**ETL Process for Populating the Gravity Books Data Warehouse Objective**

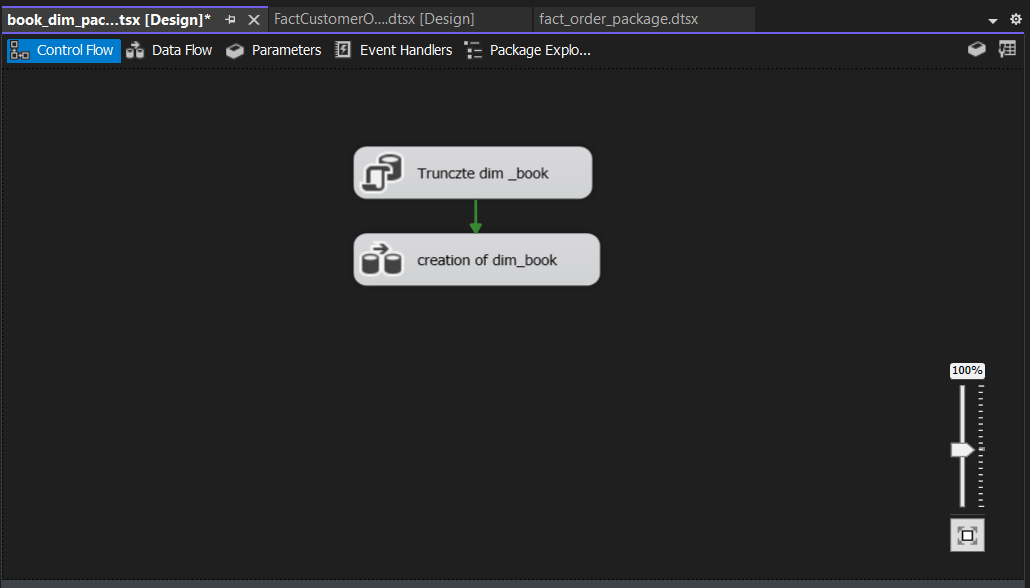
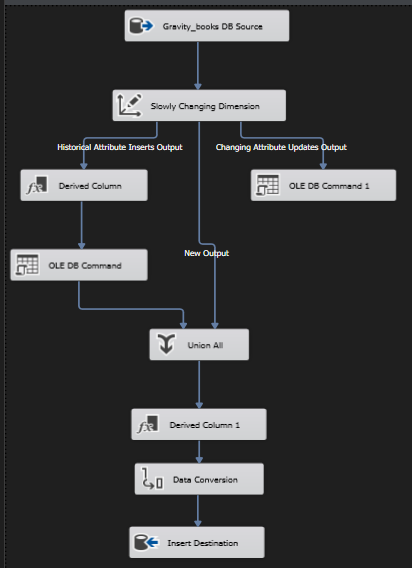
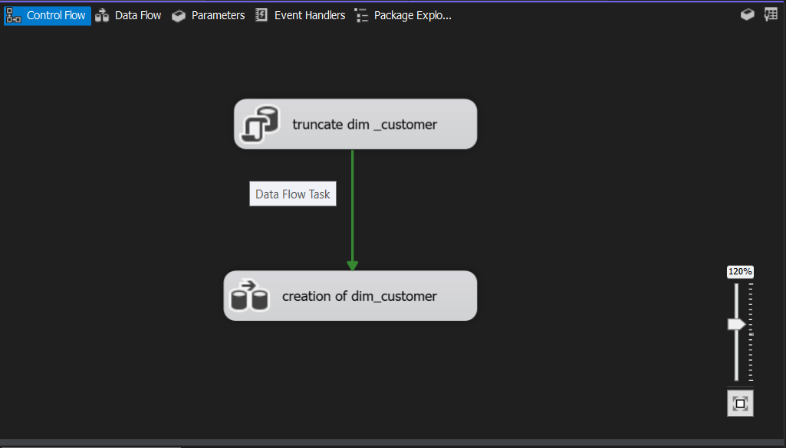
To populate the Gravity Books Data Warehouse (DWH) from the operational database (OLTP system) using SQL Server Integration Services (SSIS), a structured ETL (Extract, Transform, Load) process was implemented. The ETL process involves extracting data from the source, transforming it based on business requirements, and loading it into the target dimensional model. This ensures the data warehouse is updated and accurately reflects the latest business information.

