[A24] Data Warehousing & ETL

Student information

- Student: Carlos Jesus Caro
- Email: carlos.jesus-caro@edu.dsti.institute
- Student type: SPOC

Project

- GitHub repository: https://github.com/carlosjesuscaro/masters_de_dwh_olist
- File: OLIST.zip
- Structure:

    - Folder ”SQLQueries” - it contains all the queries used throughout the project
        - STA_Queries.sql
        - ODS_Quewries.sql
        - DWH_Queries.sql
        - DimDate_Query.sql
        - Business_Queries.sql
    - Folder “source_data” - it contains the 8 CSV files containing all the RAW data from Kaggle
    - Visual Studio data:
        - OLIST.sln
        - OLIST folder

Dataset

- Source: Kaggle
- Link to dataset: https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce
- Dataset description: The dataset is from the e-commerce business called OLIST from Brazil and it was chosen because it is very comprehensive as it includes the following tables:
    - Customers
    - Orders
    - Payments
    - Sellers
    - OrderItems
    - OrderPayments
    - OrderReviews
    - ProductCategoryNameTransation (from Portuguese to English)
- Dataset format: CSV
- Data size: ~100,000 rows

All queries are stored in the folder: “SQLQueries”

ETL Process and key transformations

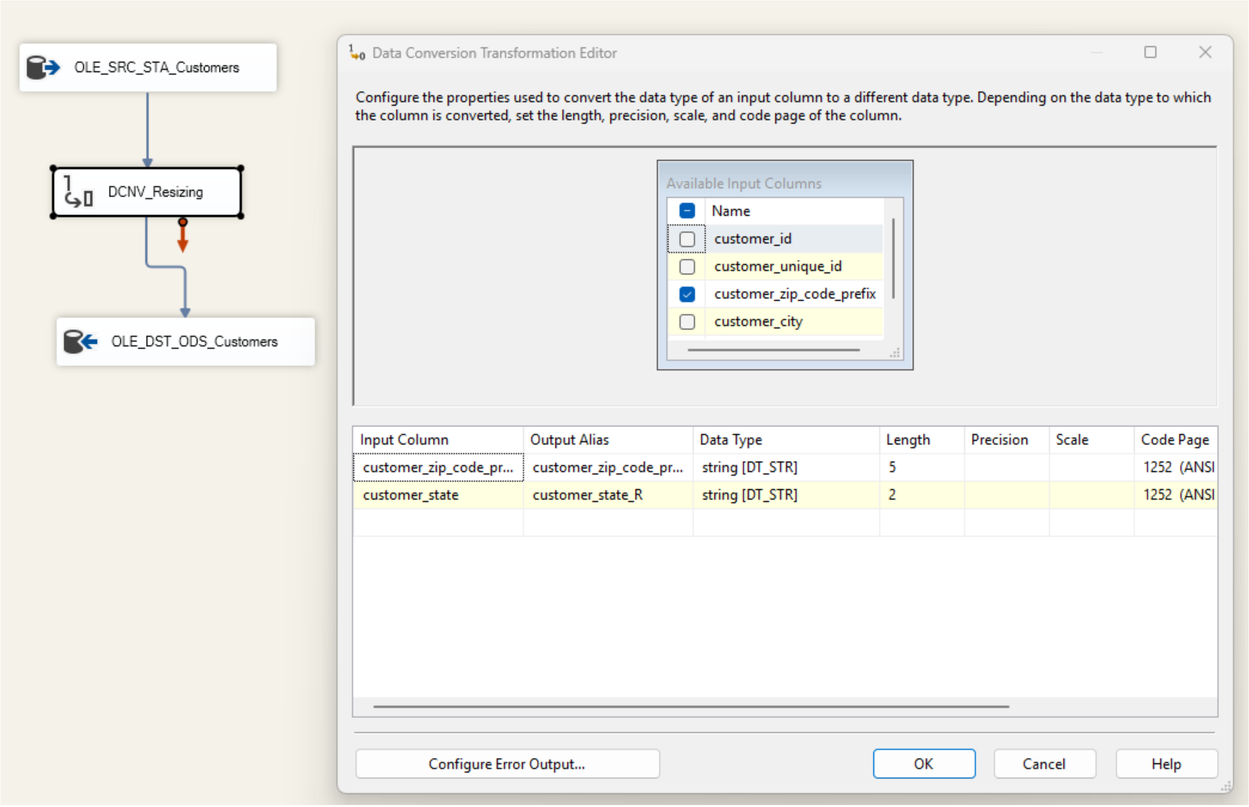
STA

All the source files were uploaded to the OLIST_STA database “as is”. Here is the mapping between source CSV files and DB tables:

Source file	OLIST_STA table
olist_customers_dataset.csv	STA_Customers
olist_order_items_dataset.csv	STA_OrderItems
olist_order_payments_dataset.csv	STA_OrderPayments
olist_order_reviews_dataset.csv	STA_OrderReviews
olist_orders_dataset.csv	STA_Orders
olist_products_dataset.csv	STA_Products
olist_sellers_dataset.csv	STA_Sellers
product_category_name_transition.csv	STA_ProductTranslations

ODS

1. STA_Customers – ODS_Customers
    - Resizing customer_zip_code and customer_state



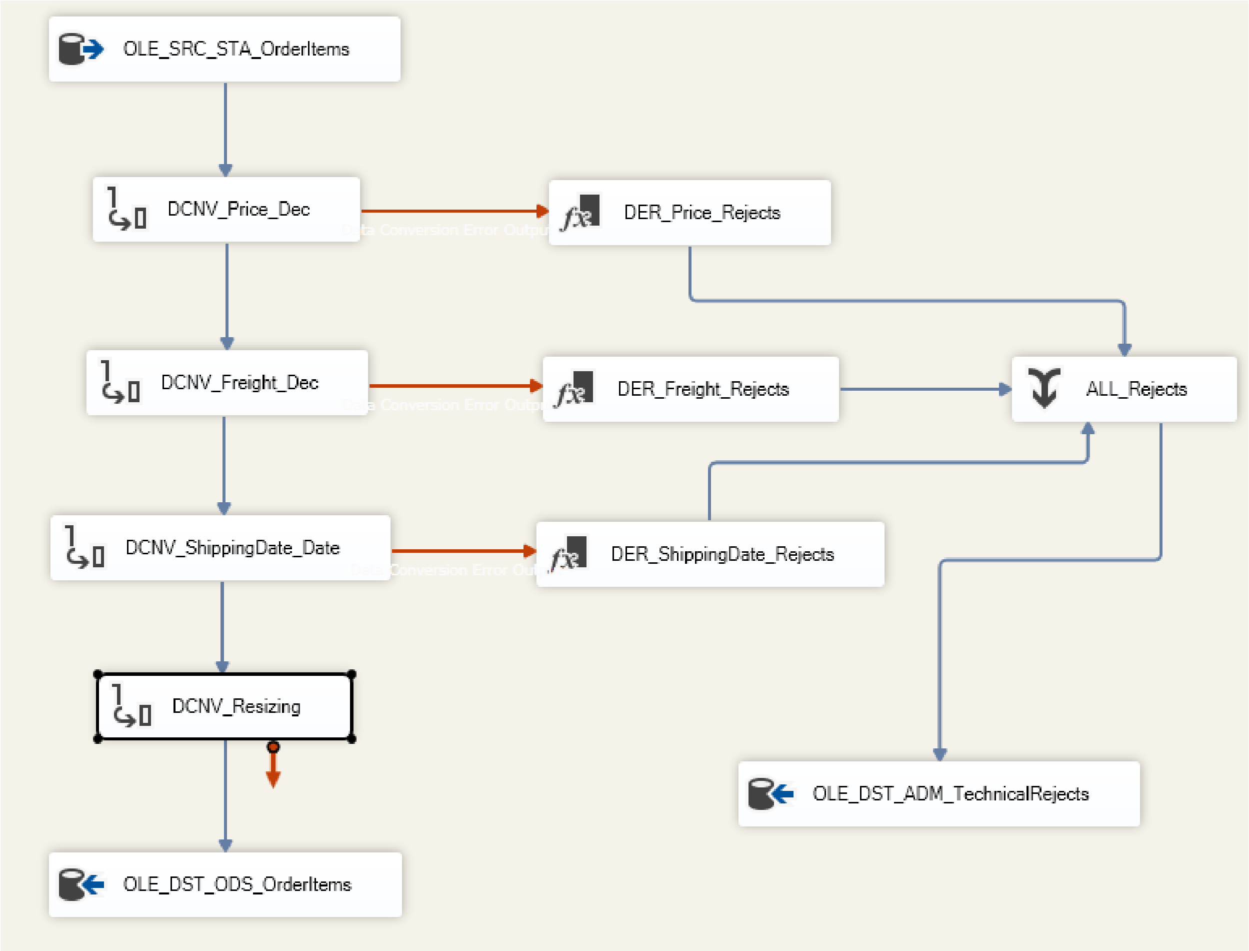
Configure the properties used to convert the data type of an input column to a different data type. Depending on the data type to which the column is converted, set the length, precision, scale, and code page of the column.

