Sri Lanka Institute of Information Technology

Data Warehousing and Business Intelligence

IT3021

Assignment 1

2022

Assignment 1 Report

Student Name – Jayasooriya C. A

IT Number – IT20250942

Table of Contents

1 Dataset Selection

1.1 Description

Dataset Selected – Insurance Claims Fraud Data (click on the text to view the original dataset)

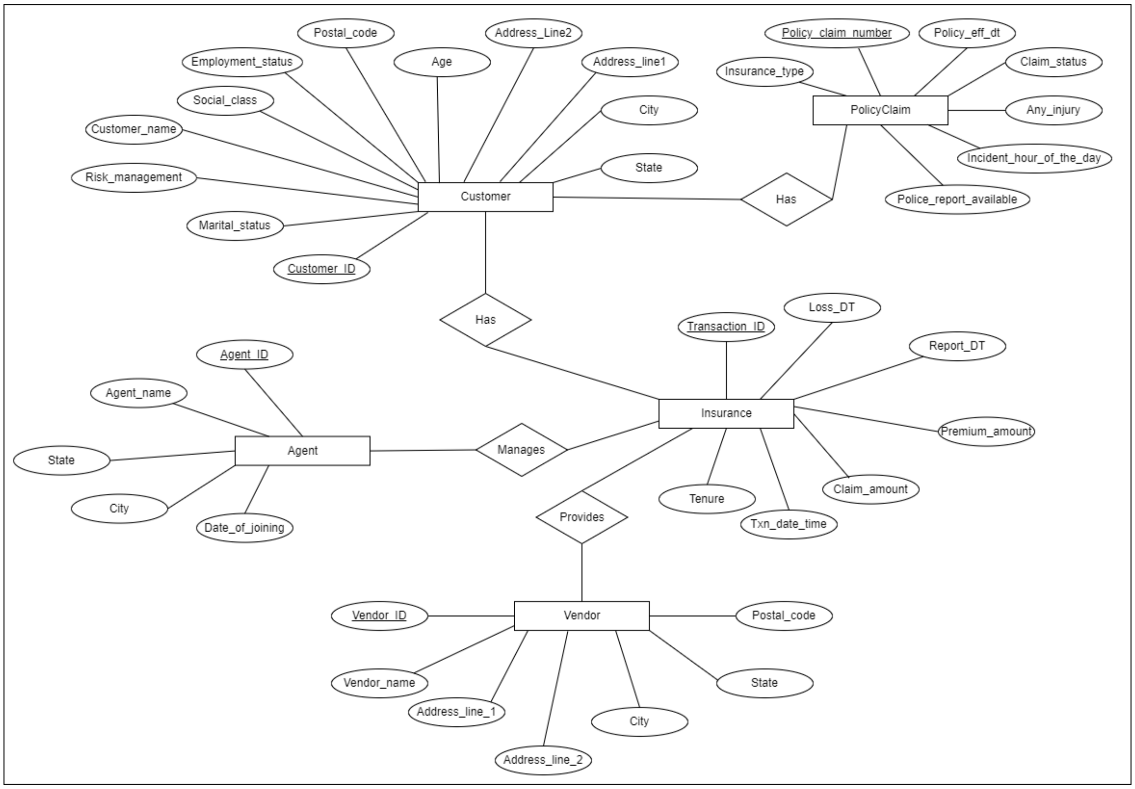
Description –

The Insurance Claims Fraud Dataset contains data that can be used in the process of insurance fraud detection. The original dataset contains a one-year worth data, from 2020/06/01 till 2021/06/30 in three data files which contains the Insurance, Vendor, and Employee data tables.

The dataset chosen gives a multidimensional view of the insurance data from the perspectives of the Customer, Agent (Employee), and the Vendor (the Insurance Company).

The original data tables of the data set have been edited, configured, and rearranged to suit the requirements of the project. Hence 5 data tables have been identified:

1. Insurance – Contains the insurance transaction data related to the insurances.
2. Agent – This contains the data about insurance agents who are involved in the managing and handling of customer insurance.
3. Customer – Contains the details about the customer or the holders of an insurance policy.
4. Vendor – Contains the details of the insurance service provider or the insurance company.
5. PolicyClaim – Contains the details of policy claims made by customers.



2 Preparation of the Data Sources

Initially, the original three data files were in the csv format. Then after they were downloaded and separated into five tables in five data files. These files were saved in different data formats.

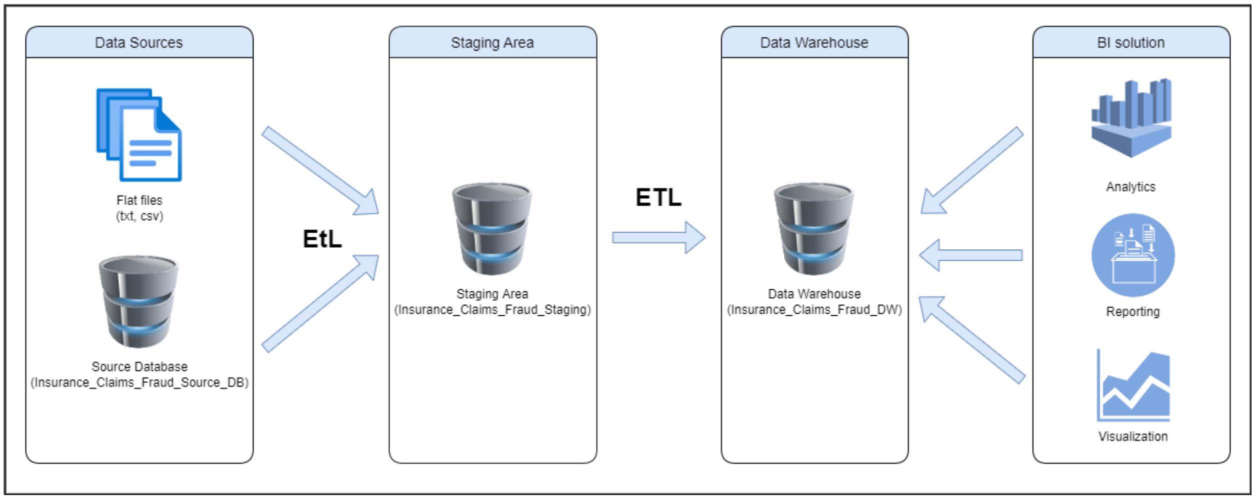
Three types of data sources were utilized: cvs, txt, database.

1. .csv –
   The Agent data were kept in the csv source type file (Agent.csv).
   This contains the data about insurance agents who handle the client / customer insurance policies.

2. .txt –
   The Policy Claims data were saved in a txt source type file (PolicyClaim.txt).
   This contains all the details of the insurance policy claims that have been made by customers.

3. Database –
   A source database (Insurance_Claims_Fraud_Source_DB) was created by importing the Insurance.csv, Customer.csv and Vendor.csv files.
   - Insurance – This table contains all the transactional details related to an insurance transaction.
   - Customer – This table contains all the details related to an insurance policy holder / owner of and insurance.
   - Vendor – This tables contains the details of the insurance service provider of the company that offers the insurance cover to the customer.

3 Solution Architecture



The above is a high-level Data Warehousing and Business Intelligence architectural solution for the chosen dataset and topic. We can identify for main layers here:

1. Data Sources –
   The first stage in establishing a solid architecture is to collect data from many data sources, such as CRM, ERP, databases, files, or APIs, depending on the goals and resources available [1].
   For the given scenario, there are 2 data sources, a source database, and flat files (csv and txt)
   - Source database (Insurance_Claims_Fraud_Source_DB) which contains the customer, vendor, and insurance data tables.
   - CSV (Agent.csv) file contains the agent details.
   - TXT (PolicyClaim.txt) file contains the policy claims details.

