytics Modernization at Globe Smart Electronics

About this project

This project delivers a modern, scalable sales analytics solution built on Microsoft Fabric Lakehouse using the Medallion Architecture. Raw sales data from ADLS Gen2 was ingested into Fabric and transformed into business-ready datasets.

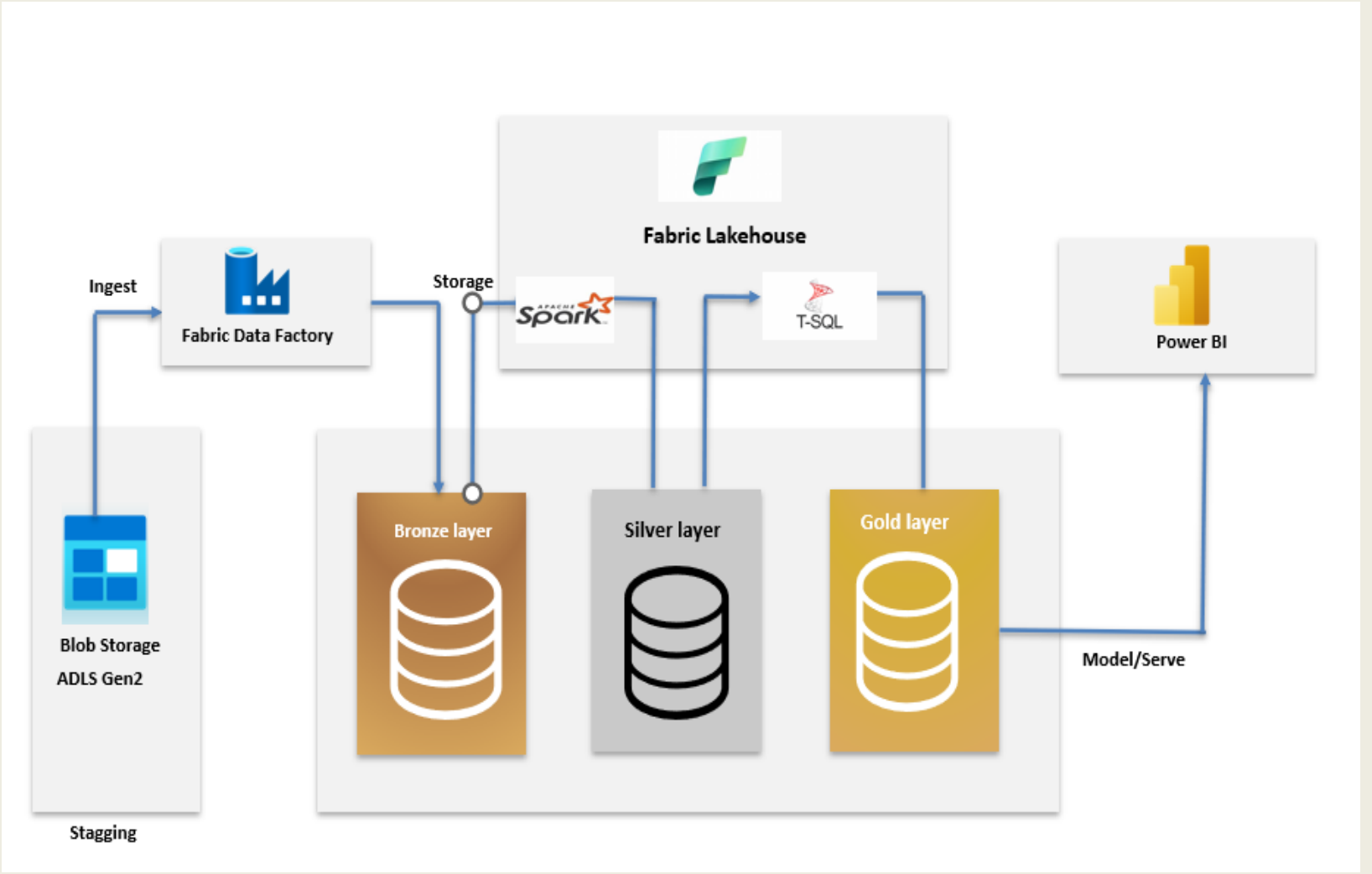
Designed as a replacement for legacy systems, the Fabric Lakehouse now supports critical financial reporting, business intelligence, and advanced analytics, while providing scalability, performance, and robust governance.