Available Input Columns

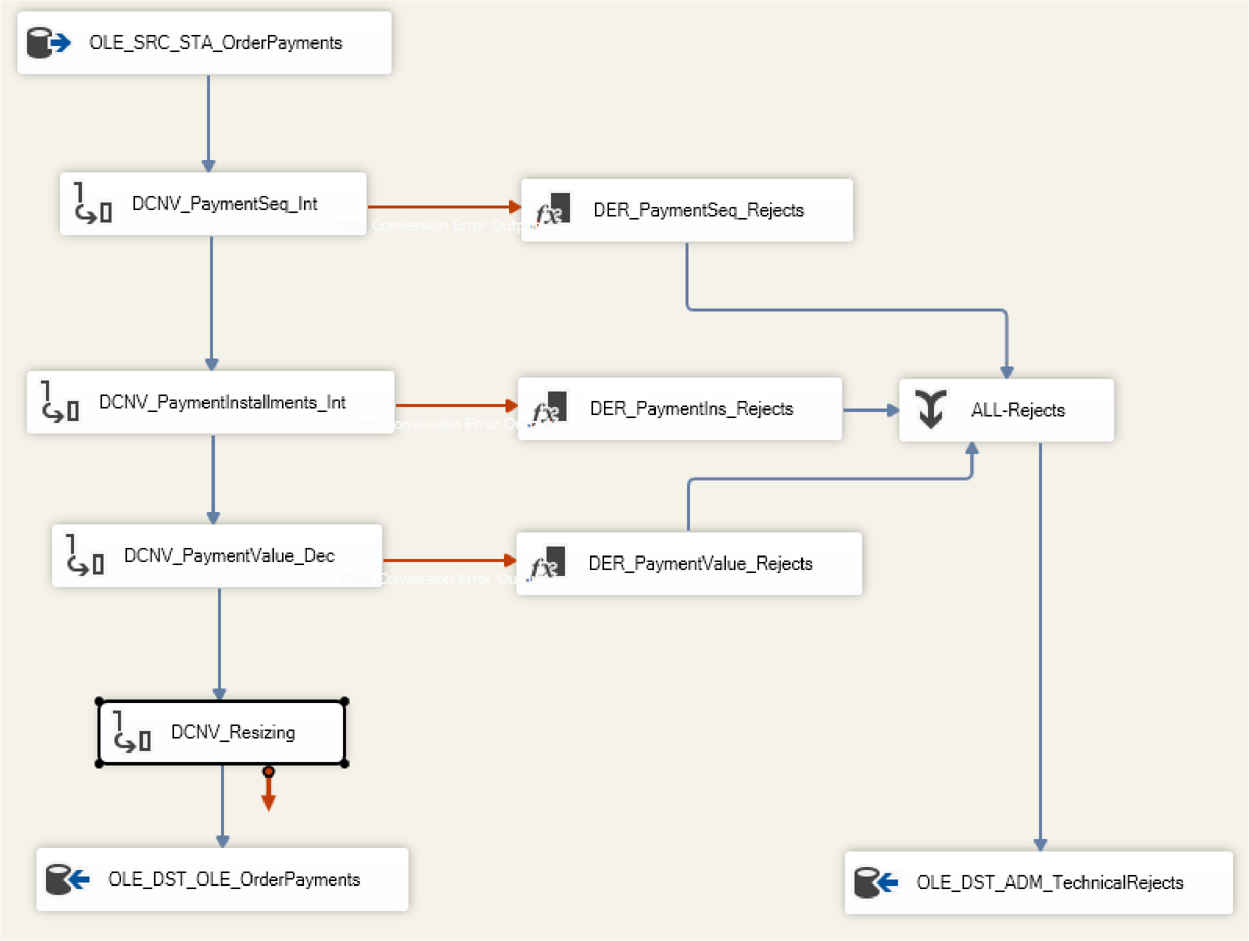
	Name
☐	customer_id
☐	customer_unique_id
☑	customer_zip_code_prefix
☐	customer_city

Input Column	Output Alias	Data Type	Length	Precision	Scale	Code Page
customer_zip_code_pr...	customer_zip_code_pr...	string [DT_STR]	5			1252 (ANSI
customer_state	customer_state_R	string [DT_STR]	2			1252 (ANSI

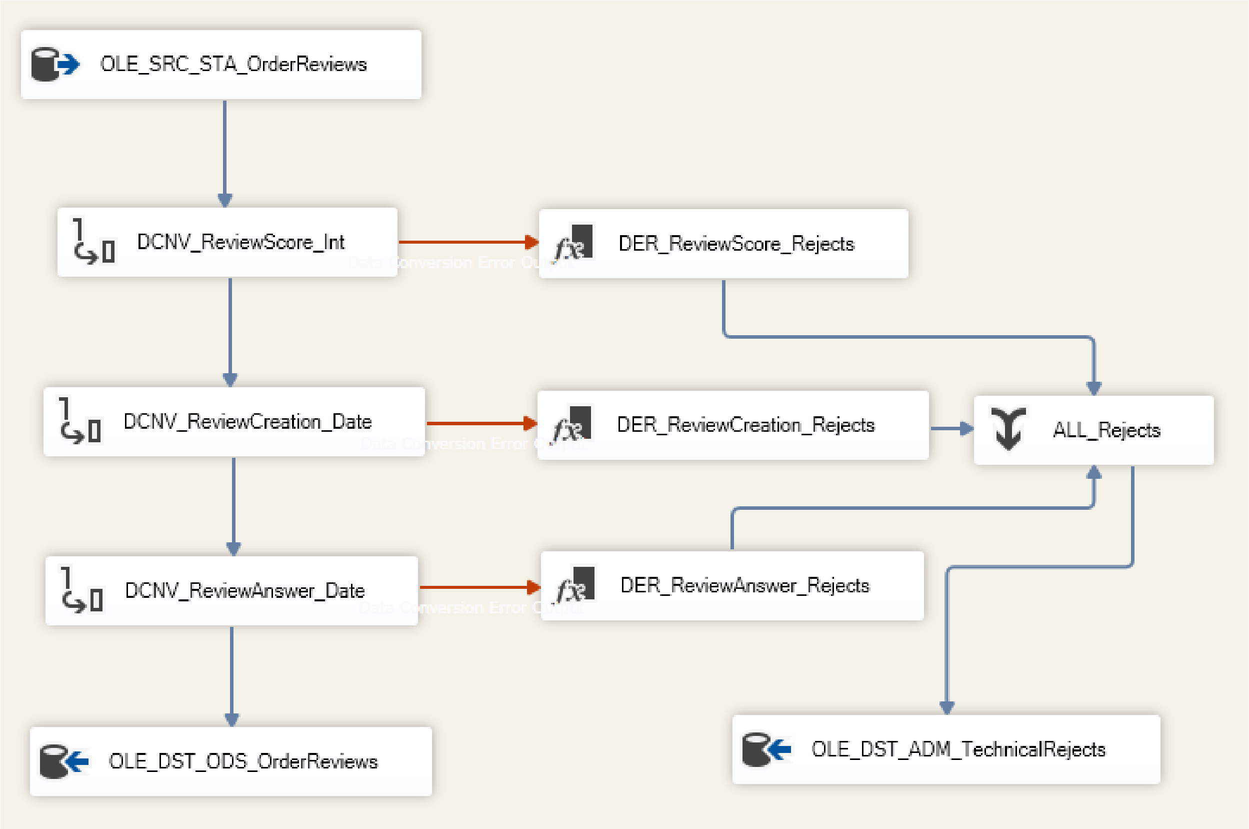
2. STA_OrderItems – ODS_OrderItems
    - Converting Price and Freight to numeric data type
    - Converting ShippingDate to a date data type
    - Resizing order_item



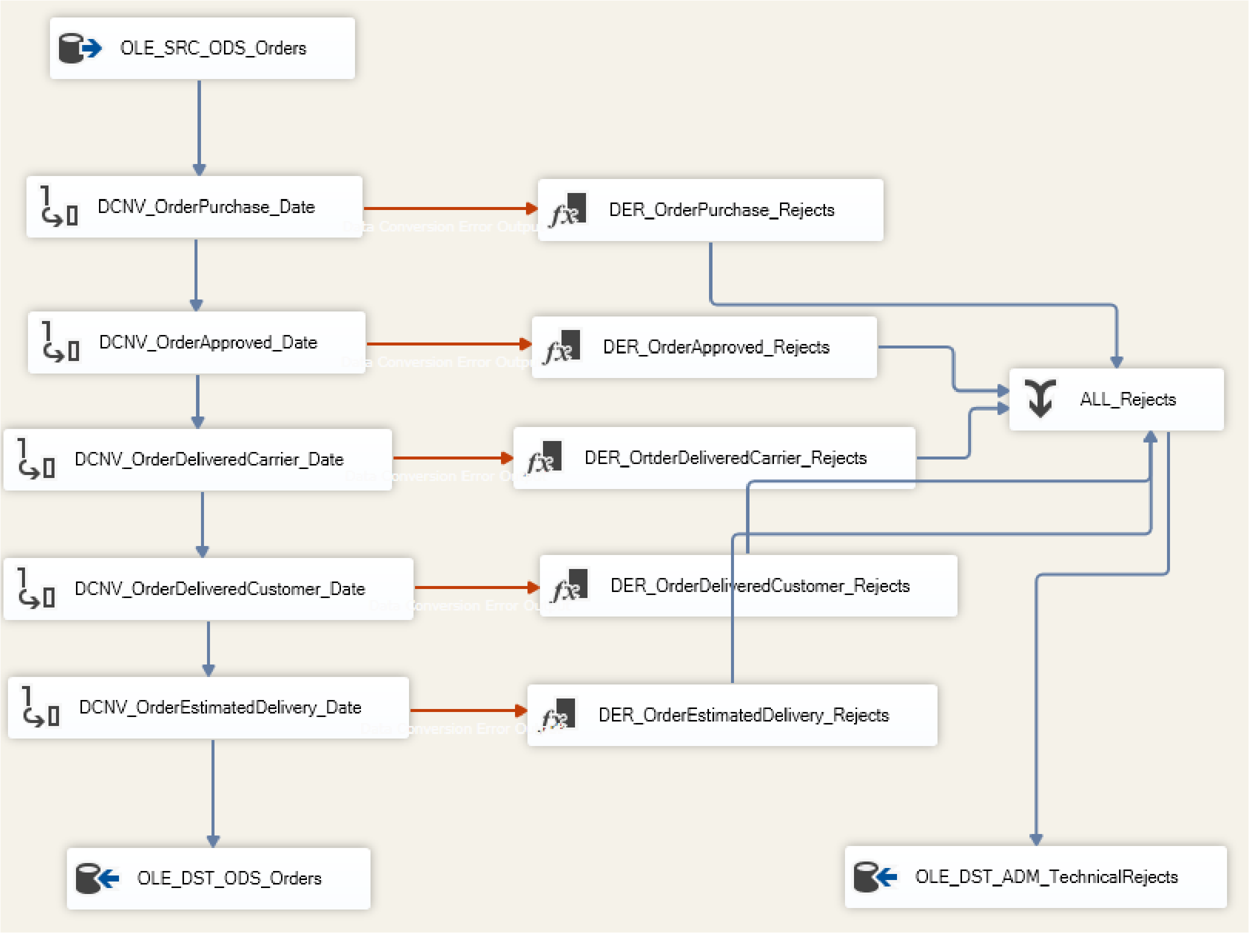
3. STA_OrderPayments – ODS_OrderPayments
    - Converting PaymentSequential, PaymentInstallments and PaymentValue to a numeric data type
    - Resizing PaymentType


OLE_SRC_STA_OrderPayments
DCNV_PaymentSeq_Int
Data Conversion Error Outp
DER_PaymentSeq_Rejects
DCNV_PaymentInstallments_Int
Data Conversion Error Out
DER_PaymentIns_Rejects
ALL-Rejects
DCNV_PaymentValue_Dec
Data Conversion Error Out
DER_PaymentValue_Rejects
DCNV_Resizing
OLE_DST_OLE_OrderPayments
OLE_DST_ADM_TechnicalRejects