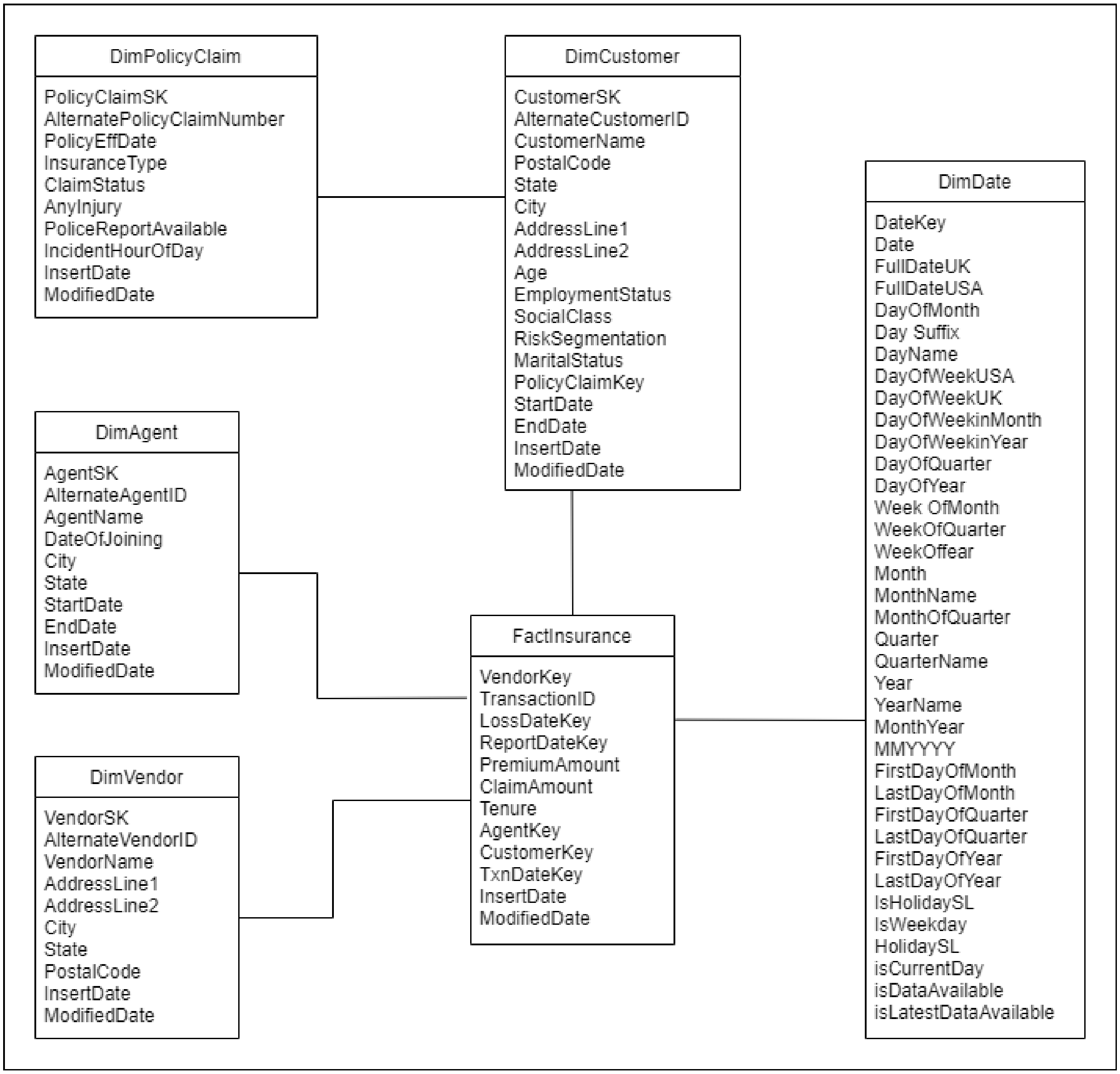
2. Staging area –
   A data staging area acts as a temporary storage facility between the data sources and the data warehouse [2]. The staging area is primarily used to extract data quickly from its data sources while minimizing the effect of the sources.
   In the given scenario a database named Insurance_Claims_Fraud_Staging acts as the data staging area

3. Data warehouse –
   A data warehouse is a large collection of business data that is used to enhance internal decision-making [3]. This has a lot of historical information.
   In this scenario a database file named Insurance_Claims_Fraud_DW is used as the data warehouse. The said data warehouse comprises of 5 dimensional tables and 1 fact table.

4. BI solution (consumption) –
   This employs technology and services to transform data into actionable insights that assist organizations in making better decisions [4].
   In this scenario's business solution, the data could be analyzed, visualized, and reported to understand the insurance frauds and to take actions necessary business decisions to mitigate them and take other necessary informed decisions.

5. ETL –
   ETL (Extract, transform, and load) is a data integration procedure that integrates data from several sources into a single, consistent data store that can subsequently be fed into a data warehouse or other destination system [5].

4 Data Warehouse Design and Development



The above is the dimensional model used for the given scenario. In summary the dimension model is designed with 5 dimensional tables (including the date dimension) and a single fact table.

- **The schema used – Snowflake Schema**
  Snowflake schema has been utilized in the dimensional modelling to reduce redundancy through normalization. As visible the customer dimension table has been normalized.

- It was assumed that the address details would provide a greater benefit in categorizing and analyzing the data in various ways, hence the customer, vendor and agent contains hierarchical attributes describing the respective attributes.
    - State > Postal code > City > AddressLine1 > Addressline2

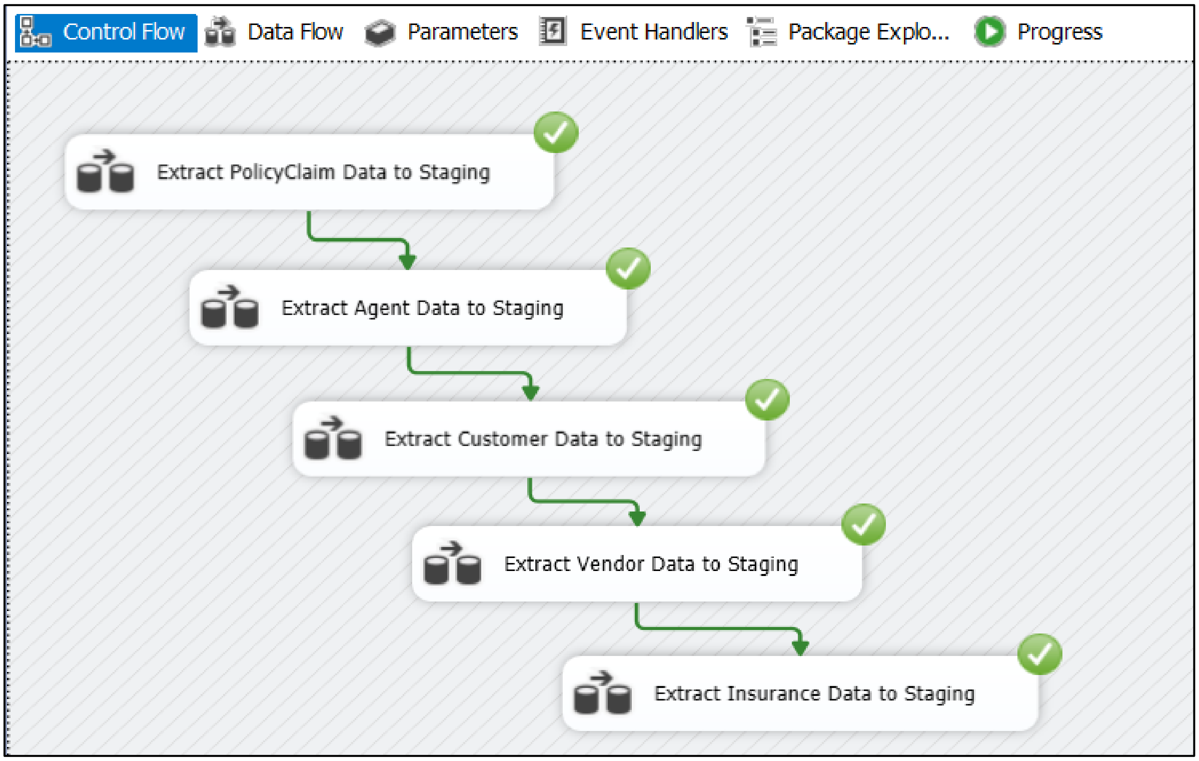
- **Dimension and Fact Tables**
  Five dimension tables and a fact table was created:
    1. DimPolicyClaim – The policy claim dimension table contains the policy claim details. PolicyClaimsSK is the surrogate key.
    2. DimCustomer – The customer dimension contains insurance policy holder / customer details. CustomerSK is the surrogate key.
    3. DimAgent – Contains details of insurance agents who manages the customer insurances. AgentSK is the surrogate key.
    4. DimVendor – Contains insurance provider details. VendorSK is the surrogate key.
    5. DimDate – This is a common dimension. DateKey is the surrogate key. An SQL script was used to generate the date dimension.
    6. FactInsurance – Contains all the transactional data. References dimension tables via foreign keys.

