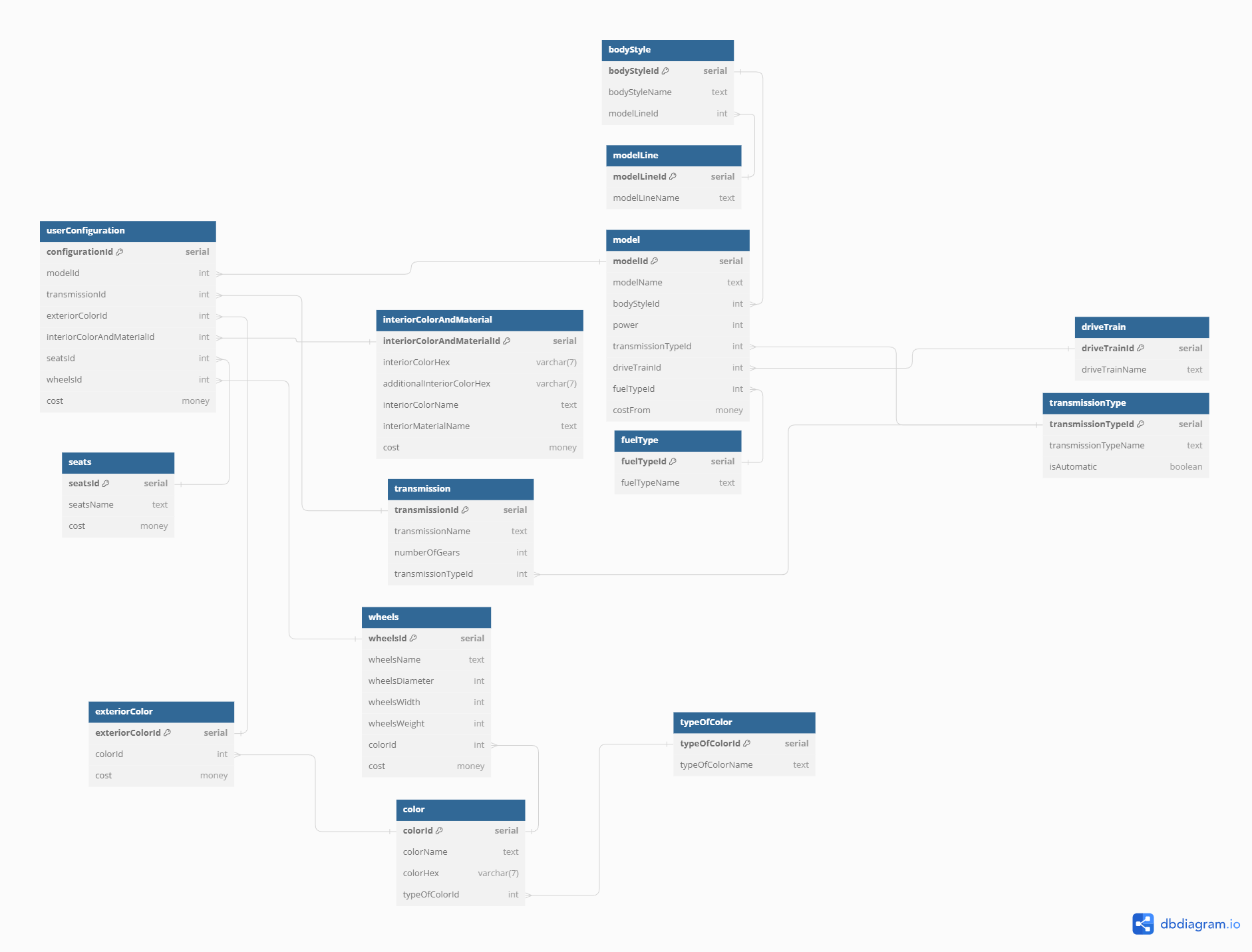
Porsche Car Configuration System Documentation

# **1. Project Overview**

This project implements a Porsche car configuration system, allowing users to select and configure different models of Porsche cars. The system includes an OLTP database for transaction processing, an OLAP database for analytics, and a Power BI report with visualization and analysis.