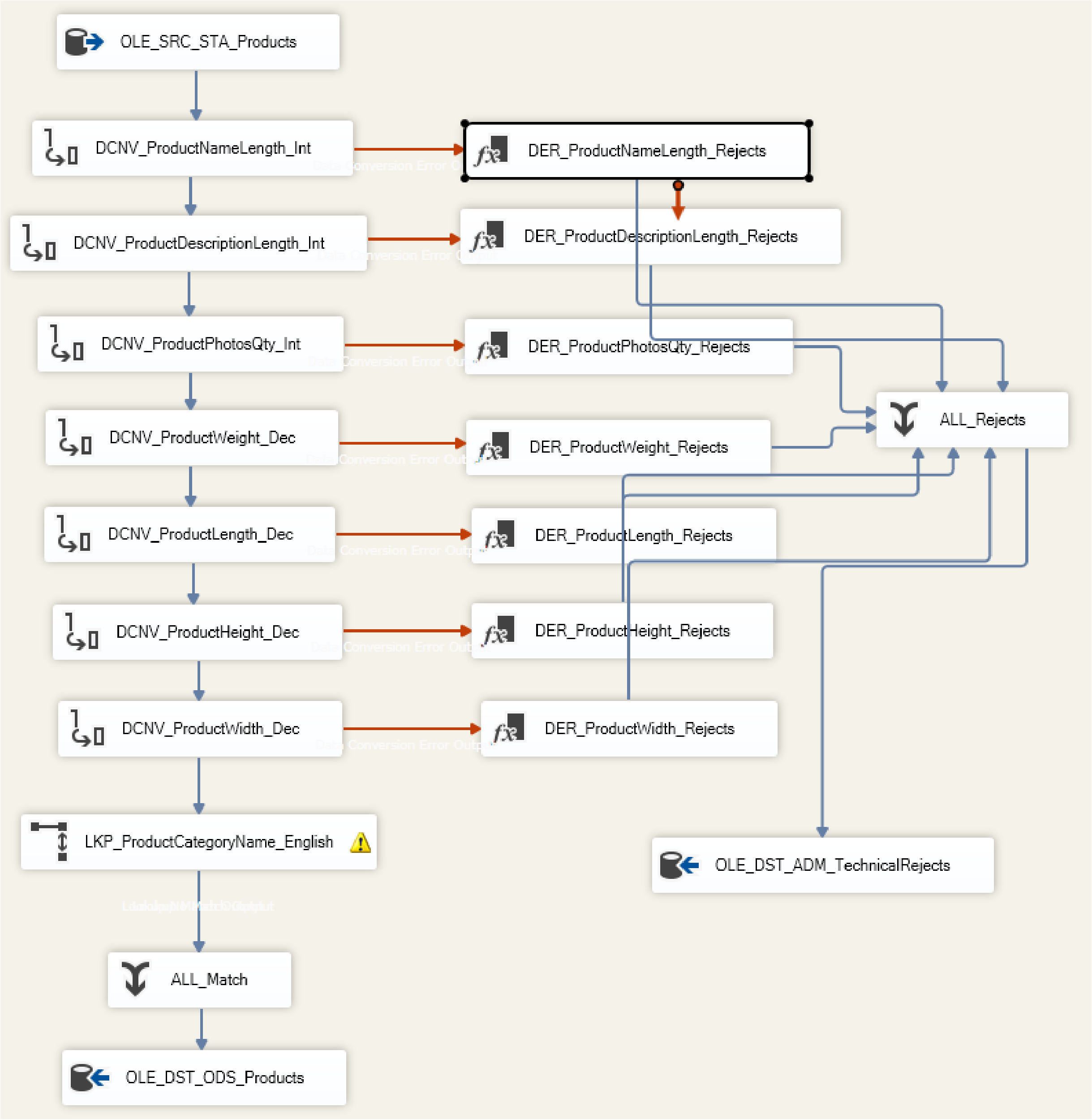
4. STA_OrderReviews – ODS_OrderReviews
    - Converting ReviewScore to a numeric data type
    - Converting ReviewCreation and ReviewAnswer to a date data type



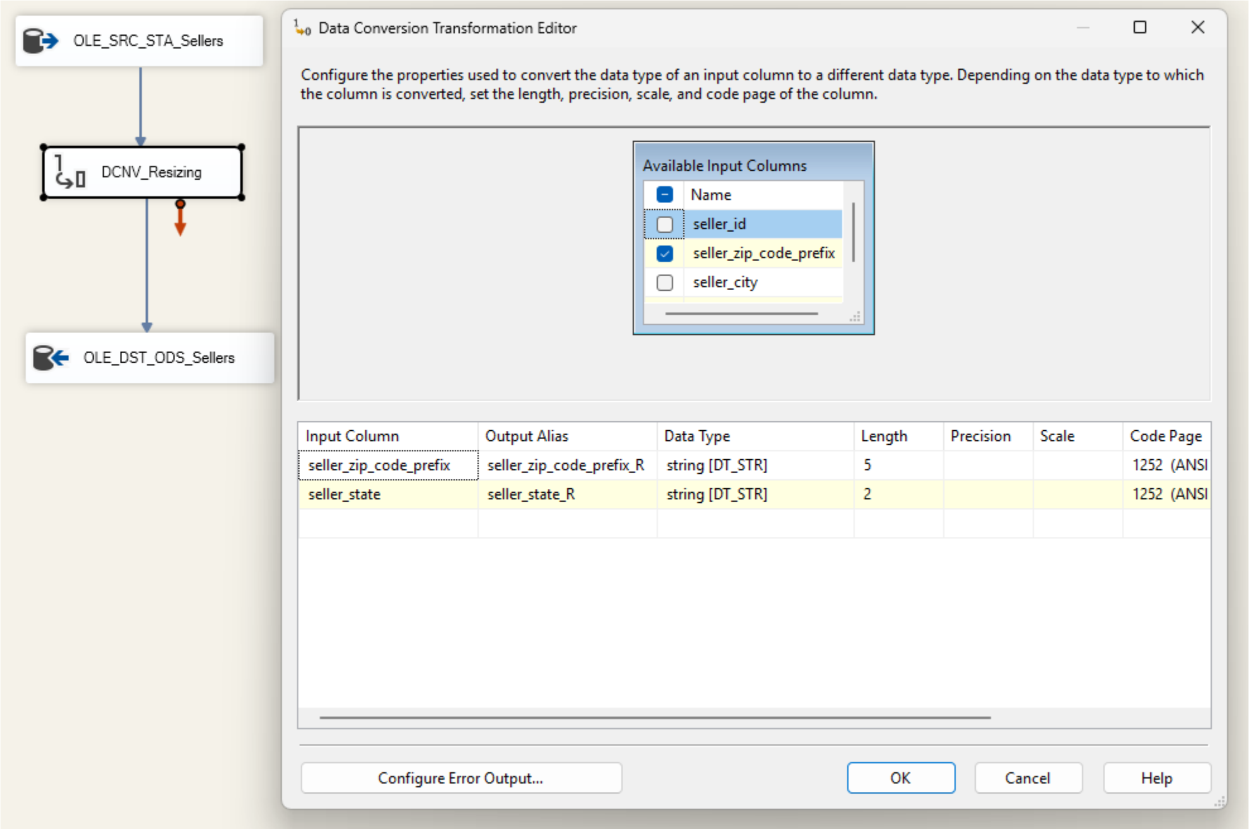
5. STA_Orders – ODS_Orders
    - Converting OrderPurchase, OrderApproved, OrderDeliveredCarrier, OrderDeliveredCustomer and OrderEstimatedDelivery to a date data type


OLE_SRC_ODS_Orders
DCNV_OrderPurchase_Date
DER_OrderPurchase_Rejects
DCNV_OrderApproved_Date
DER_OrderApproved_Rejects
ALL_Rejects
DCNV_OrderDeliveredCarrier_Date
DER_OrtderDeliveredCarrier_Rejects
DCNV_OrderDeliveredCustomer_Date
DER_OrderDeliveredCustomer_Rejects
DCNV_OrderEstimatedDelivery_Date
DER_OrderEstimatedDelivery_Rejects
OLE_DST_ODS_Orders
OLE_DST_ADM_TechnicalRejects

6. STA_Products – ODS_Products
    - Converting ProductNameLength, ProductDescriptionLength, ProductPhotosQty, ProductWeight, ProductHeight and ProductWidth to a numeric data type
    - Lookup between the oprignial product_category_name (in Portuguese) to product_category_name_english. This is why the table STA_ProductTranslations is used on this lookup but not later used anymore


OLE_SRC_STA_Products
DCNV_ProductNameLength_Int
DER_ProductNameLength_Rejects
DCNV_ProductDescriptionLength_Int
DER_ProductDescriptionLength_Rejects
DCNV_ProductPhotosQty_Int
DER_ProductPhotosQty_Rejects
ALL_Rejects
DCNV_ProductWeight_Dec
DER_ProductWeight_Rejects
DCNV_ProductLength_Dec
DER_ProductLength_Rejects
DCNV_ProductHeight_Dec
DER_ProductHeight_Rejects
DCNV_ProductWidth_Dec
DER_ProductWidth_Rejects
LKP_ProductCategoryName_English
OLE_DST_ADM_TechnicalRejects
Lookup No Match Output
ALL_Match
OLE_DST_ODS_Products

7. STA_Sellers – ODS_Sellers
    - Resizing seller_zip_code and seller_state



Configure the properties used to convert the data type of an input column to a different data type. Depending on the data type to which the column is converted, set the length, precision, scale, and code page of the column.

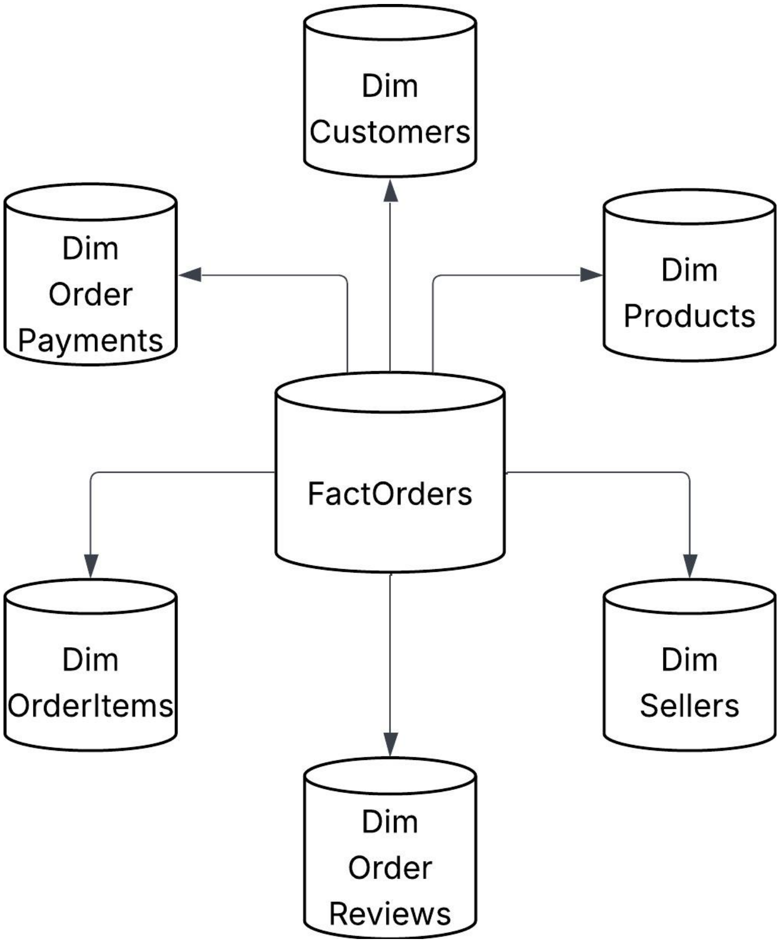
	Name
☐	seller_id
☑	seller_zip_code_prefix
☐	seller_city

Input Column	Output Alias	Data Type	Length	Precision	Scale	Code Page
seller_zip_code_prefix	seller_zip_code_prefix_R	string [DT_STR]	5			1252 (ANSI
seller_state	seller_state_R	string [DT_STR]	2			1252 (ANSI

In all cases of data conversions, the technical rejects have been populated to the TechnicalRejects table in the OLIST_ADM database.