- **Slowly changing dimensions –**
  It was assumed that the Customer and Agent could change certain details over time. Hence DimCustomer and DimAgent were considered as Slowly Changing Dimensions
  Assumption – It was assumed that as agents and customers are people, they have the ability change various attributes that belong to them (ex: An agent can change his/her city, a customer can change address, marital status and etc.), hence considering this Agent and Customer dimensions were chosen as slowly changing dimensions.
    - DimAgent – City and State are historical attributes.
    - DimCustomer – City, State, PostalCode, AddressLine1, AddressLine2, EmploymentStatus, SocialClass, RiskSegmentation and MaritalStatus are historical attributes.

5 ETL Development

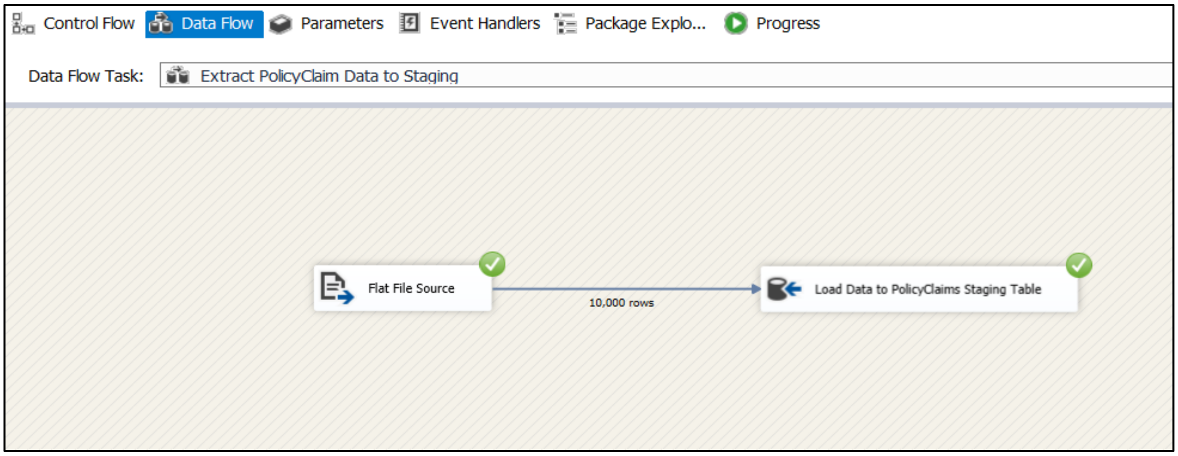
5.1 Extract Data from Source to Staging

- Order of Execution

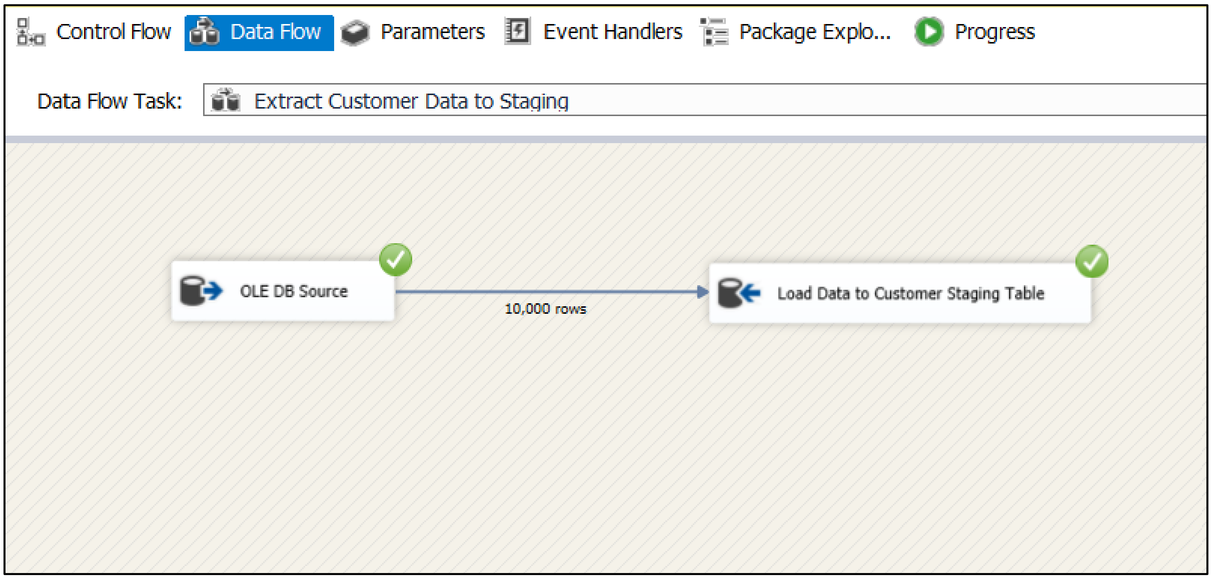

Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Extract PolicyClaim Data to Staging
Extract Agent Data to Staging
Extract Customer Data to Staging
Extract Vendor Data to Staging
Extract Insurance Data to Staging

Data is extracted from various data sources and is staged in an intermediate location until being loaded into the data warehouse. Individual extractions into the staging database happens as below images:

- Policy Claim data are extracted from a flat file source and is loaded into staging area.

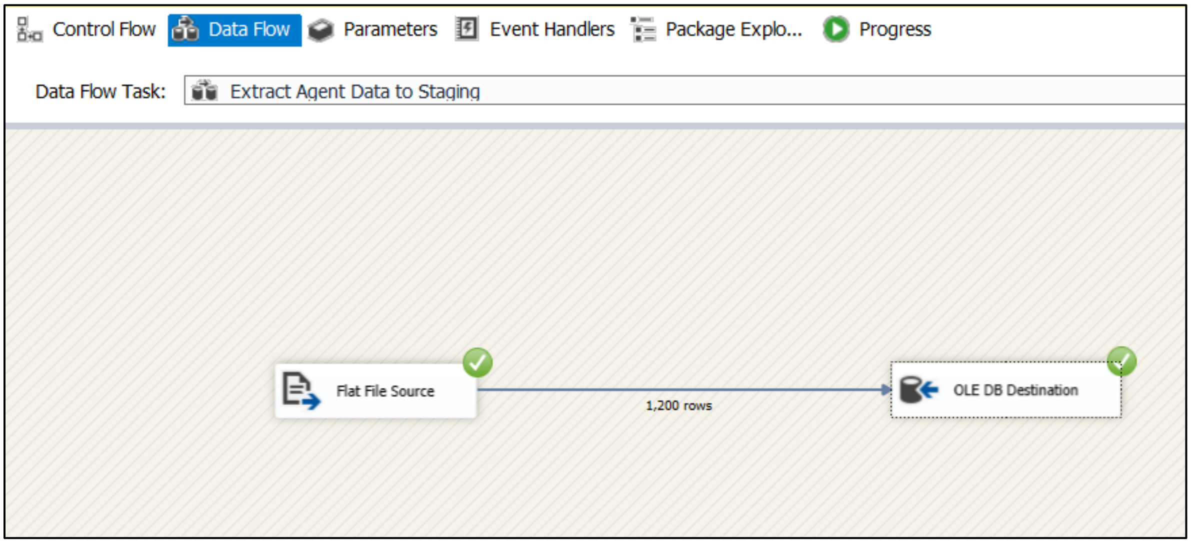


- Customer data are extracted from the customer data table of a source database.



