Customer360 Project Report

Objective

The objective of this project was to create a Customer360 view for an online retailer by integrating data from multiple tables. This unified view provides detailed insights into customer conversions, order history, and cumulative revenue, aiding in a deeper understanding of customer behavior and informed business decisions.

Method

To achieve this, we utilized several tables from the mmai_db database: fact_tables.orders, fact_tables.conversions, dimensions.date_dimension, dimensions.product_dimension, and dimensions.customer_dimension. These tables were essential in constructing a unified and detailed view of each customer's activity.

The first step in this project was to create a new schema named customer360. This schema was designed to store the final Customer360 view and any intermediate views required for data integration.

Next, we created a CTE named CustomerConversionData to capture static customer conversion details. This included conversion type, date, week, and channel. The use of SQL functions like ROW_NUMBER() and LEAD() allowed us to count conversions and find the next conversion week, ensuring no overlap in conversion periods.

Another CTE named FirstOrderPlaced was created to get the first order details relating to each conversion. The conversion events were linked to the first purchasing behavior of the customer by aggregating the first order data, including the date, week, and total paid. The FIRST_VALUE() function ensured accurate retrieval of first-order details.

Next, another CTE called OrderHistory was created to aggregate weekly order data. This was used to capture metrics such as the number of orders placed, total before discounts, total discounts, and total paid in each week. A cross-join between the date dimension and customer dimension tables ensured that weeks without orders were accounted for in the analysis. Conditions like order_week < next_conversion_week OR next_conversion_week IS NULL prevented overlaps in revenue calculations.

A final CTE named Customer360_CTE combined all previous CTEs to generate rows from the conversion week to the next. Window functions were used to calculate cumulative revenue for both the conversion period and lifetime revenue. This ensured a comprehensive understanding of customer purchasing behavior over time.

The data from the Customer360_CTE was then ordered by customer_id, conversion_number, and week_counter to create the final Customer360 view. This final selection ensured that the data was presented in a logical and useful order, facilitating easy analysis and interpretation. Finally, the table was saved into the customer360 schema in our personal database.

Challenges Faced

We encountered challenges in accurately capturing first-order details and avoiding overlaps in conversion periods. By leveraging FIRST_VALUE() and carefully structuring conditions for conversion periods, these challenges were addressed effectively. Ensuring proper joins between fact and dimension tables required attention to data integrity, and optimizations like indexing improved performance for handling large datasets. Finally, originality was emphasized through unique query structures and naming conventions.

Conclusion

The Customer360 view effectively integrates data to provide a comprehensive overview of customer behavior. Key features include detailed conversion tracking, weekly order history, and cumulative revenue metrics. This project demonstrates the use of SQL for complex data integration, resulting in a valuable tool for understanding customer behavior and supporting business decisions.