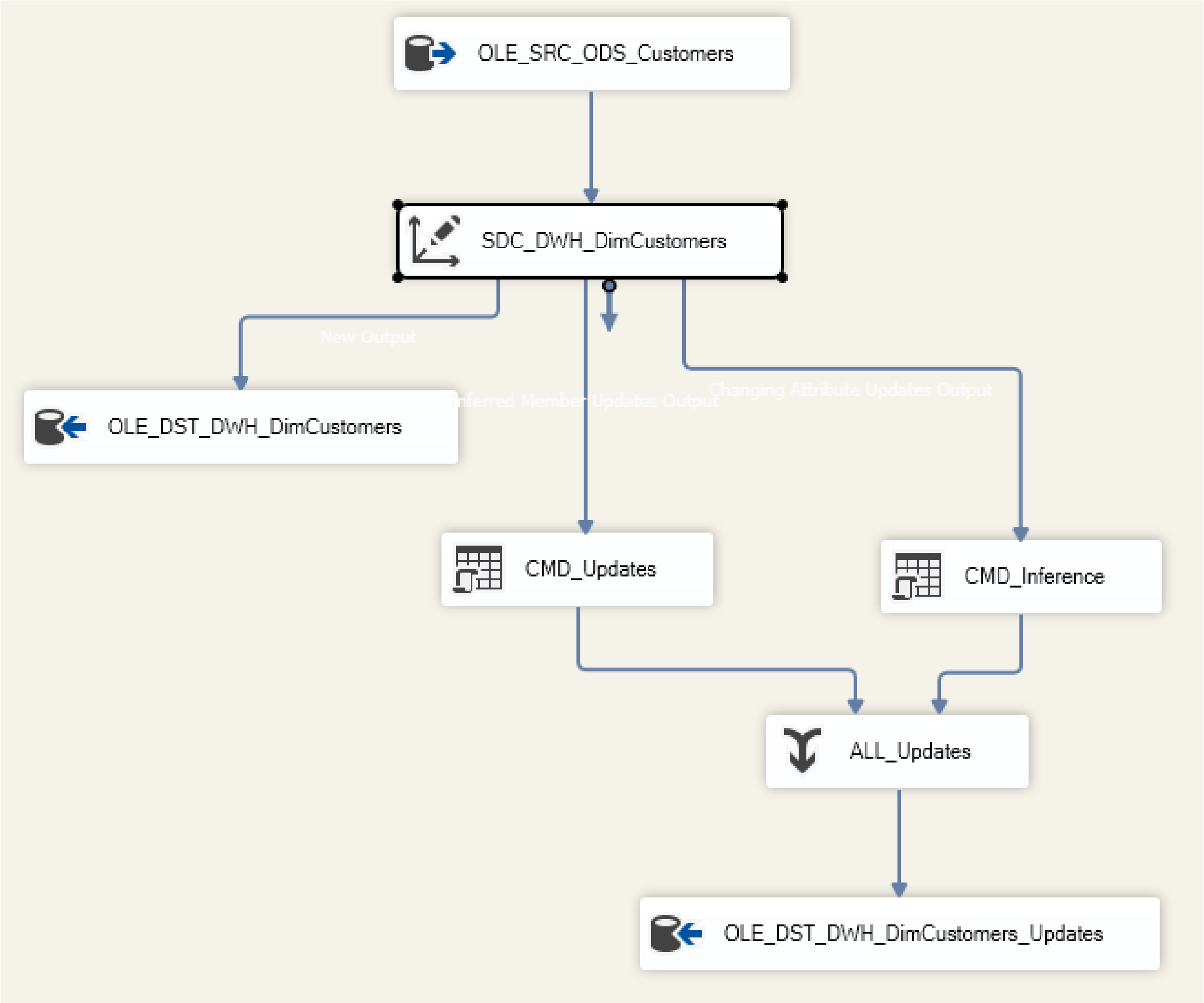
Datawarehouse

STAR Schema

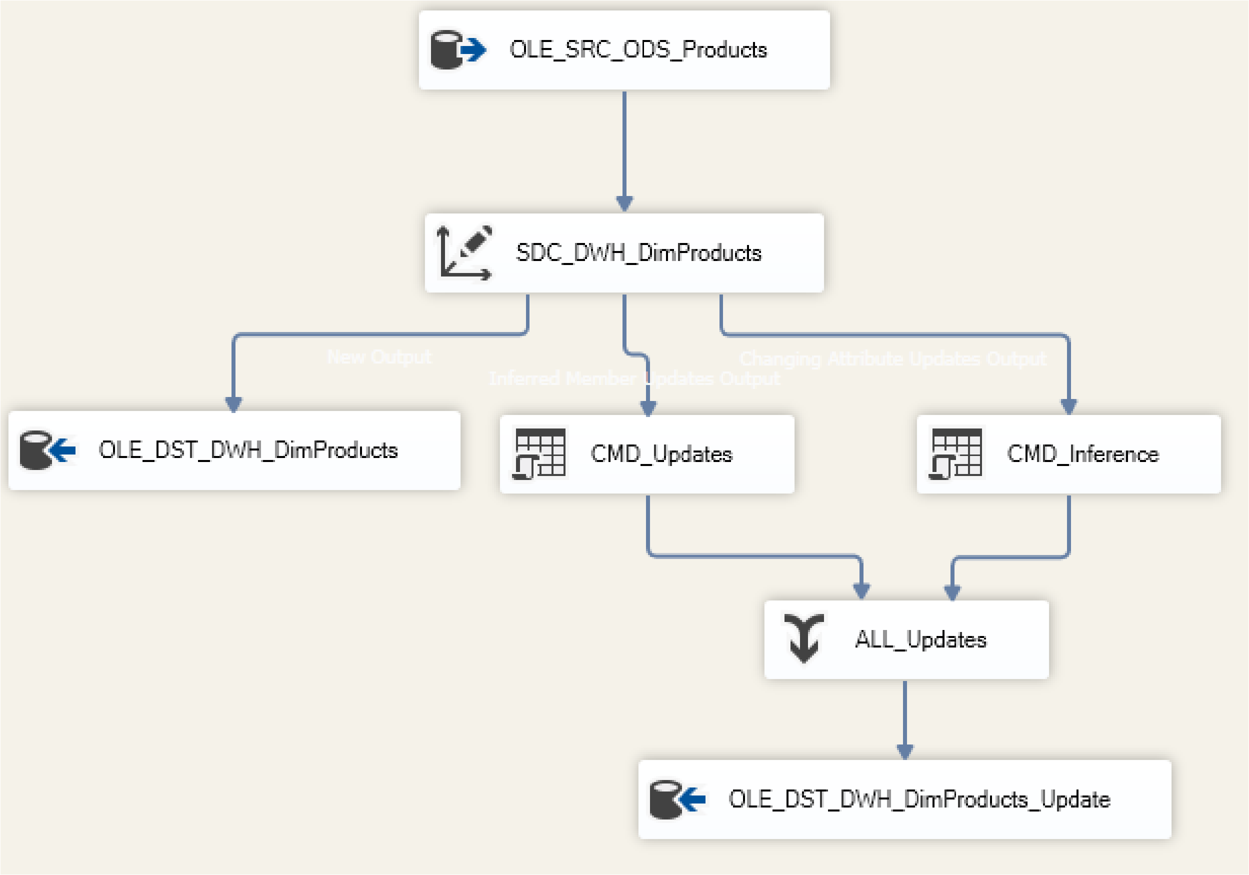


Dimension tables

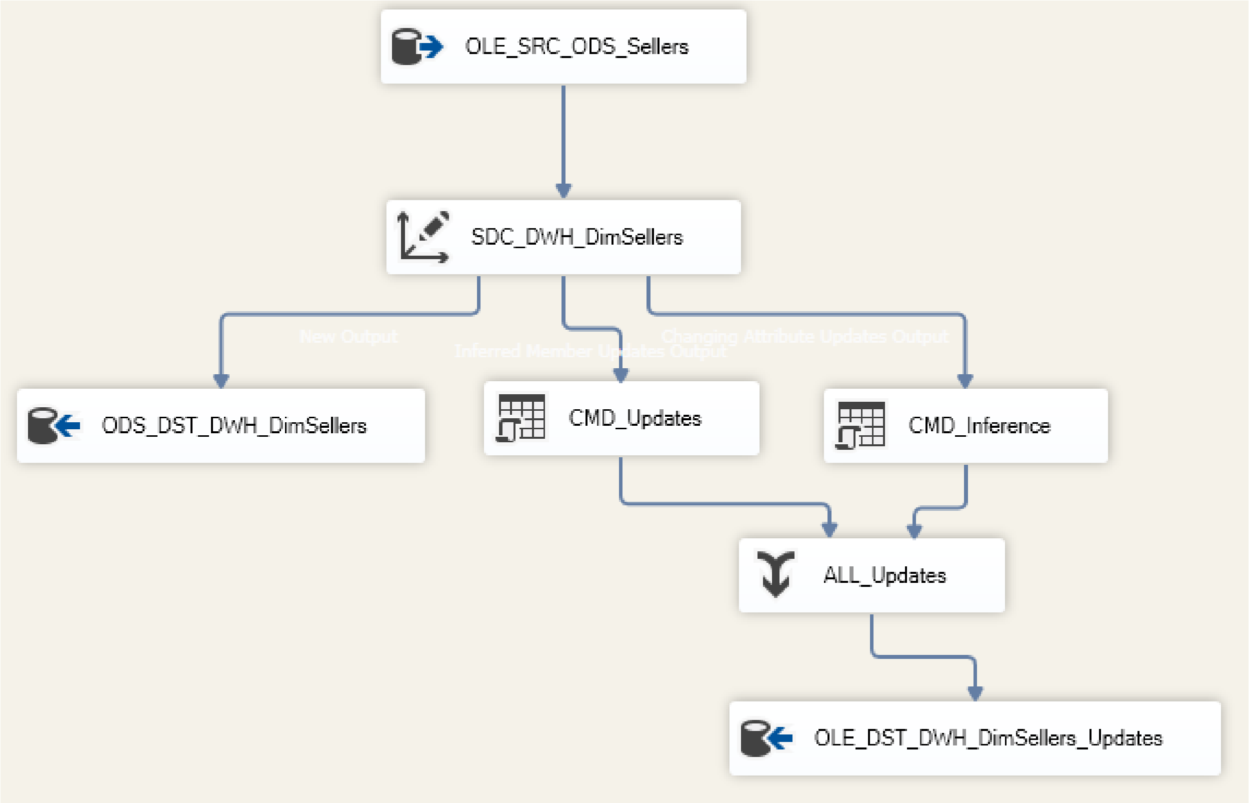
- The chosen option for “Slow Changing Dimension – SDC" was type 1through the use of built-in feature in the SSIS toolbox
- A surrogate key has been added to all the dimension tables
- Tables:
  - DWH_DimDate
  - DWH_DimCustomer