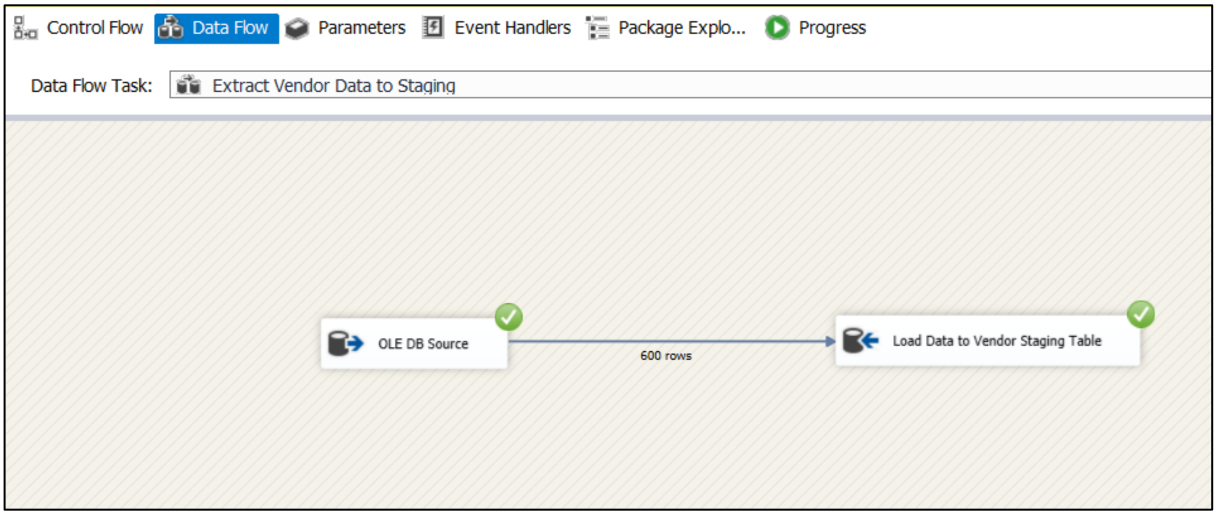
Data Flow Task: Extract Customer Data to Staging

- Agent data are extracted from another flat file source.

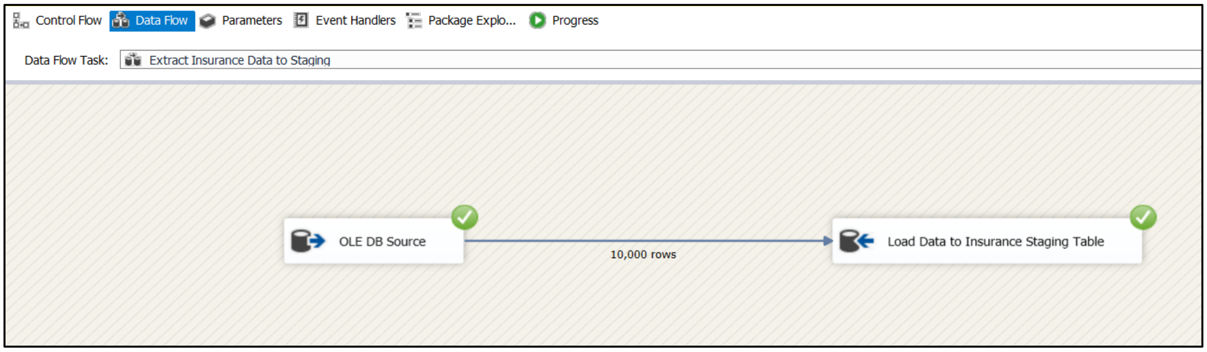


Data Flow Task: Extract Agent Data to Staging

- Vendor data are extracted from the customer data table of a source database.

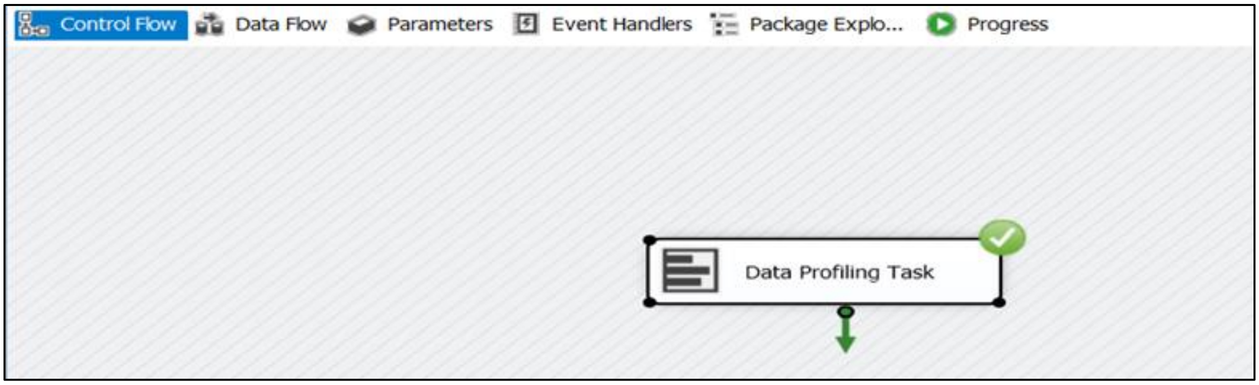


- Insurance data are extracted from the Insurance data table of the source database.



5.1.1 Data Profiling Package

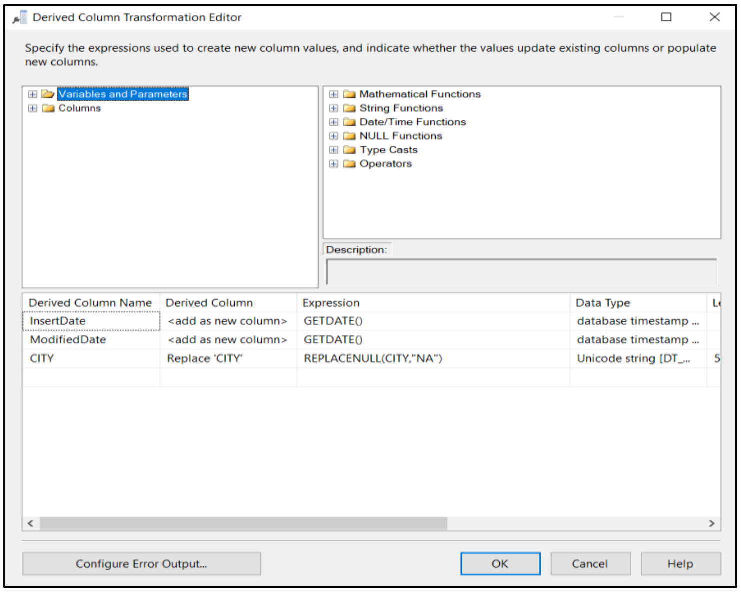
The data profiling for all the staged data are done in one data profiling task. This provides an in-depth understanding about the data that has been staged which gives us the necessary information to handle the transformations of the data in the upcoming stages.