**SSIS Project Design**

The SSIS project consists of four packages:

1. Customer Dimension ETL
2. Book Dimensions ETL
3. Shipping Dimensions ETL
4. Fact Table ETL

**Customer Dimension and Book Dimensions ETL Process**

****

**Data Extraction**

The package begins with an OLE DB Source component to query the relevant data from the OLTP database. The following SQL query retrieves customer and book-related data, including personal details, address information, and status:

SQL Query for customer dimension

SELECT customer.customer\_id,

customer.first\_name,

customer.last\_name,

customer.email,

country.country\_name,

address.street\_name,

address.street\_number,

address.city,

address\_status.address\_status

FROM address

INNER JOIN country

ON address.country\_id = country.country\_id

INNER JOIN customer\_address

ON address.address\_id = customer\_address.address\_id

INNER JOIN address\_status

ON customer\_address.status\_id = address\_status.status\_id

INNER JOIN customer

ON customer\_address.customer\_id = customer.customer\_id;

SQL Query for book dimension

SELECT book.book\_id, author.author\_name, book\_language.language\_code,

book\_language.language\_name, book.title, book.isbn13, book.num\_pages,

book.publication\_date, publisher.publisher\_name

FROM author INNER JOIN

book\_author ON author.author\_id = book\_author.author\_id INNER JOIN

book ON book\_author.book\_id = book.book\_id INNER JOIN

book\_language ON book.language\_id = book\_language.language\_id INNER JOIN

publisher ON book.publisher\_id = publisher.publisher\_id

**Data Transformation**

Key transformations were applied using SSIS components to handle changes in the source data and ensure historical tracking:

1. Slowly Changing Dimension (SCD):
   * Changing Attribute (e.g., First Name): The "First Name" column and “publication date” column was treated as a changing attribute, which means updates in this column overwrite the previous value to reflect the latest information.
   * Historical Attribute (e.g., Street Name): The "Street Name" column and “book title” column was treated as a historical attribute, ensuring any changes in the address are recorded with new rows in the dimension table, tracking the start and end dates for each record.
   * Start and End Dates: These columns identify whether a record is current or expired, supporting time-based analysis.
2. Data Type Conversion: The Data Conversion transformation was used to standardize column data types (e.g., combining Unicode and non-Unicode data types).
3. Derived Columns: Additional columns, such as calculated fields or transformations, were added as needed like end date and start date.

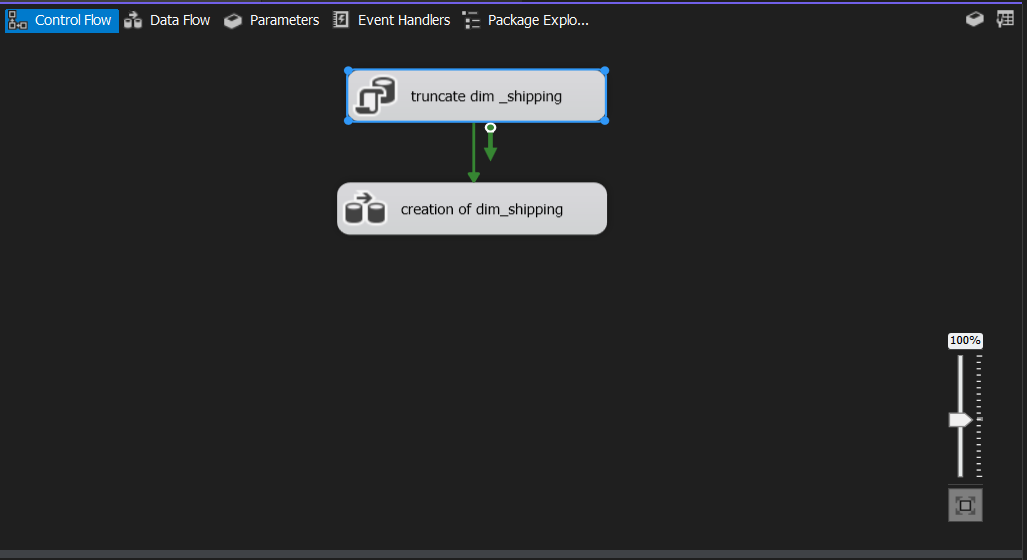
**Data Loading**

The transformed data was loaded into the Customer and book Dimension Table in the data warehouse using the OLE DB Destination component.

**Why These Design Choices?**

1. SCD for Data Accuracy:
   * Changing Attribute: Columns like "First Name”, “publication date” reflect dynamic customer and book information and do not require history tracking.
   * Historical Attribute: Columns like "Street Name”, “book title” track historical changes, allowing analysis of customer and book relocations over time.
2. Start/End Dates: These help in time-based querying and ensure accurate historical tracking for slowly changing attributes.
3. Data Type Conversion: Standardizing data types ensures compatibility between the source system and the target data warehouse.

**Customer Dimension and Book Dimensions ETL Process**

A screenshot of a computer

Description automatically generated

**Data Extraction**

An *OLE DB Source* was used to extract data from the OLTP database with the following

SQL query:

SELECT method\_id, method\_name

FROM shipping\_method

This retrieves the unique *method\_id* and descriptive *method\_name* for shipping methods.