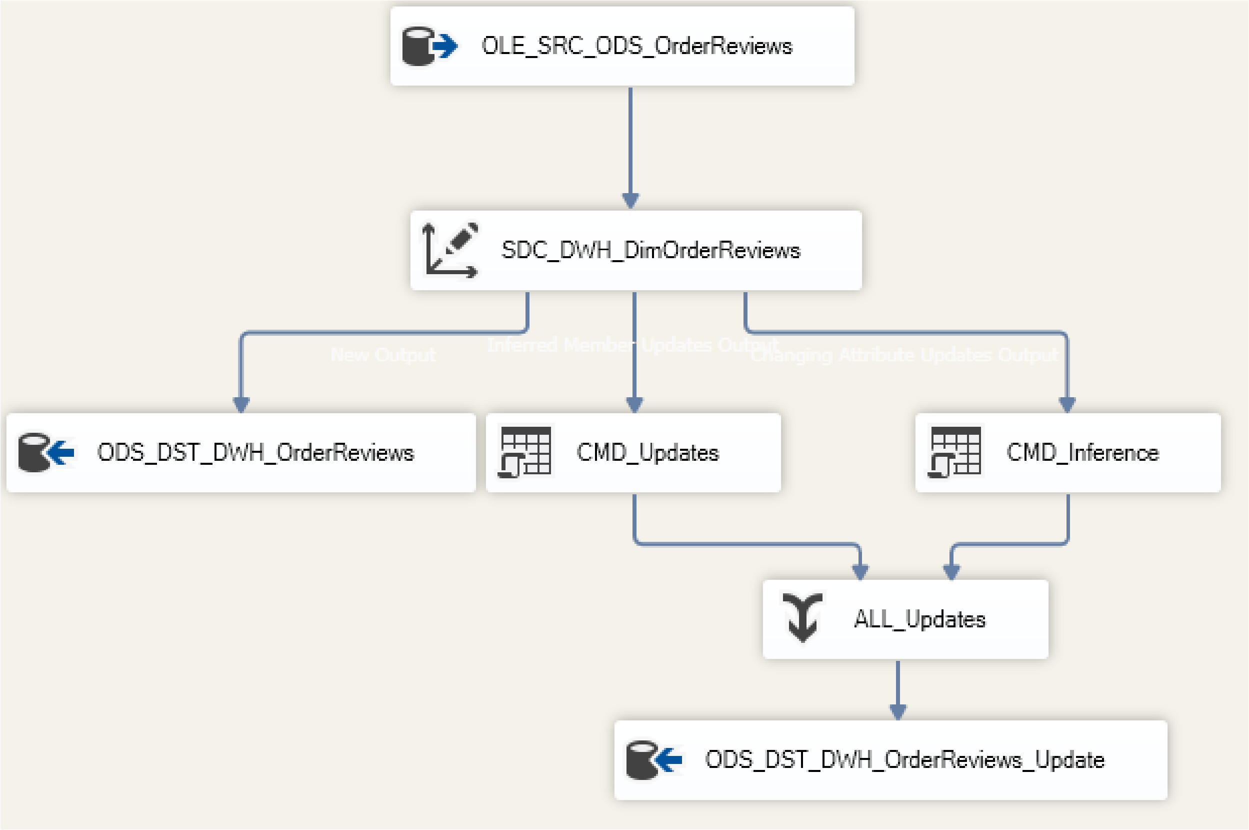
- DWH_DimProducts



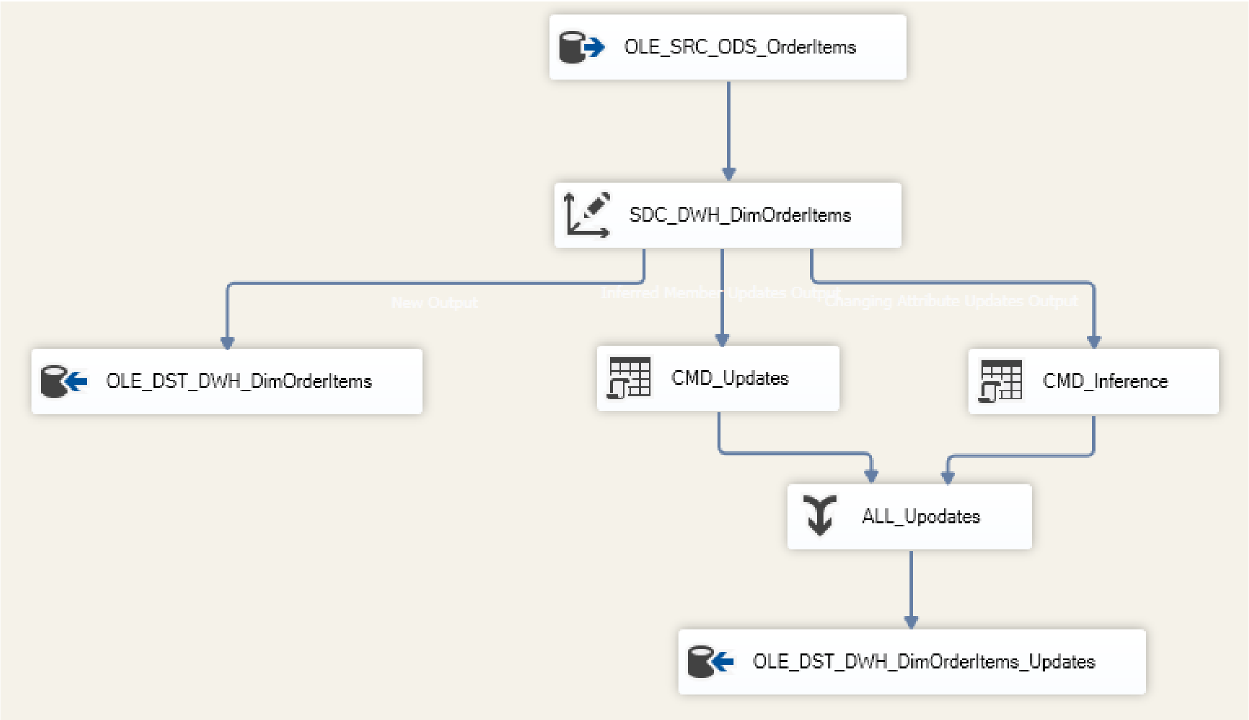
- DWH_DimSellers



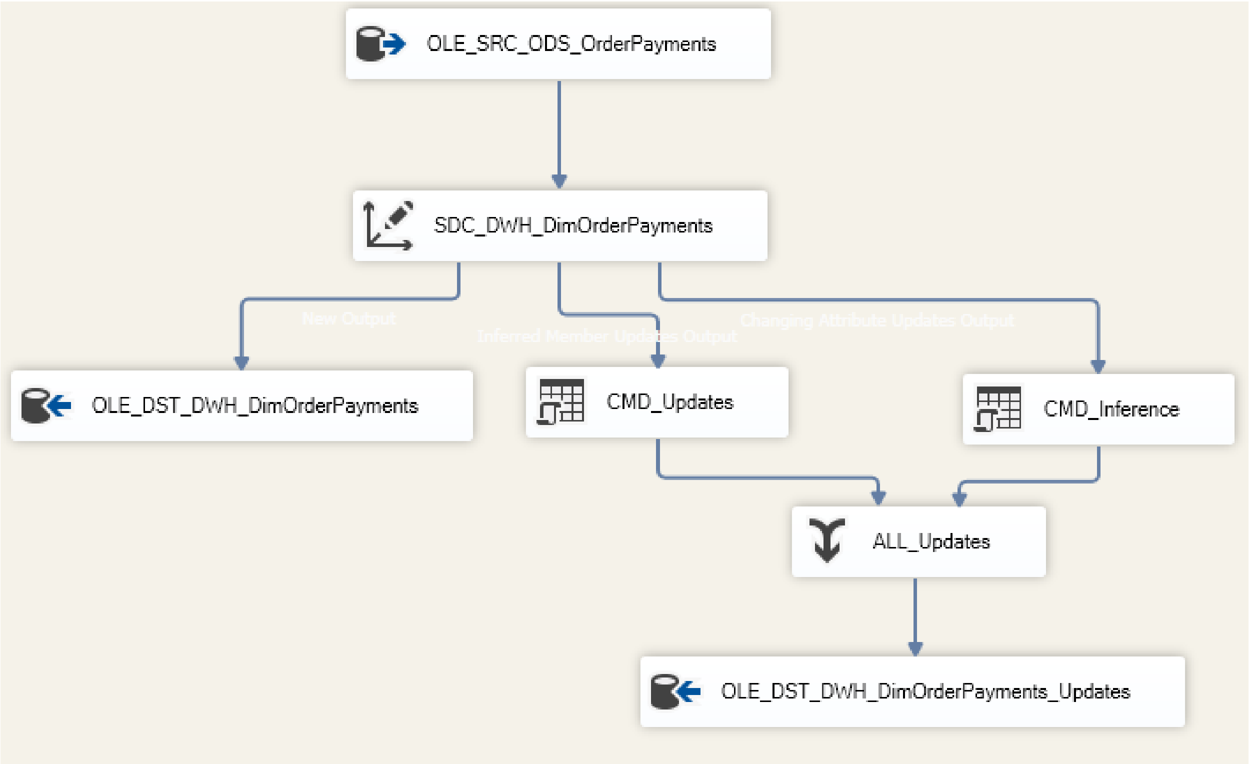
- DWH_DimOrderReviews



- DWH_DimOrderItems


OLE_SRC_ODS_OrderItems
SDC_DWH_DimOrderItems
New Output
Inferred Member Updates Output
Changing Attribute Updates Output
OLE_DST_DWH_DimOrderItems
CMD_Updates
CMD_Inference
ALL_Updates
OLE_DST_DWH_DimOrderItems_Updates

- DWH_DimOrderPayments


OLE_SRC_ODS_OrderPayments
SDC_DWH_DimOrderPayments
New Output
Inferred Member Updates Output
Changing Attribute Updates Output
OLE_DST_DWH_DimOrderPayments
CMD_Updates
CMD_Inference
ALL_Updates
OLE_DST_DWH_DimOrderPayments_Updates