Key insights include sales by channel, customer tier profitability, and country-level performance, making the pipeline not just a workflow, but a strategic decision-making tool

Tools and Services

Fabric Data Factory (Pipelines) – Orchestrates ingestion pipelines from ADLS Gen2 into the Fabric Lakehouse Bronze layer.
Azure Data Lake Storage Gen2 (ADLS Gen2) – Used as the staging layer for raw source files before ingestion into Fabric.
Microsoft Fabric Lakehouse (Pyspark + SQL) – Implements Medallion Architecture in fabric Lakehouse (Bronze → Silver → Gold).
Pyspark Notebooks – Data cleaning, deduplication, type standardization, SCD Type 2 logic, and FX normalization.
SQL Views – Star schema modeling in the Gold layer.
Power BI – Consumes curated Gold layer views for interactive dashboards with KPIs, sales analysis, and customer insights.

Project architecture



Challenges

1. Scattered Data Sources – Sales transactions were pulled from ERP systems, customer and country data came from SharePoint files, and salesperson details were managed in warehouse systems. This fragmentation caused manual integration, frequent errors, and no single source of truth

2. No Historical Tracking – Changes in customer tiers and sales events were not tracked historically, making it difficult to analyze trends, customer evolution, and long-term performance.

3. Currency Fragmentation – Sales were recorded in multiple local currencies without a standardized USD conversion, leading to challenges in consolidated reporting and global insights

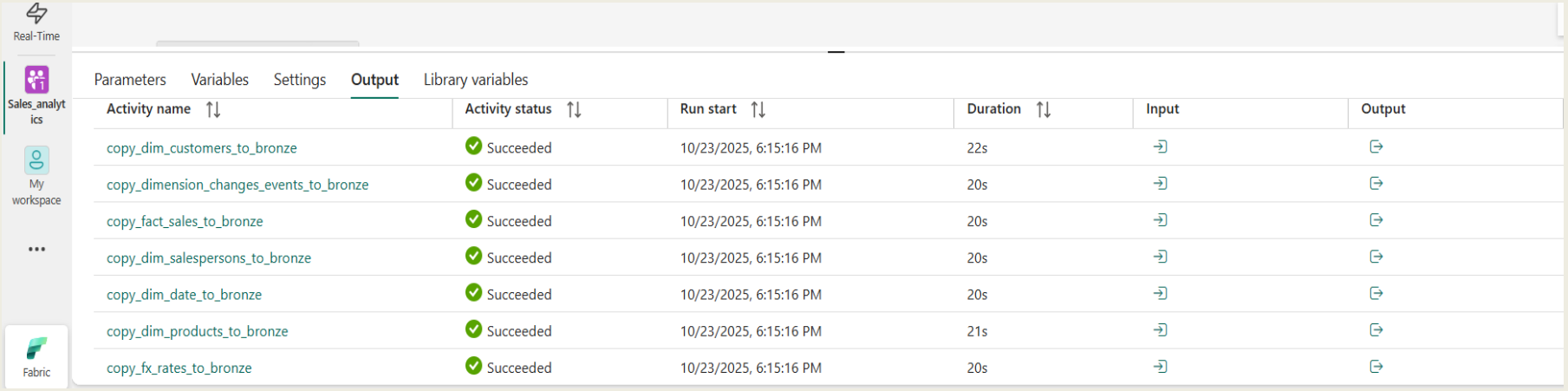
Solution

1. Unified Data Ingestion with Fabric – All source data (ERP sales, SharePoint customer/country files, and warehouse salesperson data) was ingested into ADLS Gen2 and then loaded into the Fabric Lakehouse Bronze layer using Fabric Data Factory pipelines, eliminating manual effort and ensuring a single source of truth.