# **2. Database Schema Description**

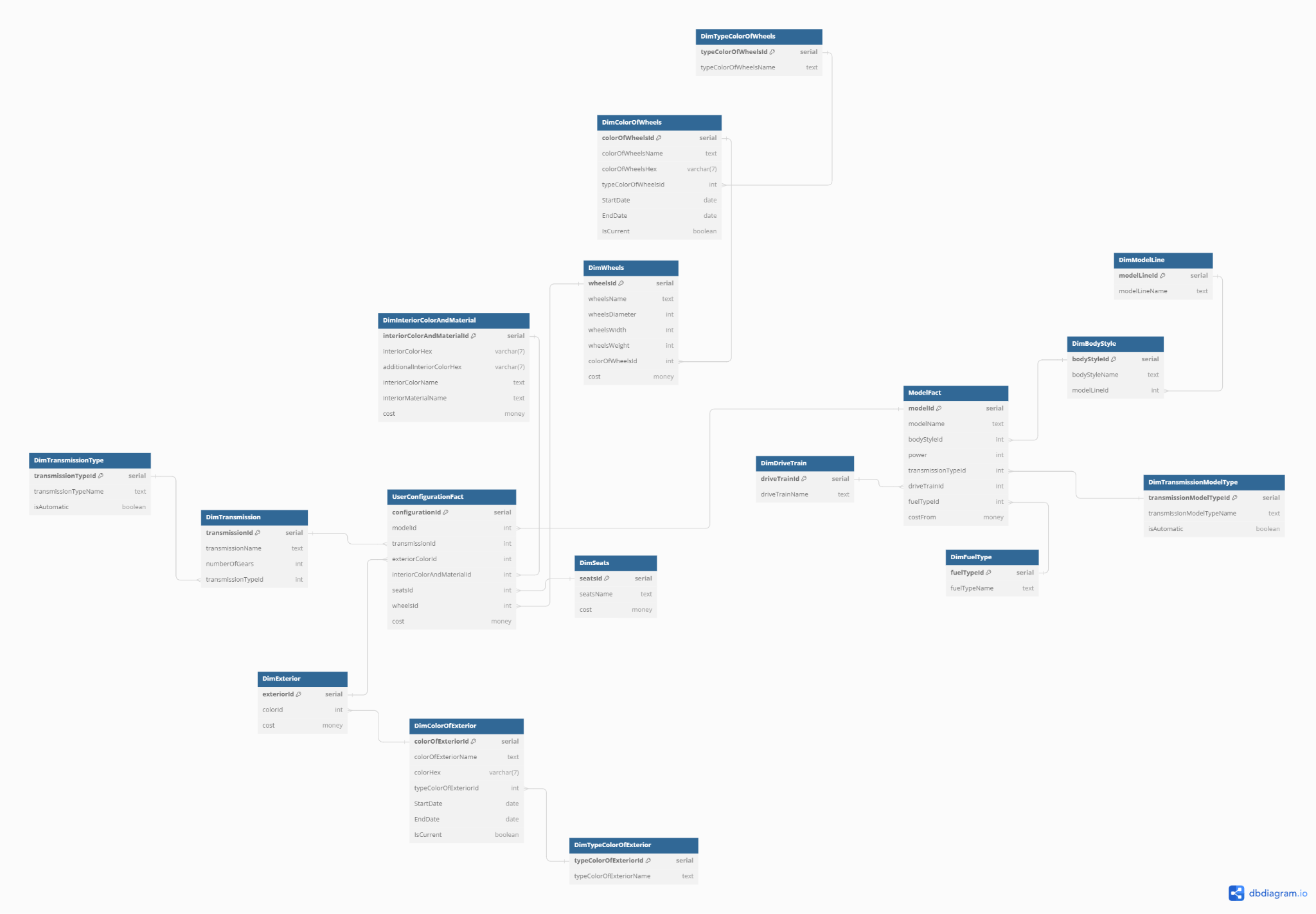
## **2.1 OLTP Schema**

The OLTP schema includes tables such as model, color, transmission, bodyStyle, and userConfiguration, which are interconnected to manage car configurations efficiently.  
  