Fact table

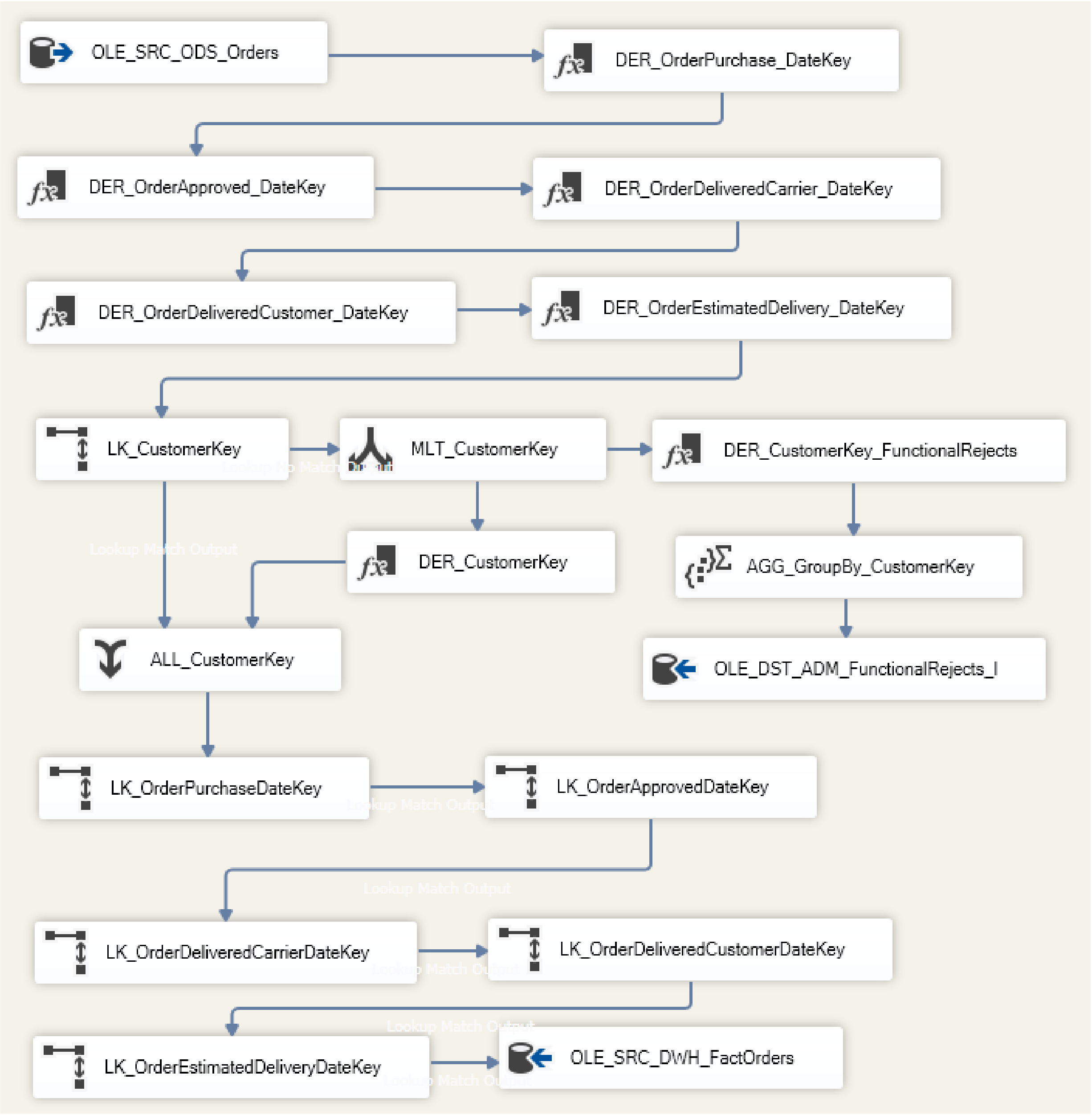
The source table to the FACT table is ODS_Orders which contains the following fields:

- Order_id
- Customer_id
- Order_status
- Order_purchase_timestamp (timestamp)
- Order_approved_at (timestamp)
- Order_delivered_carrier_date (timestamp)
- Order_delivered_customer_date (timestamp)
- Order_estimated_delivery_date (timestamp)

The FACT table is DWH_FactOrders contains the following fields:

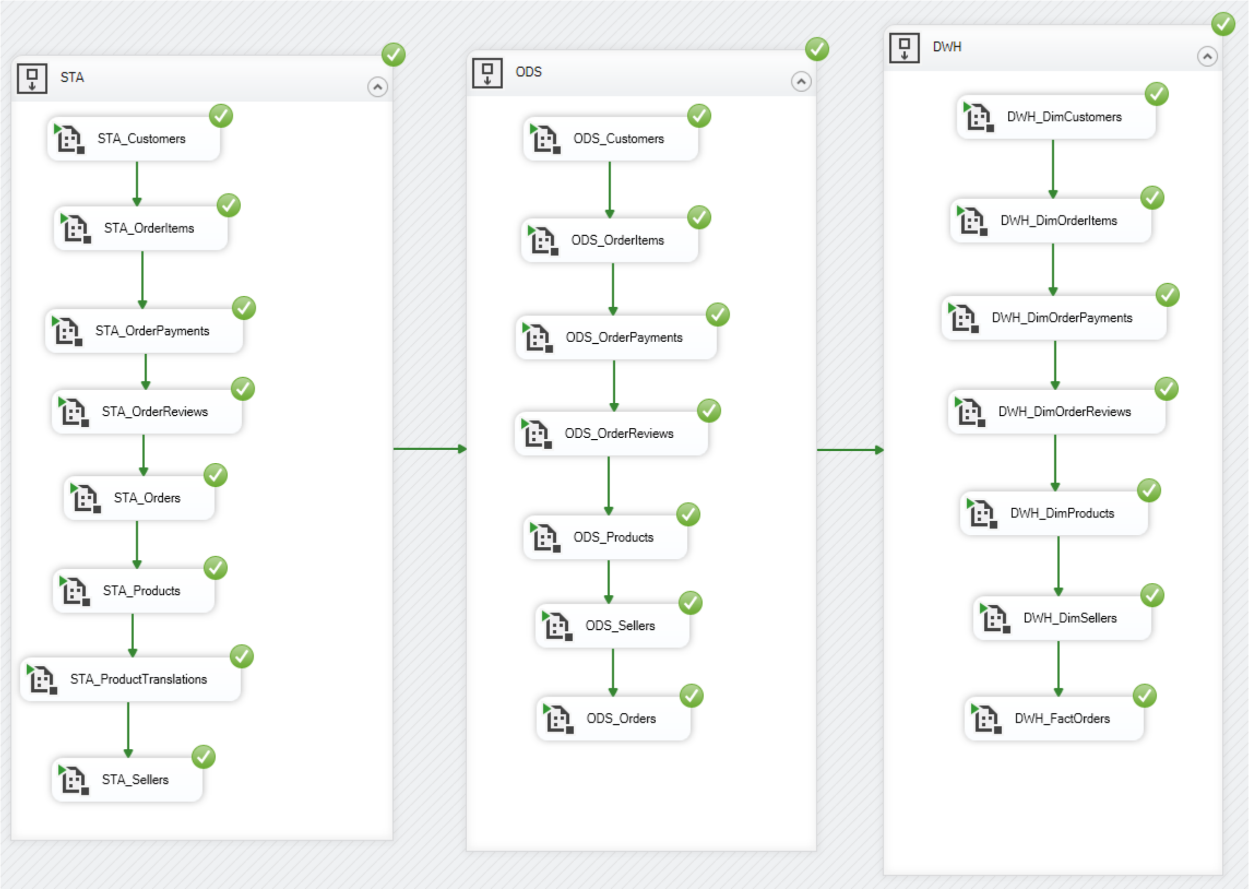
- Order_id
- Customer_key (surrogate key from the DWH_DimCustomers)
- Order_status
- Order_purchase_datekey (date) - removing the timestamp to match the date_key INT from the DimDate table
- Order_approved_datekey (date) - removing the timestamp to match the date_key INT from the DimDate table
- Order_delivered_carrier_datekey (date) - removing the timestamp to match the date_key INT from the DimDate table
- Order_delivered_customer_datekey (date) - removing the timestamp to match the date_key INT from the DimDate table
- Order_estimated_delivery_datekey (date) - removing the timestamp to match the date_key INT from the DimDate table
- Order_purchase_timestamp (timestamp)
- Order_approved_at (timestamp)
- Order_delivered_carrier_date (timestamp)
- Order_delivered_customer_date (timestamp)
- Order_estimated_delivery_date (timestamp)

In the cases where the surrogate key does not exist because the record does not exist, the surrogate key is created as a –1 and added to the table FunctionalRejects in the OLIST_ADM database


OLE_SRC_ODS_Orders
DER_OrderPurchase_DateKey
DER_OrderApproved_DateKey
DER_OrderDeliveredCarrier_DateKey
DER_OrderDeliveredCustomer_DateKey
DER_OrderEstimatedDelivery_DateKey
LK_CustomerKey
MLT_CustomerKey
DER_CustomerKey_FunctionalRejects
Lookup Match Output
DER_CustomerKey
AGG_GroupBy_CustomerKey
ALL_CustomerKey
OLE_DST_ADM_FunctionalRejects_I
LK_OrderPurchaseDateKey
Lookup Match Output
LK_OrderApprovedDateKey
Lookup Match Output
LK_OrderDeliveredCarrierDateKey
LK_OrderDeliveredCustomerDateKey
Lookup Match Output
LK_OrderEstimatedDeliveryDateKey
OLE_SRC_DWH_FactOrders