2. Historical Tracking with SCD Type 2 – Implemented Slowly Changing Dimension Type 2 (SCD2) in the Silver layer using PySpark notebooks to track changes in customer tiers and event history, enabling full historical analysis.

3. Currency Normalization & Star Schema Modeling – Applied FX rate normalization to USD in Silver and designed a Gold layer star schema with SQL views, supporting fast, accurate, and consolidated financial reporting in Power BI.

Bronze layer Ingestion



Activity name	Activity status	Run start	Duration	Input	Output
copy_dim_customers_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	22s		
copy_dimension_changes_events_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	20s		
copy_fact_sales_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	20s		
copy_dim_salespersons_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	20s		
copy_dim_date_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	20s		
copy_dim_products_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	21s		
copy_fx_rates_to_bronze	Succeeded	10/23/2025, 6:15:16 PM	20s		

Pipeline successfully ingested raw sales data into Bronze Layer from ADLS Gen2 into Fabric Lakehouse

This screenshot shows the successful execution of the Fabric Data Factory pipeline that ingests raw sales data from Azure Data Lake Storage Gen2 (ADLS Gen2) into the Fabric Lakehouse Bronze layer.

Each activity corresponds to copying customers, sales, products, salespersons, date, FX rates, and event history into the Bronze layer.

The succeeded status confirms that all ingestion jobs ran without errors, ensuring the raw data is available for further processing.

The Bronze layer now serves as the single source of truth for all raw sales data, eliminating manual file handling and reducing errors.

Silver layer Transformation

SCD TYPE 2 Implementation

```python
#Bronze → Silver (SCD2) for Events

from pyspark.sql import functions as F
from pyspark.sql.window import Window

# 1) Read Bronze
ch = spark.table("bronze_dimension_changes_events")

# 2) Standardize values
ch = (ch
      .withColumn("entity", F.trim(F.col("entity")))
      .withColumn("business_key", F.trim(F.col("business_key")))
      .withColumn("attribute", F.lower(F.trim(F.col("attribute"))))
      .withColumn("effective_ts", F.to_timestamp("effective_ts"))
)

# 3) Pivot: turn attribute rows into actual columns
pivoted = (ch
    .groupBy("entity","business_key","effective_ts")
    .pivot("attribute")
    .agg(F.first("new_value"))
)

# 4) Build SCD2 validity ranges
w = Window.partitionBy("entity","business_key").orderBy("effective_ts")

silver = (pivoted
    .withColumn("effective_start", F.col("effective_ts"))
    .withColumn("effective_end", F.lead("effective_ts").over(w))
    .withColumn("is_current", F.when(F.col("effective_end").isNull(), True).otherwise(False))
    .drop("effective_ts")
)
```

Silver layer Transformation

```
# 5) Write Silver table
(silver.write
   .mode("overwrite")
   .format("delta")
   .option("overwriteSchema","true")
   .saveAsTable("silver_dim_events"))

print("✅ silver_dim_events (SCD2):", spark.table("silver_dim_events").count())

count_rows = spark.table("silver_dim_events").count()

print("SCD Type 2 Processing Complete for Events")
print("Target Table: silver_dim_events")
print(f" Total Records Loaded: {count_rows}")
display(spark.table("silver_dim_events").limit(20))
```

[1] ✓ 1 min 8 sec - Session ready in 14 sec 706 ms. Command executed in 54 sec 111 ms by fabricuser1 on 7:13:20 PM, 10/23/25