**Data Transformation**

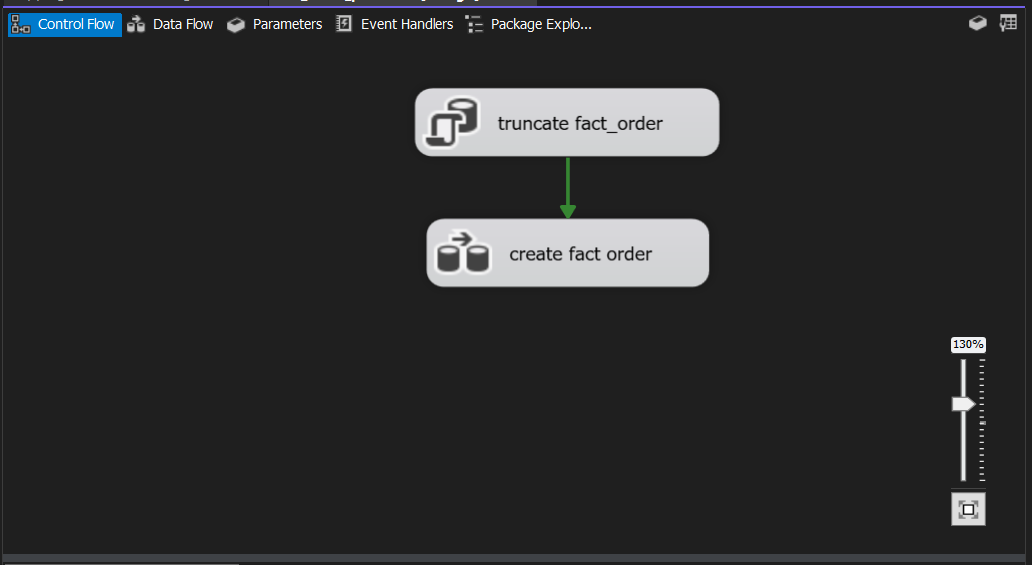
* Slowly Changing Dimension (SCD):
  + *Method Name* was set as a *historical attribute* to track changes. Updates create new records, preserving historical data.
  + *Start and End Dates* were added to identify current and expired records.
* Data Conversion: Ensured compatibility between Unicode and non-Unicode data.

**Data Loading**

The transformed data was loaded into the *Shipping Dimension Table* using the *OLE DB Destination* component.

**Fact Table ETL Process:**

The final ETL package populates the *Fact Table (Order Facts)* in the Gravity Books Data Warehouse (DWH). This step integrates data from the transactional system, linking it with all dimensions and calculating key metrics for analysis.

A screenshot of a computer

Description automatically generated

**Data Extraction**

An *OLE DB Source* component is used to extract the necessary data from the OLTP database. The following SQL query retrieves order details, including customer, book, shipping method, and order status:

SQL Query

SELECT

co.customer\_id, b.book\_id, co.shipping\_method\_id,

CAST(format(co.order\_date,'yyyy-MM-dd 00:00:00.000')

AS Datetime) AS order\_date ,

co.order\_id ,ol.line\_id , status\_value ,

ol.price, s.cost

FROM

cust\_order AS co LEFT OUTER JOIN order\_line AS ol ON co.order\_id = ol.order\_id LEFT OUTER JOIN

book AS b ON ol.book\_id = b.book\_id LEFT OUTER JOIN

shipping\_method AS s ON s.method\_id = co.shipping\_method\_id

LEFT OUTER JOIN order\_history as oh on co.order\_id = oh.order\_id

LEFT OUTER JOIN order\_status as os on oh.status\_id = os.status\_id

**Data Transformation**

1. *Lookup Components:*
   * Four *Lookup Components* are used to match business keys from the OLTP database with surrogate keys in the dimensions:
     + *Customer Dimension*
     + *Book Dimension*
     + *Shipping Method Dimension*
     + *Date Dimension*

These ensure the fact table references dimension tables through surrogate keys.

1. *Slowly Changing Dimension (SCD):*
   * The *SCD Component* is used to handle updates in the OLTP database. Any changes in fact table attributes are updated accordingly to maintain data accuracy.
2. *Start and End Dates:*
   * Start and end dates are added to track the validity of records, supporting historical analysis.
3. *Data Conversion:*
   * A *Data Conversion* transformation standardizes data types (e.g., Unicode vs. non-Unicode) to ensure compatibility between the source and target systems.

**Data Loading**

The transformed data is loaded into the *Order Fact Table* in the data warehouse using an *OLE DB Destination* component.