Execution

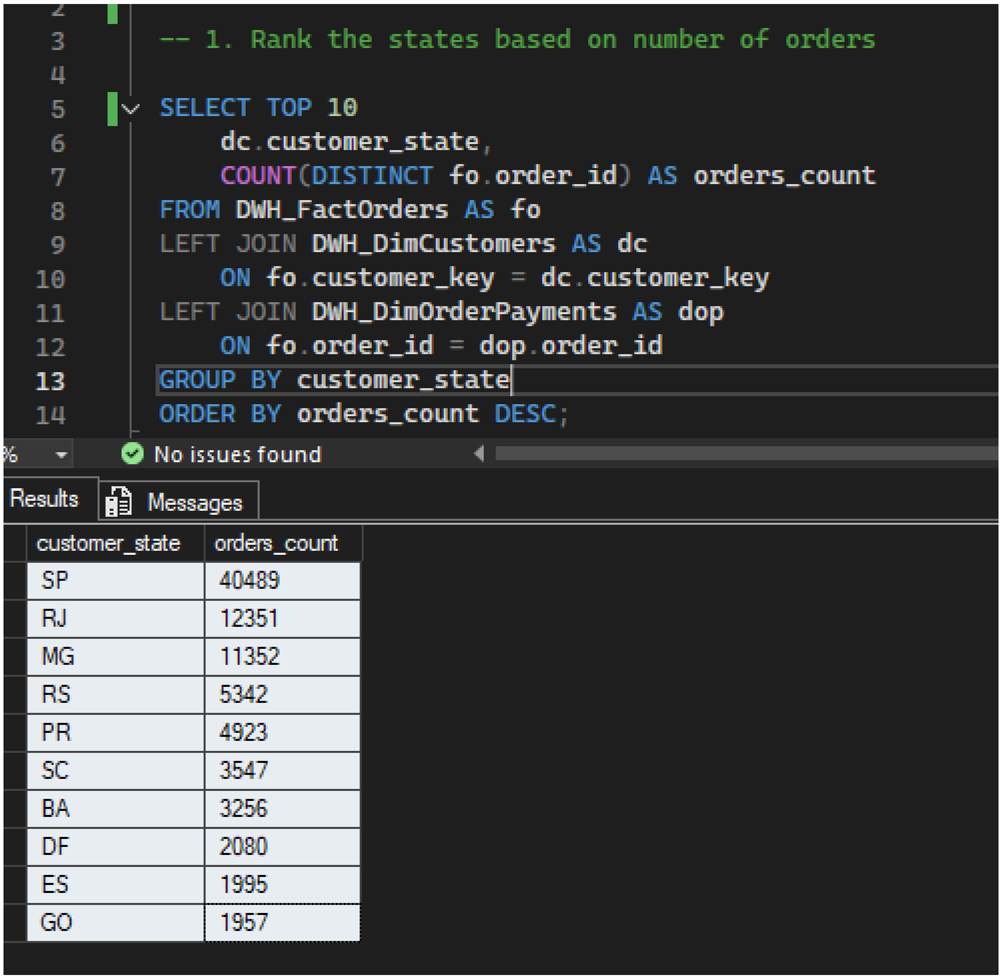
Execute the SSIS package “OLIST_Exec.dtsx” to execute the entire pipeline in the required order



Data insights

Note: all the queries are in the file Business_Queries.sql

1. Top 10 Brazilian states with the highest number of orders

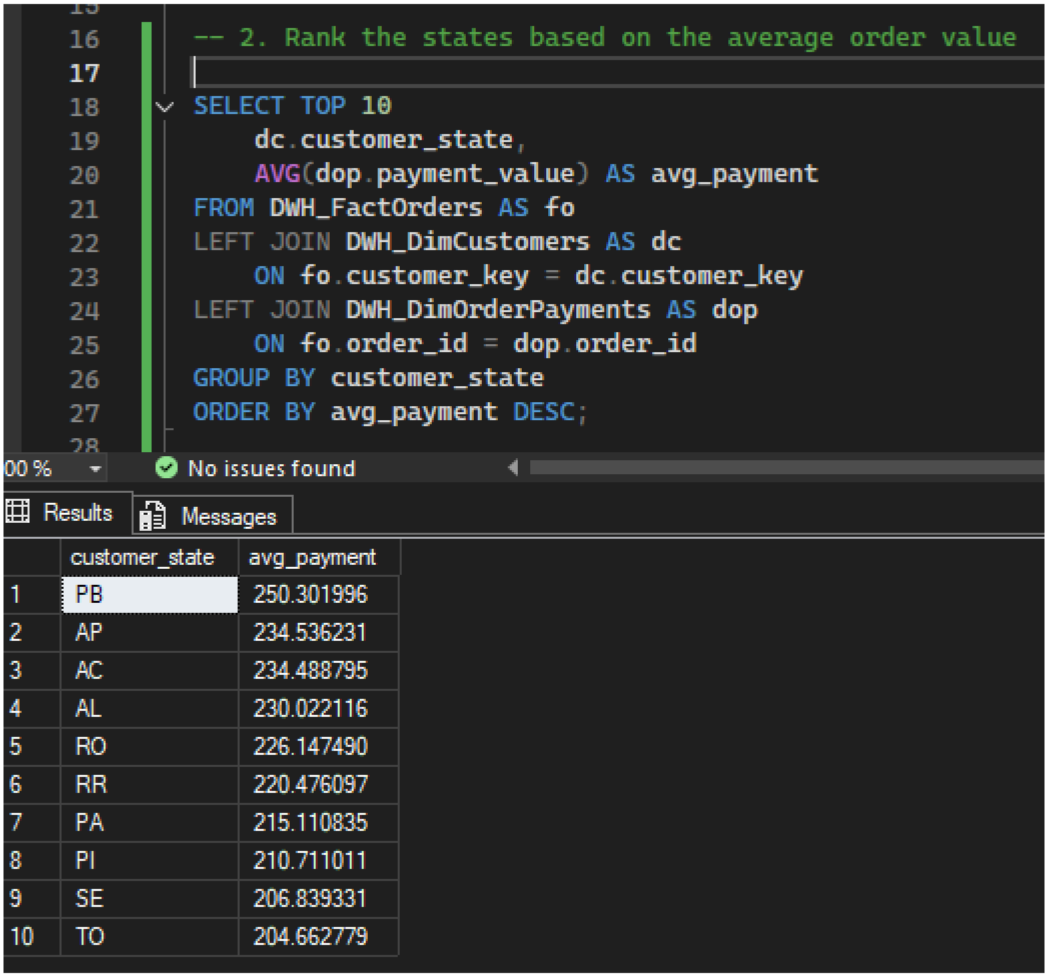


```sql
-- 1. Rank the states based on number of orders

SELECT TOP 10
    dc.customer_state,
    COUNT(DISTINCT fo.order_id) AS orders_count
FROM DWH_FactOrders AS fo
LEFT JOIN DWH_DimCustomers AS dc
    ON fo.customer_key = dc.customer_key
LEFT JOIN DWH_DimOrderPayments AS dop
    ON fo.order_id = dop.order_id
GROUP BY customer_state
ORDER BY orders_count DESC;
```

customer_state	orders_count
SP	40489
RJ	12351
MG	11352
RS	5342
PR	4923
SC	3547
BA	3256
DF	2080
ES	1995
GO	1957

2. Top 10 Brazilian states with the highest average payment value per order

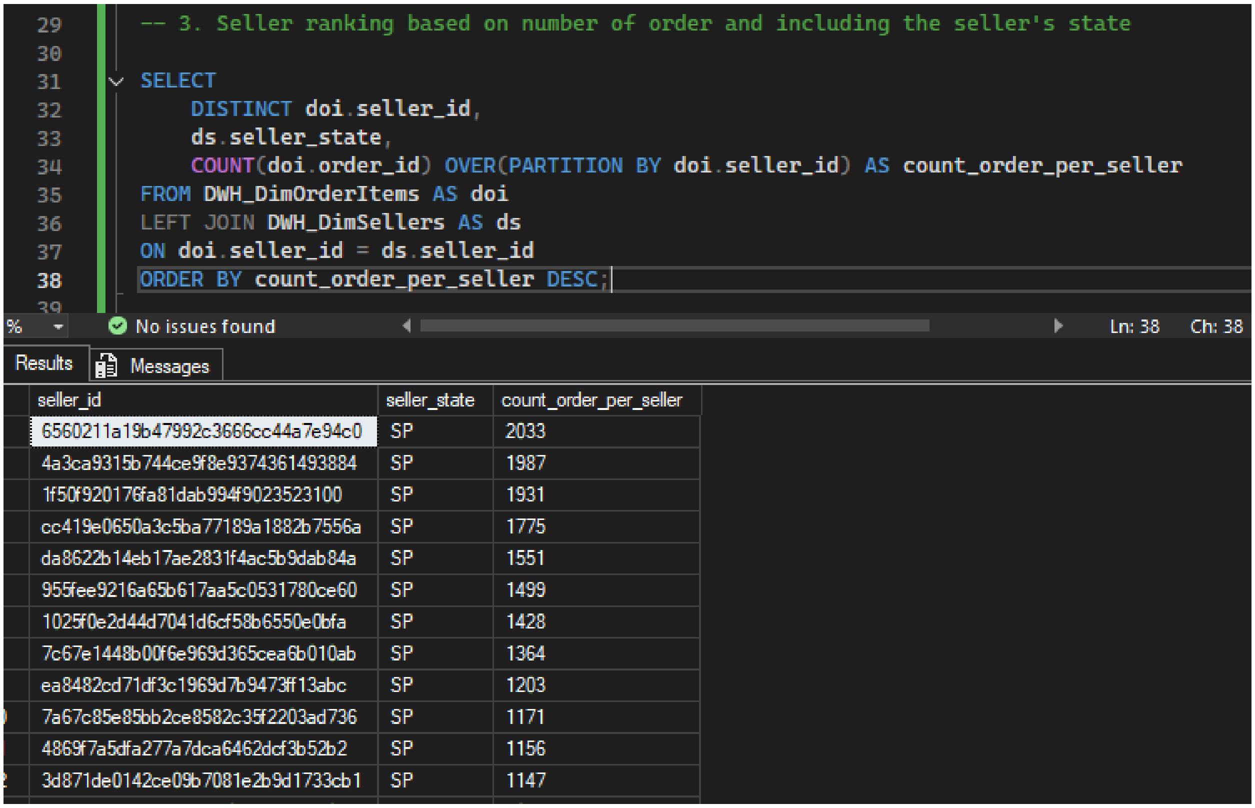


```sql
-- 2. Rank the states based on the average order value

SELECT TOP 10
    dc.customer_state,
    AVG(dop.payment_value) AS avg_payment
FROM DWH_FactOrders AS fo
LEFT JOIN DWH_DimCustomers AS dc
    ON fo.customer_key = dc.customer_key
LEFT JOIN DWH_DimOrderPayments AS dop
    ON fo.order_id = dop.order_id
GROUP BY customer_state
ORDER BY avg_payment DESC;
```

	customer_state	avg_payment
1	PB	250.301996
2	AP	234.536231
3	AC	234.488795
4	AL	230.022116
5	RO	226.147490
6	RR	220.476097
7	PA	215.110835
8	PI	210.711011
9	SE	206.839331
10	TO	204.662779