> Spark jobs (20 of 20 succeeded) Resources Log

```
✅ silver_dim_events (SCD2): 187
SCD Type 2 Processing Complete for Events
Target Table: silver_dim_events
 Total Records Loaded: 187
```

Table | + New chart

Table view

	entity	business_key	category	reporting_region	tier	effective_start	effective_end	is_current
1	customer	CUST00002	NULL	NULL	Bronze	2025-09-27 00:00:00	NULL	true
2	customer	CUST00004	NULL	NULL	Bronze	2025-05-15 00:00:00	NULL	true
3	customer	CUST00005	NULL	NULL	Bronze	2025-04-30 00:00:00	NULL	true

Silver Layer Transformation – SCD Type 2 implemented successfully to capture full historical changes in customer and event data

This step applies Slowly Changing Dimension (SCD Type 2) logic to capture historical changes in customer and event data.

Each change in attributes (e.g., customer tier, region) is assigned an effective start and end date, with a flag (is_current) to mark the active record.

Ensures full historical tracking of customer journeys and business events.
Output is written as Delta tables in Fabric Lakehouse, providing a clean, standardized Silver dataset

Gold layer Aggregation

```sql
CREATE VIEW gold_fact_sales AS
SELECT
    f.*,                                -- all columns from silver_fact_sales
    fx.rate                     AS fx_rate_usd,
    (f.gross_amount * fx.rate)  AS gross_amount_usd
FROM silver_fact_sales f
JOIN silver_fx_rates fx
      ON fx.currency = f.currency
     AND fx.date     = CAST(f.order_date AS date)   -- align types
WHERE f.order_date >= '2024-01-01';
```

[24] ✓ - Command executed using sales_lakehouse in 2 sec by fabricuser1 on 11:44:42 PM, 10/18/25 T-SQL

```sql
SELECT TOP 10 *
FROM gold_fact_sales;
```

[25] ✓ - Command executed using sales_lakehouse in 2 sec by fabricuser1 on 11:45:07 PM, 10/18/25 T-SQL

Messages

Table | New chart | 17 columns, 10 rows

Table view | Download | Save as table | Open in Excel | Filter by keyword | Inspect

	order_id	order_date	customer_id	product_id	salesperson_id	channel	currency	warehouse_id	qty	unit_price	discount_pct	gross_amount
1	SO308740	2024-08-04	CUST00340	PRD00106	SP0032	ONLINE	AED	WH3	18	129.58	0.09	2332.44
2	SO317735	2025-03-11	CUST00348	PRD00107	SP0043	ONLINE	USD	WH3	11	80.77	0.04	888.47