5.2 Transforming the Staged Data

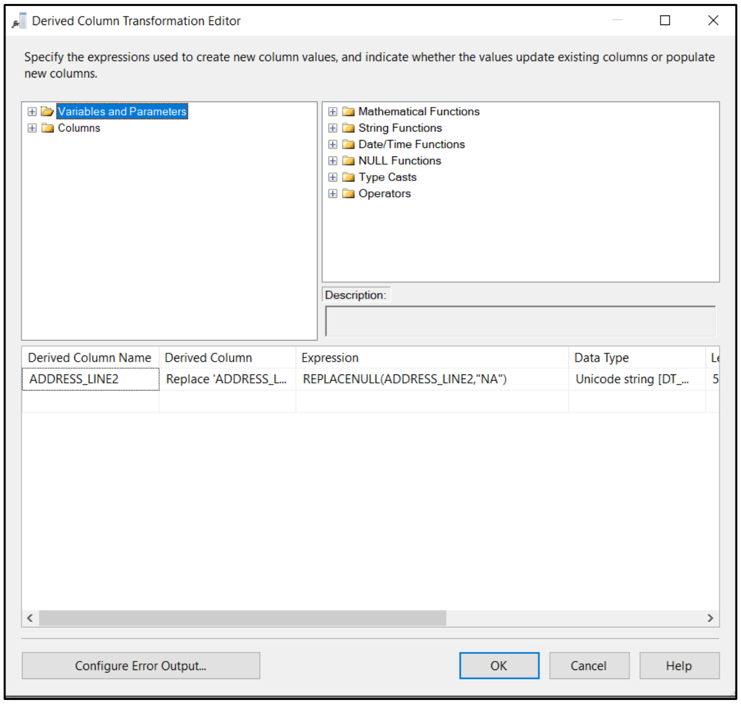
The staged data then undergoes some transformations before they get loaded into the data warehouse as shown below:

- The following transformations are done to handle the NULL values of the city and addressline2 fields where they are found and replaced. Insertions of the modified and insert date are also assigned to obtain the system dates during insertion and modification.



Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
InsertDate	<add as new column>	GETDATE()	database timestamp ...	
ModifiedDate	<add as new column>	GETDATE()	database timestamp ...	
CITY	Replace 'CITY'	REPLACENULL(CITY,"NA")	Unicode string [DT_...	5



Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
ADDRESS_LINE2	Replace 'ADDRESS_L...	REPLACENULL(ADDRESS_LINE2,"NA")	Unicode string [DT_...	5

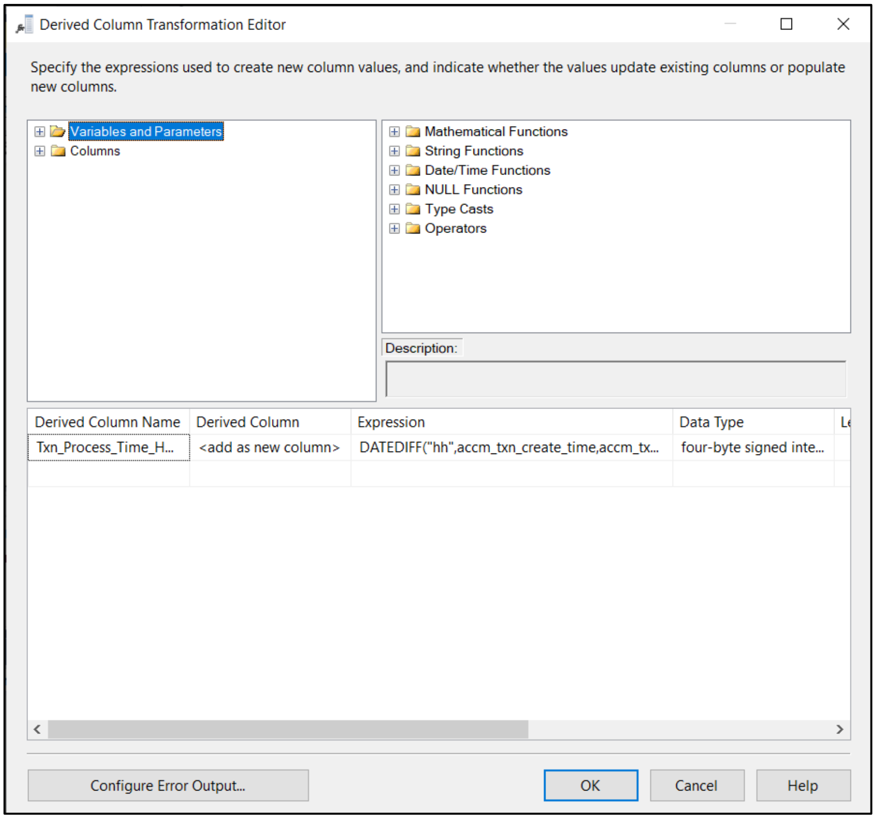
- Lookups have been used to create foreign key references between tables and to map the proper surrogate keys of the referring dimensional table as the foreign key.

- Only the required columns in creating dimensions have been chosen properly and unwanted outliers have been filtered and proper data filtering has been done.

- Derived columns have been used to add the insertion and modified date columns.

- Rounding-off the premium amount floating point number to the nearest 3 decimal places using a derived column.

- Calculation of the process time in hours have been done with the difference of the create time and complete time columns.

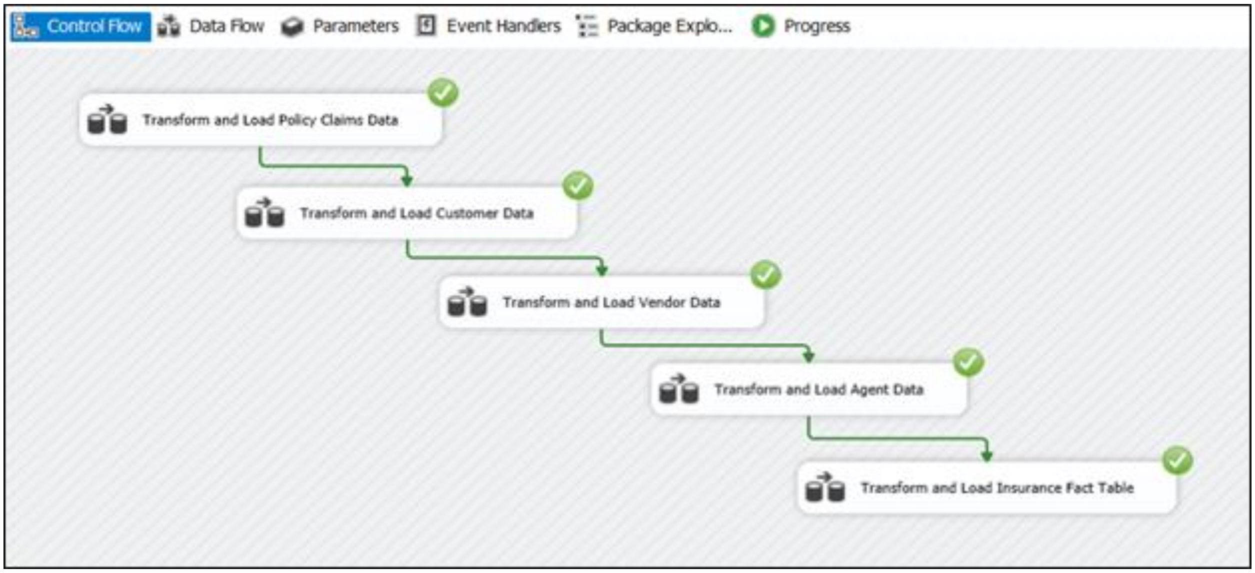


Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
Txn_Process_Time_H...	<add as new column>	DATEDIFF("hh",accm_txn_create_time,accm_tx...	four-byte signed inte...	