Link to diagram for more detailed inspection - https://dbdiagram.io/d/Porshe-665f86e1b65d93387982c70a

## **2.2 OLAP Schema**

The OLAP schema includes dimension tables like DimColor, DimTransmission, and DimBodyStyle, and fact tables such as FactUserConfiguration. This setup supports efficient querying and data aggregation for reporting purposes.



Link to diagram for more detailed inspection - https://dbdiagram.io/d/DimPorsche-6660f28f8f6e135d4a66e3cc

# **3.** Loading Data csv2oltp

**Purpose:** The load\_data\_from\_csv function is designed to facilitate the batch loading of configuration data for various components of Porsche cars from CSV files into the OLTP system. This function is essential for initializing and updating the database with new or modified configuration options, such as seats, drive trains, and colors.

**Function Signature:**

CREATE OR REPLACE FUNCTION load\_data\_from\_csv(

seats\_file\_path TEXT,

drive\_train\_file\_path TEXT,

transmission\_type\_file\_path TEXT,

transmission\_file\_path TEXT,

fuel\_type\_file\_path TEXT,

type\_of\_color\_file\_path TEXT,

color\_file\_path TEXT,

exterior\_color\_file\_path TEXT,

wheels\_file\_path TEXT,

interior\_color\_and\_material\_file\_path TEXT,

model\_line\_file\_path TEXT,

body\_style\_file\_path TEXT,

model\_file\_path TEXT,

user\_configuration\_file\_path TEXT

) RETURNS VOID

**Parameters:**

* \*\_file\_path: Paths to the CSV files for each type of car configuration data. These parameters allow the function to adapt to data stored in different locations.

**Main Operations:**

1. **Temporary Tables Creation:**
   * The function begins by creating a series of temporary tables for each type of configuration data. These tables are designed to temporarily hold the data imported from the provided CSV files, facilitating data manipulation before it is inserted into the main tables.
2. **Data Loading:**
   * Data is loaded from CSV files into the corresponding temporary tables using the PostgreSQL COPY command. This command is executed dynamically using the format function to accommodate various file paths and ensure proper data format handling.
3. **Data Insertion into Main Tables:**
   * After loading the data, it is then inserted into the permanent tables in the database. This step involves selecting data from temporary tables and handling potential conflicts (e.g., duplicate entries) using the ON CONFLICT clause, which prevents errors during data insertion.
4. **Temporary Tables Cleanup:**
   * Once the data has been successfully transferred to the main tables, all temporary tables are dropped to free up resources and maintain database cleanliness.

**Usage:**

* The function is called with paths to the respective CSV files as arguments. This modular approach allows for flexible data updates to any of the configuration parameters by specifying only the paths to the changed data files.

**Example Call**

SELECT load\_data\_from\_csv(

'path/to/seats.csv',

'path/to/driveTrain.csv',

'path/to/transmissionType.csv',

...

);

This detailed documentation provides an understanding of how the load\_data\_from\_csv function operates within the system, ensuring proper data management and system updates.

# **4.** Oltp2Olap

**Purpose:** The script is designed to synchronize data from an OLTP database (Porsche) to an OLAP schema (DimPorsche), enabling seamless data flow for reporting and analytical purposes. This synchronization process is crucial for maintaining up-to-date and consistent data across different database systems.