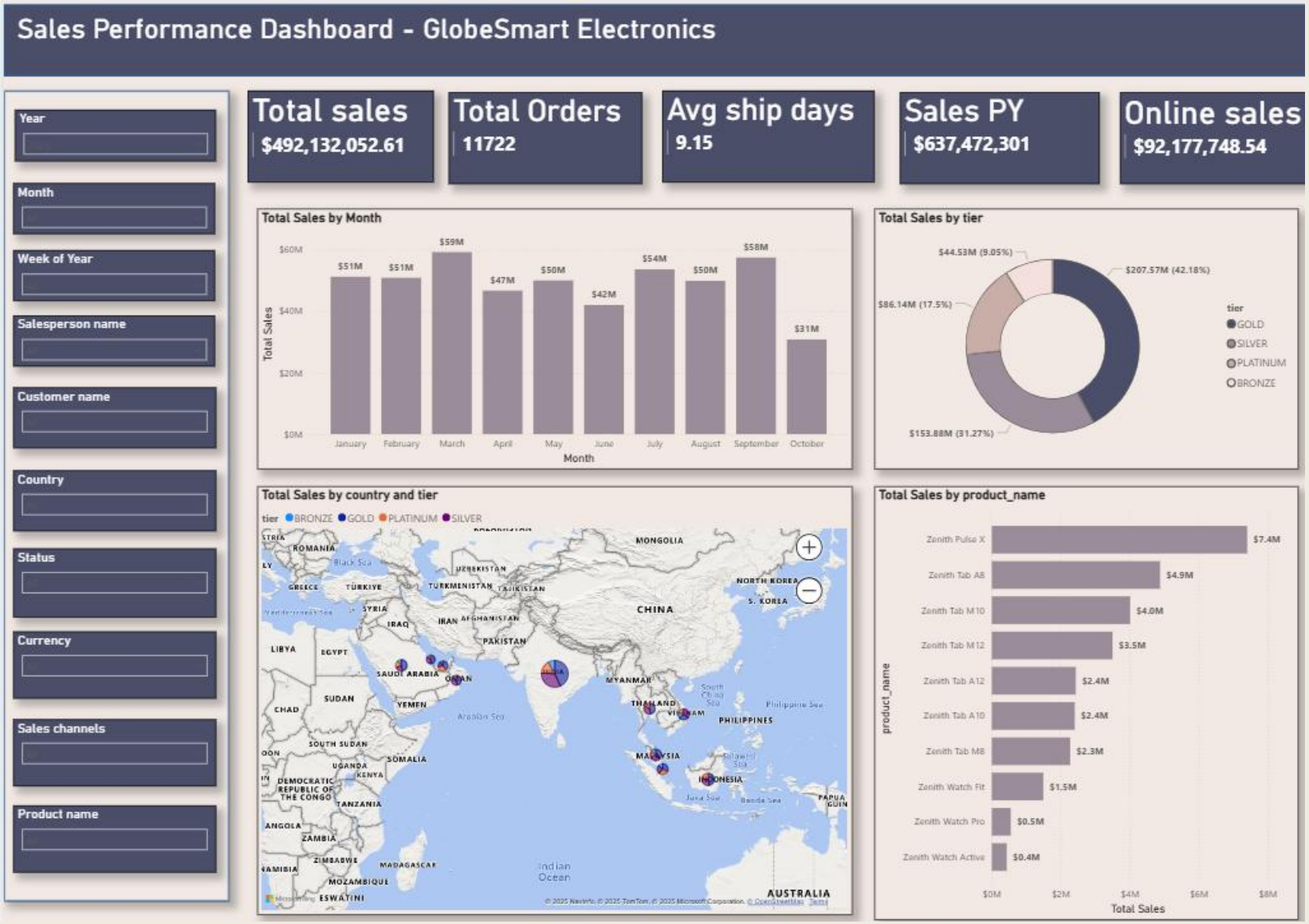
The Gold layer consolidates all fact and dimension data into a business-ready star schema.

•Exchange rates are joined to calculate gross sales in USD, enabling global financial reporting.