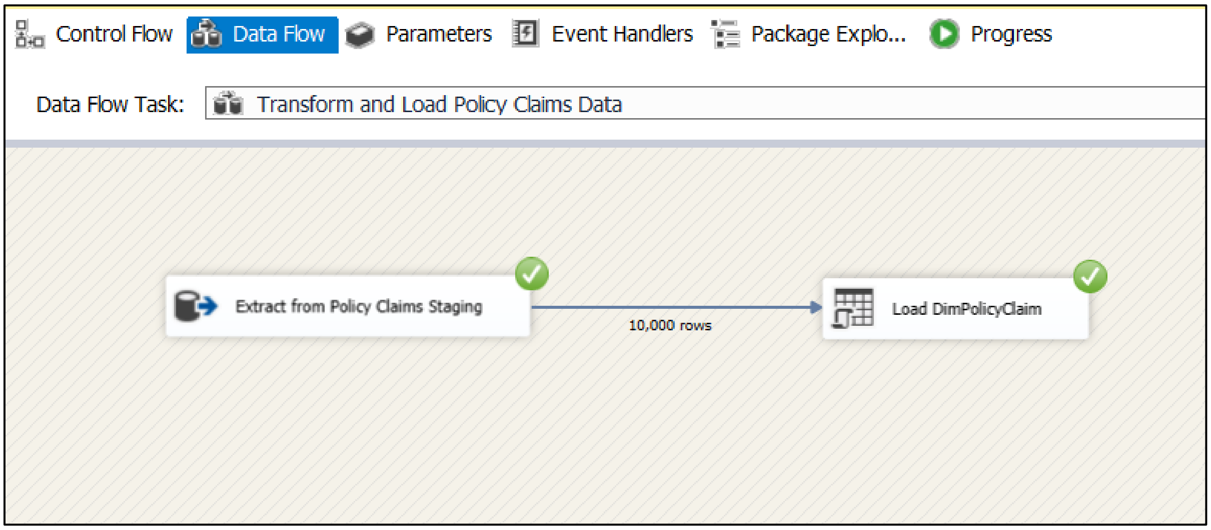
5.3 Loading the Transformed Data into the Data Warehouse

- Order of Execution


Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Transform and Load Policy Claims Data
Transform and Load Customer Data
Transform and Load Vendor Data
Transform and Load Agent Data
Transform and Load Insurance Fact Table

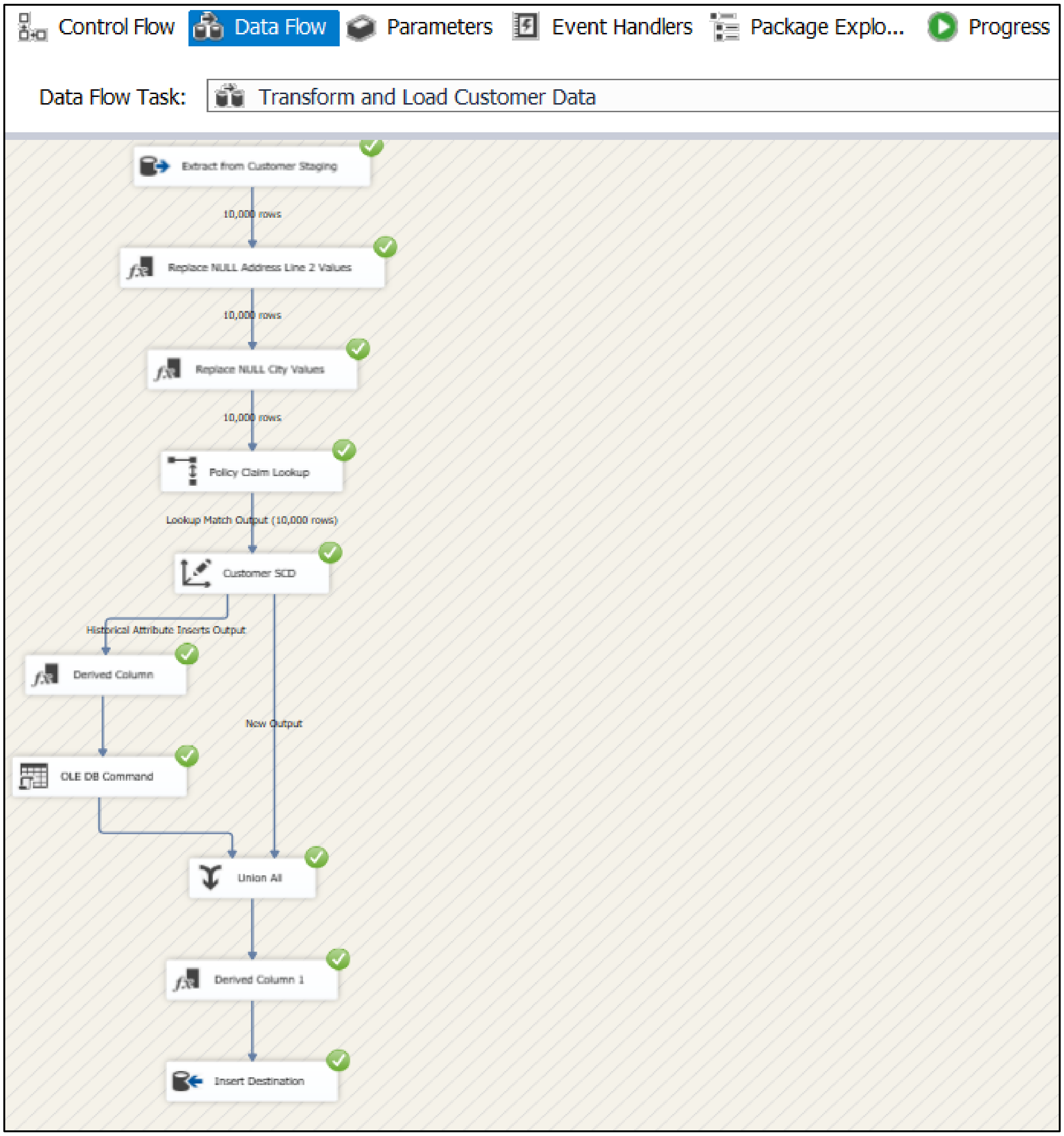
After the necessary transformations have been done to the data they get loaded into the data base to conclude the ETL process.

- Loading the transformed policy claims into the data warehouse.