**Operation:**

1. **Extension Loading:**
   * The script starts by ensuring the dblink extension is available, which is essential for connecting and querying the OLTP database from within the OLAP database system.
2. **Data Transfer Process:**
   * The script employs a DO block containing PL/pgSQL code to perform the data transfer.
   * For each type of data (e.g., modelLine, bodyStyle, transmissionType, etc.), the script:
     + Connects to the OLTP database using dblink.
     + Fetches data using a SELECT query.
     + Iterates over the fetched records and inserts them into the corresponding OLAP dimension tables only if they do not already exist.
3. **Error Handling:**
   * The script checks for the existence of each record before insertion to prevent duplicate entries, which could lead to data integrity issues.

**Key Components:**

* source\_record: A variable used to store each row fetched from the OLTP database.
* dblink: Utilized to execute queries on a remote database and fetch results directly into the local session.
* LOOP and conditional checks: Ensure that only new or updated records are transferred to maintain database efficiency and accuracy.

**Usage:**

* This script is executed whenever there is a need to update the OLAP database with changes from the OLTP system. It can be scheduled as a batch process or triggered by specific events, depending on the system's requirements.

**Example of Data Transfer Operation:**

-- Example of transferring modelLine data

FOR source\_record IN SELECT \* FROM dblink('dbname=Porsche port=5432 user=postgres password=password', 'SELECT \* FROM modelLine')

AS t1(modelLineId int, modelLineName text)

LOOP

IF NOT EXISTS (SELECT 1 FROM DimModelLine WHERE modelLineId = source\_record.modelLineId) THEN

INSERT INTO DimModelLine VALUES (source\_record.modelLineId, source\_record.modelLineName);

END IF;

END LOOP;

**Advantages:**

* The script automates the process of data transfer, reducing manual intervention and the potential for errors.
* Using dblink for this process allows for real-time or near-real-time data updates, which is essential for dynamic reporting environments.

**Considerations:**

* Ensure that the PostgreSQL service has the appropriate permissions and that dblink is correctly configured to connect to the OLTP database.
* Regular monitoring and logging of the data transfer process are recommended to detect and address any issues promptly.

# **6. Analysis and Reporting**

**Report Overview:** This Power BI report provides a comprehensive analysis of Porsche car configurations, focusing on various aspects like model popularity, drive train distribution, and the relationship between vehicle power and fuel type costs. It aims to assist in understanding market trends, consumer preferences, and strategic decision-making for product offerings.

**Visual Components:**

1. **Volume of Configurations by Model:**
   * **Type:** Bar Chart
   * **Description:** This visual displays the total number of configurations per Porsche model. It provides a clear comparison, highlighting the most popular models based on configuration volume. The 911 GT3 RS appears to be the most popular, significantly outperforming others.
   * **Data Fields:**
     + X-Axis: Model
     + Y-Axis: Sum of Configurations
2. **Volume of Models by Drive Train:**
   * **Type:** Pie Chart
   * **Description:** This chart shows the proportion of models divided by drive train type, offering insights into the preferred drive train options (Rear-Wheel Drive vs. All-Wheel Drive). It is almost evenly split, indicating a balanced preference in drive train choices among Porsche models.
   * **Data Fields:**
     + Legend: Drive Train
     + Values: Count of Model Names
3. **Volume of Fuel by Power:**
   * **Type:** Bubble Chart
   * **Description:** Analyzes the cost associated with different fuel types (Electric, Hybrid, Petrol) against the total power output of the models. Larger bubbles indicate higher numbers of configurations, providing insights into how fuel type preferences correlate with vehicle power.
   * **Data Fields:**
     + X-Axis: Sum of Power
     + Y-Axis: Cost
     + Bubble Size: Sum of Configurations
     + Bubble Color: Fuel Type

**Data Insights:**

* The distribution of configurations by model and the detailed breakdown by drive train type help identify which models and features are preferred in different market segments.
* The cost analysis by fuel type and power provides valuable information for pricing strategies and could influence future product development focusing on efficiency and performance.