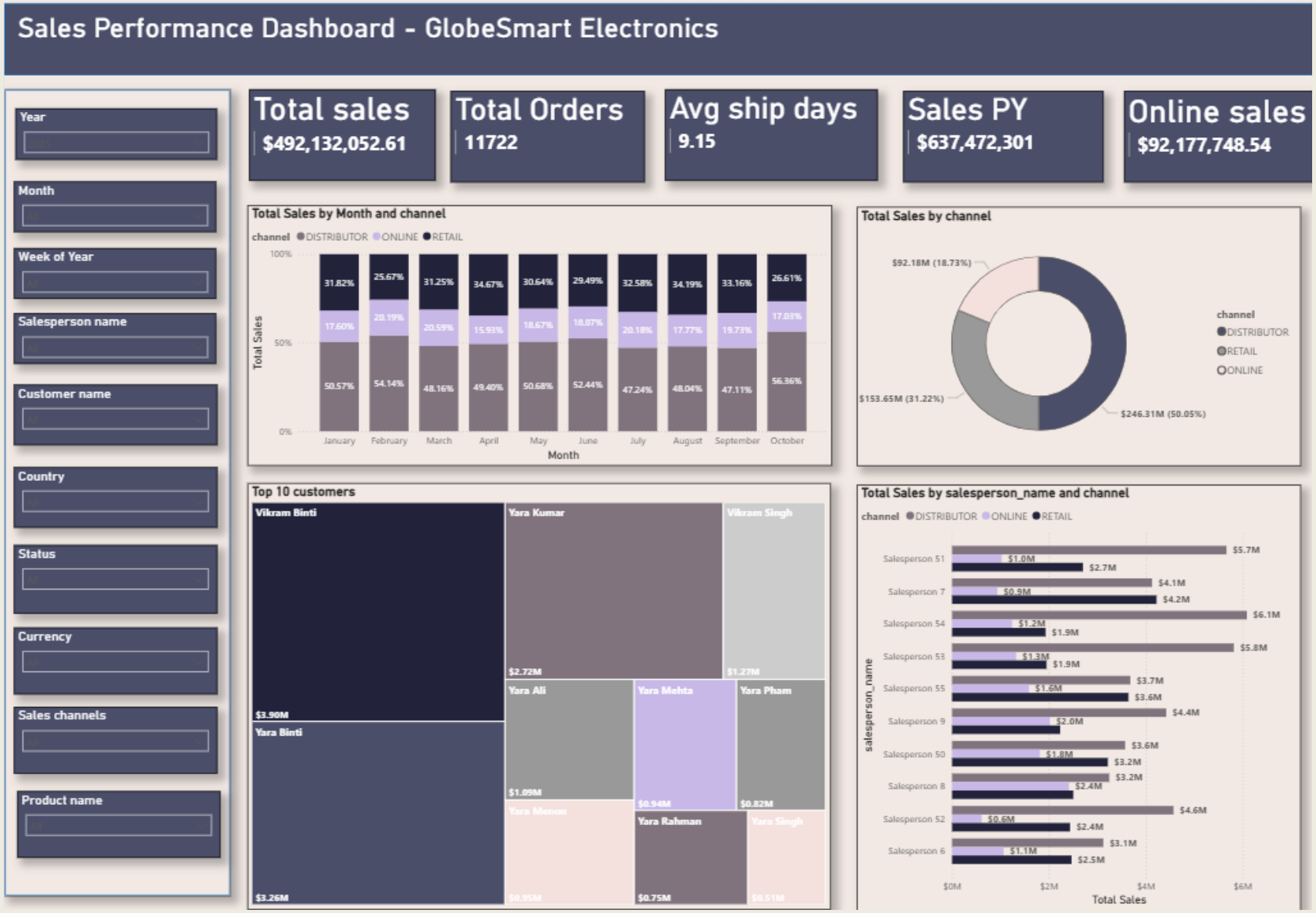
•Dimension tables (Customers, Products, Salespersons, Events) are integrated to enrich the fact table with attributes like customer tier, product category, and reporting region.

•The output is modeled as SQL Views on top of Fabric Lakehouse Delta tables, providing a clean and standardized datasets

Power BI dashboard



Power BI dashboard



Business Impact

•**Automated Ingestion** – Sales, customer, product, and FX data flows directly from ADLS Gen2 into Fabric Lakehouse, reducing manual file handling by ~60%.

•**Historical Tracking** – With SCD Type 2 in Silver, customer tier and event changes are fully tracked, enabling year-over-year trend analysis.

•**Unified Global Reporting** – Currency normalization into USD enabled consistent cross-country comparisons, improving financial accuracy.

•**Faster Analytics** – Gold star schema views improved query performance, reducing reporting time from minutes to seconds in Power BI

•**Actionable KPIs** – Executives can now monitor Total Sales, Orders, SLA (ship days), Customer Tier profitability, and Geography-level sales in real time

•**Future Ready** – Fabric Lakehouse architecture allows easy scaling to new regions, data sources, or reporting needs without rework.