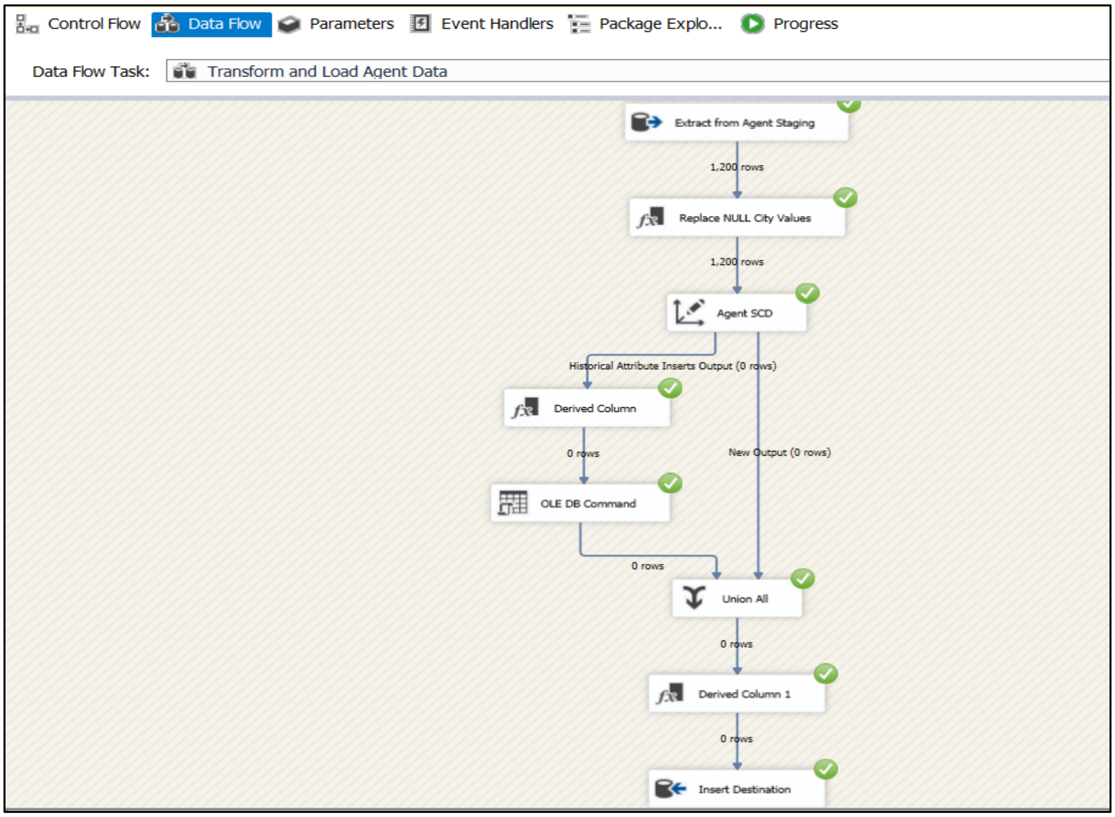


Data Flow Task: Transform and Load Policy Claims Data

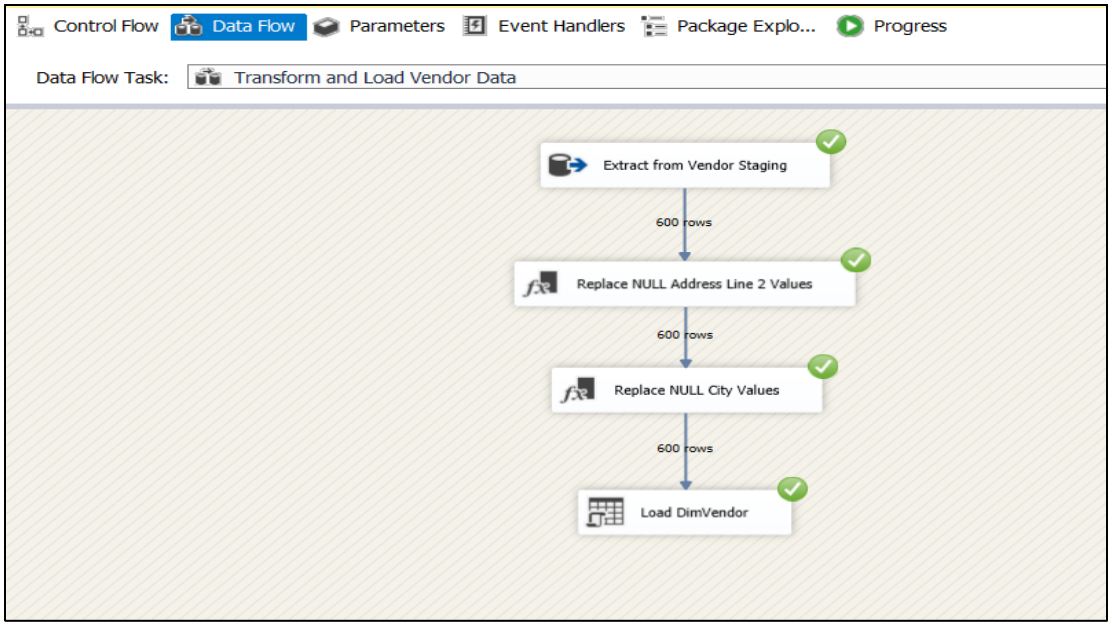
- Loading the transformed customer data as a slowly changing dimension into the date warehouse.



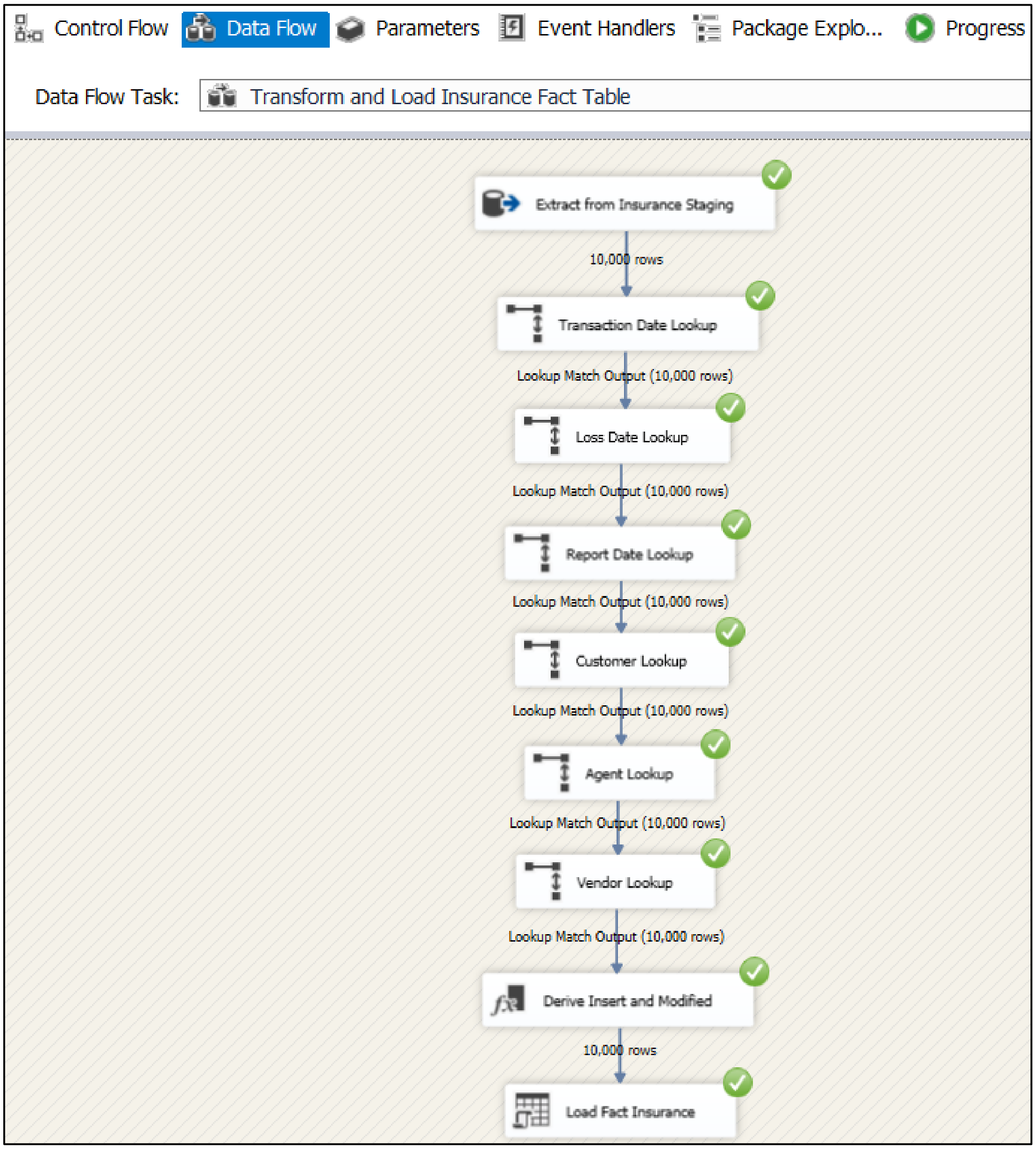
- Loading the transformed agent data as a slowly changing dimension into the date warehouse.



- Loading the transformed vendor data into the data warehouse.