3. Listing the sellers based on number of order and the stated where they belong to

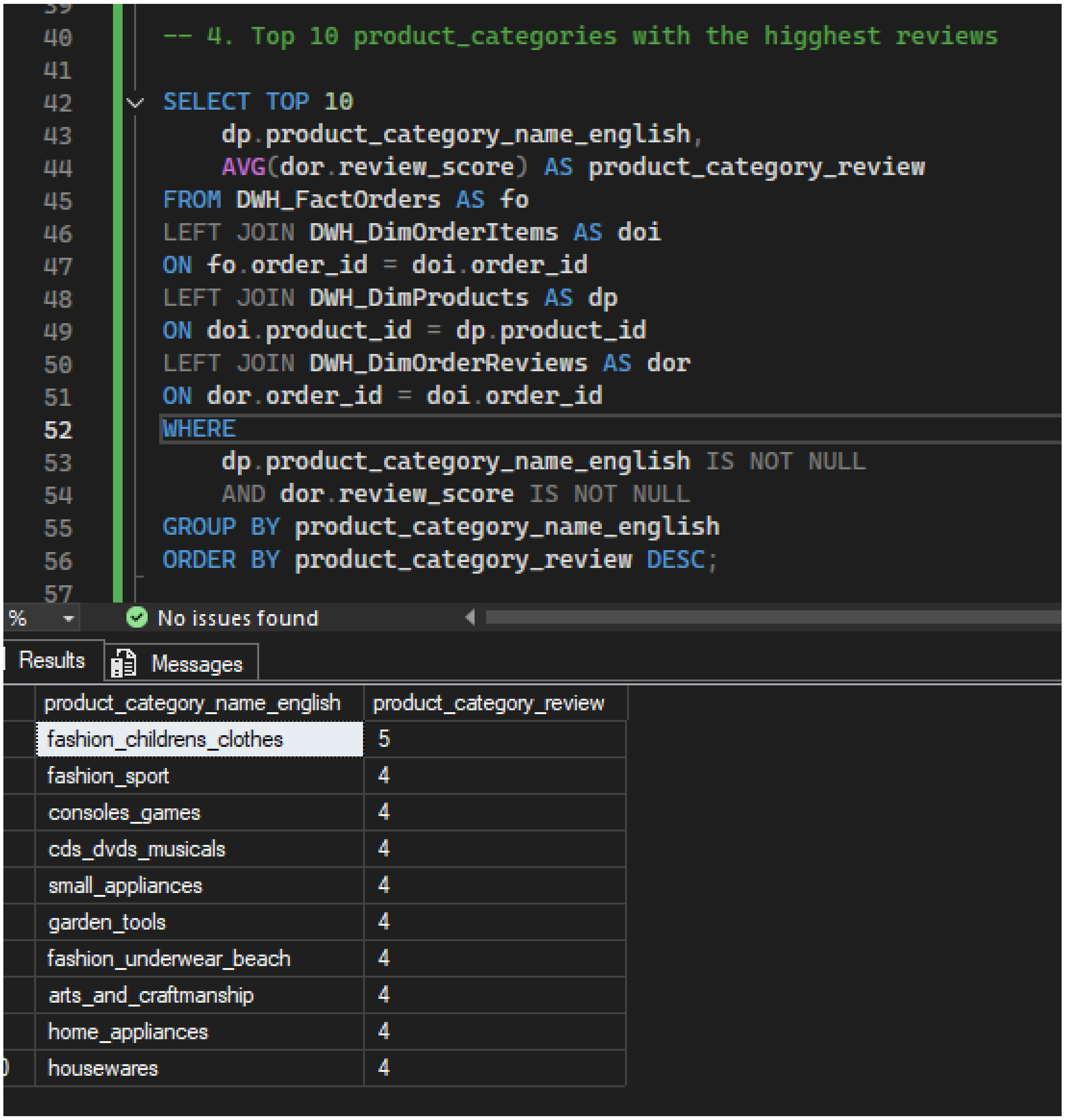


```sql
-- 3. Seller ranking based on number of order and including the seller's state

SELECT
    DISTINCT doi.seller_id,
    ds.seller_state,
    COUNT(doi.order_id) OVER(PARTITION BY doi.seller_id) AS count_order_per_seller
FROM DWH_DimOrderItems AS doi
LEFT JOIN DWH_DimSellers AS ds
ON doi.seller_id = ds.seller_id
ORDER BY count_order_per_seller DESC;
```

seller_id	seller_state	count_order_per_seller
6560211a19b47992c3666cc44a7e94c0	SP	2033
4a3ca9315b744ce9f8e9374361493884	SP	1987
1f50f920176fa81dab994f9023523100	SP	1931
cc419e0650a3c5ba77189a1882b7556a	SP	1775
da8622b14eb17ae2831f4ac5b9dab84a	SP	1551
955fee9216a65b617aa5c0531780ce60	SP	1499
1025f0e2d44d7041d6cf58b6550e0bfa	SP	1428
7c67e1448b00f6e969d365cea6b010ab	SP	1364
ea8482cd71df3c1969d7b9473ff13abc	SP	1203
7a67c85e85bb2ce8582c35f2203ad736	SP	1171
4869f7a5dfa277a7dca6462dcf3b52b2	SP	1156
3d871de0142ce09b7081e2b9d1733cb1	SP	1147

4. Top 10 product categories with the highest review scores

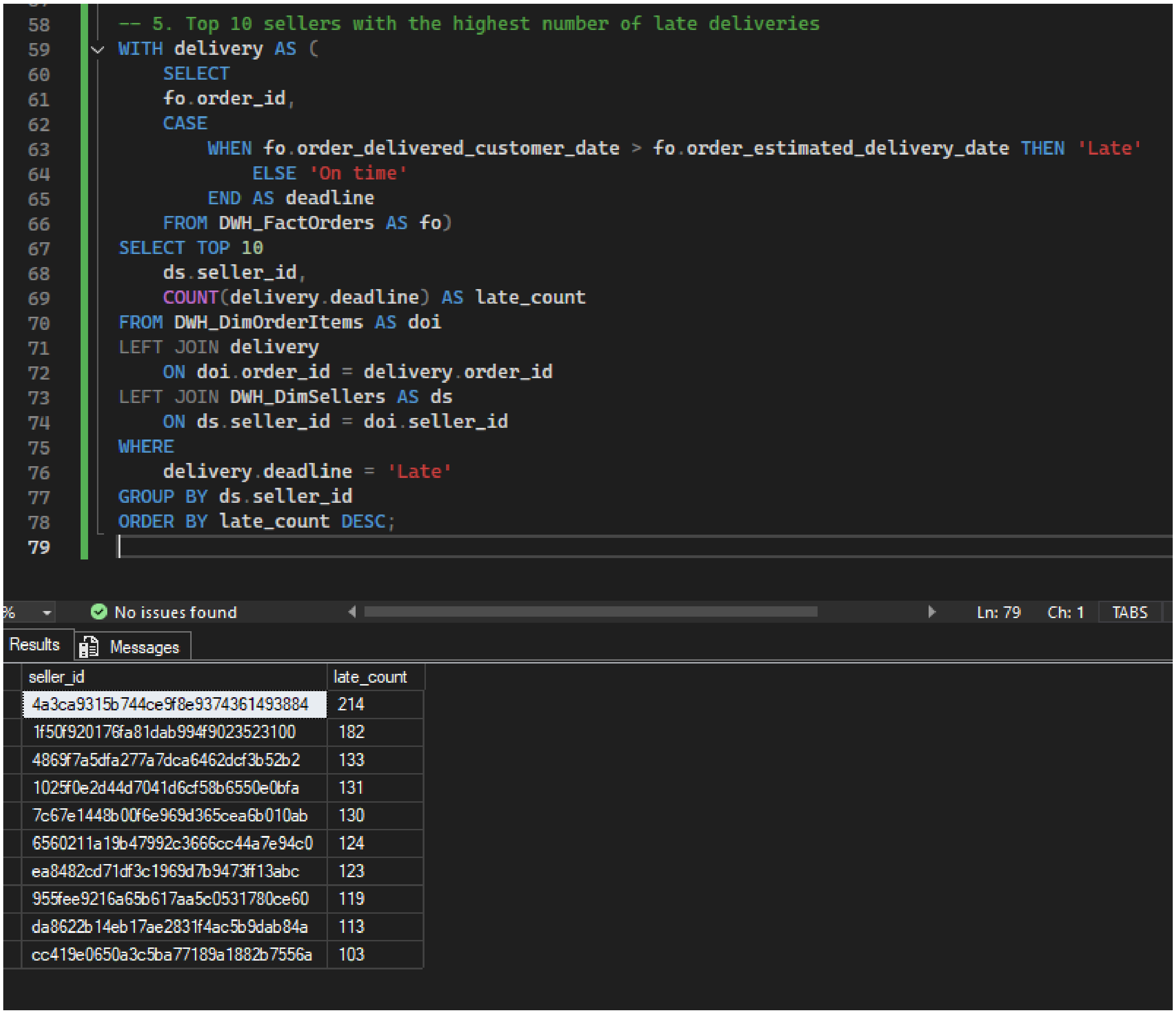


```sql
-- 4. Top 10 product_categories with the higghest reviews

SELECT TOP 10
    dp.product_category_name_english,
    AVG(dor.review_score) AS product_category_review
FROM DWH_FactOrders AS fo
LEFT JOIN DWH_DimOrderItems AS doi
ON fo.order_id = doi.order_id
LEFT JOIN DWH_DimProducts AS dp
ON doi.product_id = dp.product_id
LEFT JOIN DWH_DimOrderReviews AS dor
ON dor.order_id = doi.order_id
WHERE
    dp.product_category_name_english IS NOT NULL
    AND dor.review_score IS NOT NULL
GROUP BY product_category_name_english
ORDER BY product_category_review DESC;
```

product_category_name_english	product_category_review
fashion_childrens_clothes	5
fashion_sport	4
consoles_games	4
cds_dvds_musicals	4
small_appliances	4
garden_tools	4
fashion_underwear_beach	4
arts_and_craftmanship	4
home_appliances	4
housewares	4

5. Top 10 sellers with the highest number of late deliveries



```sql
-- 5. Top 10 sellers with the highest number of late deliveries
WITH delivery AS (
    SELECT
    fo.order_id,
    CASE
        WHEN fo.order_delivered_customer_date > fo.order_estimated_delivery_date THEN 'Late'
            ELSE 'On time'
        END AS deadline
    FROM DWH_FactOrders AS fo)
SELECT TOP 10
    ds.seller_id,
    COUNT(delivery.deadline) AS late_count
FROM DWH_DimOrderItems AS doi
LEFT JOIN delivery
    ON doi.order_id = delivery.order_id
LEFT JOIN DWH_DimSellers AS ds
    ON ds.seller_id = doi.seller_id
WHERE
    delivery.deadline = 'Late'
GROUP BY ds.seller_id
ORDER BY late_count DESC;
```

seller_id	late_count
4a3ca9315b744ce9f8e9374361493884	214
1f50f920176fa81dab994f9023523100	182
4869f7a5dfa277a7dca6462dcf3b52b2	133
1025f0e2d44d7041d6cf58b6550e0bfa	131
7c67e1448b00f6e969d365cea6b010ab	130
6560211a19b47992c3666cc44a7e94c0	124
ea8482cd71df3c1969d7b9473ff13abc	123
955fee9216a65b617aa5c0531780ce60	119
da8622b14eb17ae2831f4ac5b9dab84a	113
cc419e0650a3c5ba77189a1882b7556a	103