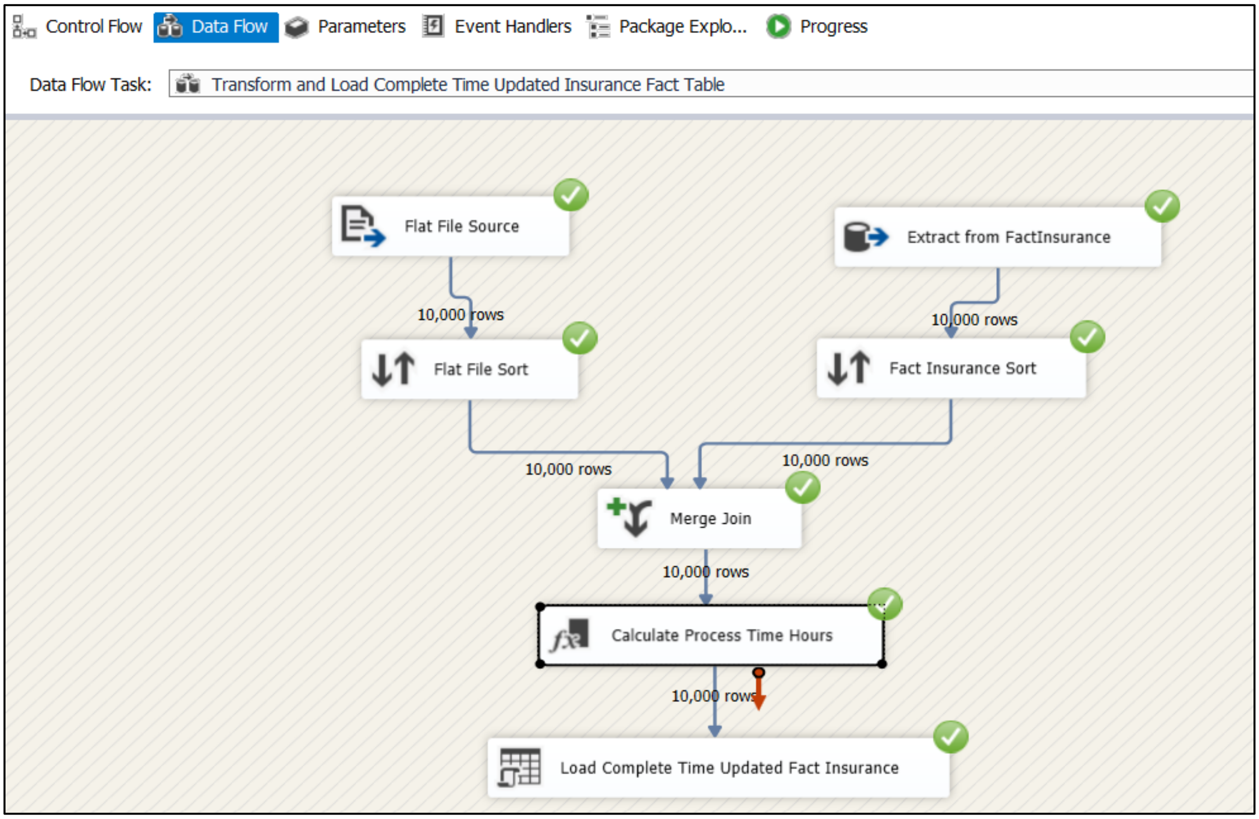


- Loading the transactional data of this scenario into the data warehouse as a fact table (FactInsurance).


Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load Insurance Fact Table
Extract from Insurance Staging
10,000 rows
Transaction Date Lookup
Lookup Match Output (10,000 rows)
Loss Date Lookup
Lookup Match Output (10,000 rows)
Report Date Lookup
Lookup Match Output (10,000 rows)
Customer Lookup
Lookup Match Output (10,000 rows)
Agent Lookup
Lookup Match Output (10,000 rows)
Vendor Lookup
Lookup Match Output (10,000 rows)
Derive Insert and Modified
10,000 rows
Load Fact Insurance

6 ETL Development – Accumulating Fact Tables

The process time in hours is calculated with the difference of create time and complete time and updated into the present fact table. This task is performed in a separate package where a derived column calculates the process time and updates the txn_process_time_hours column and the accm_txn_complete_time columns with the derived data and the data from the external flat file.

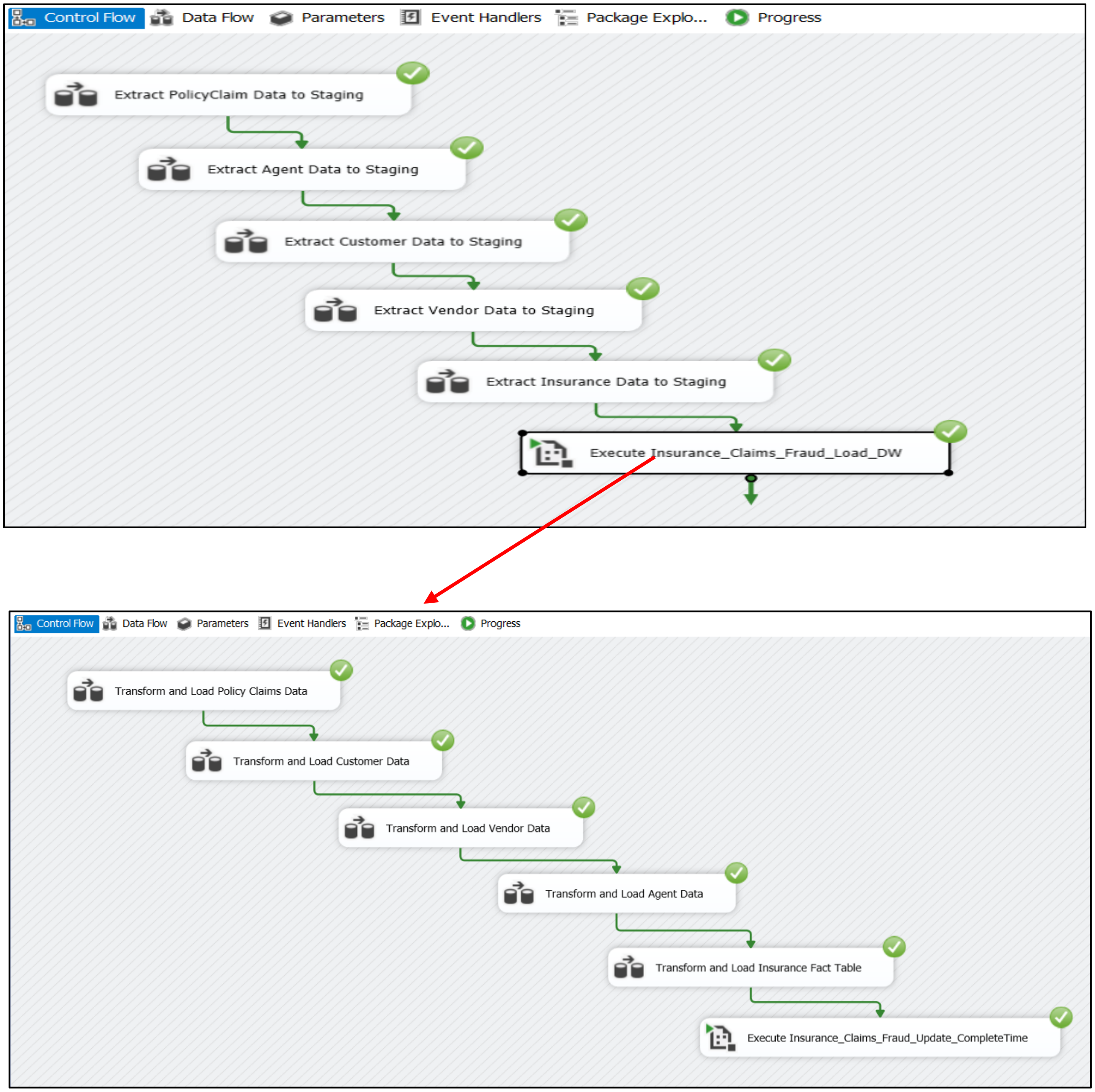


7 Overall Execution Flow of the Total Solution

The following is an overall execution of the staging data flow tasks, followed by the execution of the data loading tasks to the data warehouse.

After the completion of the data loading, the final data flow task in the data loading package is set as a package execution task of the time update package, where it will be executed to update the fact table with the transaction complete times and the process times in hours.

This enables the proper flow execution of the data staging and then the execution of loading the staged data into the data warehouse and the updating of the fact table with the proper data in proper order.


Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Extract PolicyClaim Data to Staging
Extract Agent Data to Staging
Extract Customer Data to Staging
Extract Vendor Data to Staging
Extract Insurance Data to Staging
Execute Insurance_Claims_Fraud_Load_DW
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Transform and Load Policy Claims Data
Transform and Load Customer Data
Transform and Load Vendor Data
Transform and Load Agent Data
Transform and Load Insurance Fact Table
Execute Insurance_Claims_Fraud_Update_CompleteTime

8 References

[1] "sas.com," [Online]. Available: https://www.sas.com/en_in/insights/data-management/data-warehouse.html.

[2] T. Christiansen, "timextender.com," [Online]. Available: https://support.timextender.com/hc/en-us/articles/210438083-What-is-a-Data-Staging-Area.

[3] "talend.com," [Online]. Available: https://www.talend.com/resources/what-is-data-warehouse.

[4] "heavy.ai," [Online]. Available: https://www.heavy.ai/technical-glossary/business-intelligence.

[5] "ibm.com," [Online]. Available: ibm.com/cloud/learn/etl.