Sri Lanka Institute of Information Technology

Data Warehousing and Business Intelligence IT3021

Assignment 1

2025

Assignment 1 Report

Student Name – Withanage W W D T H

IT Number – IT22223012

Contents

1 Dataset Selection

1.1 Description

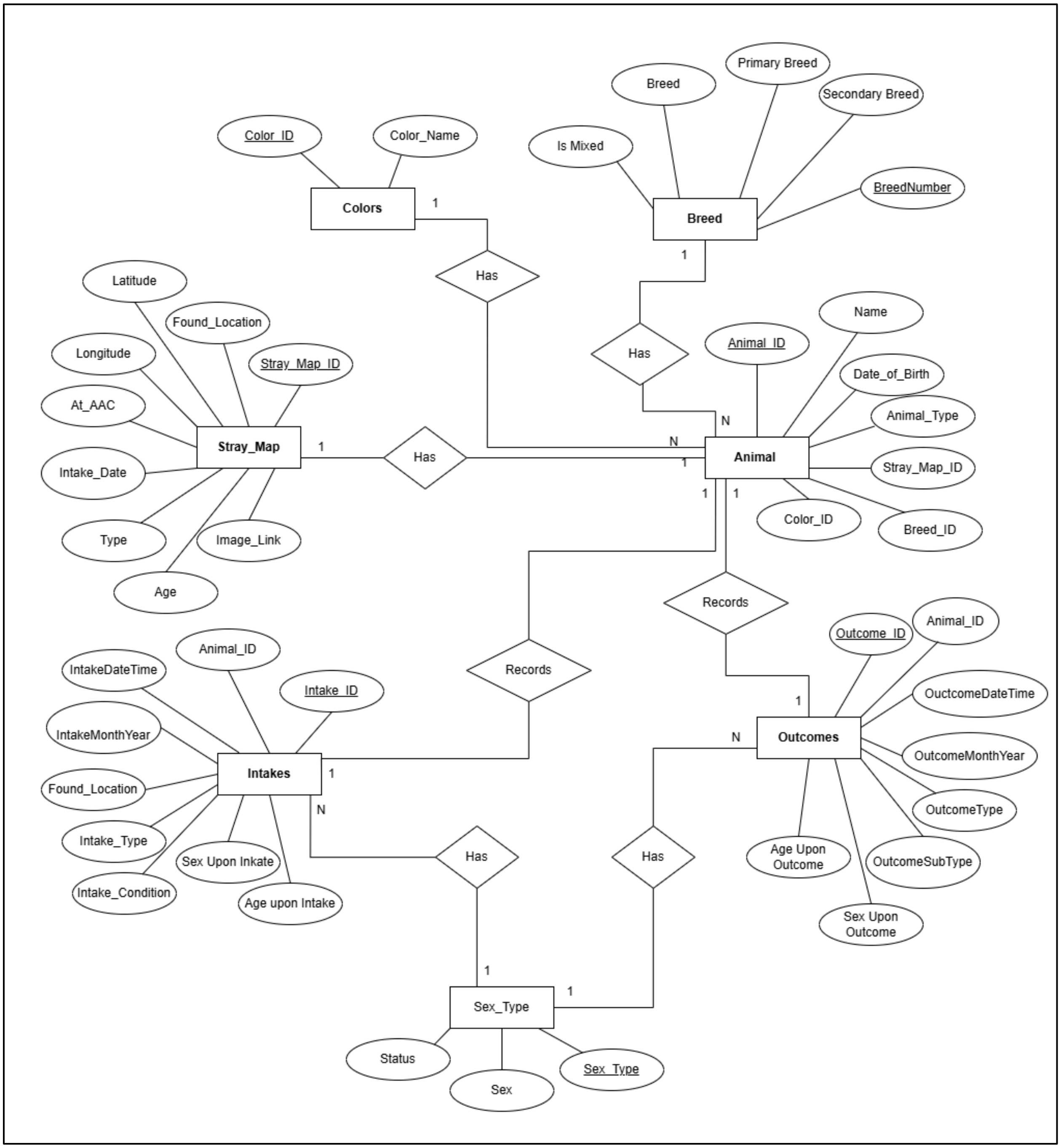
Dataset -Austin Animal Center(link to original dataset)

Description

The Animal Center dataset is designed to support the analyze and monitor activities related to animal intake, sheltering, and outcomes (such as adoptions or transfers). This dataset provides a well-rounded view of the shelter's operations, focusing on key aspects such as the animals themselves, how and when they enter or leave the shelter, the locations involved, and details like breed, color, and sex.
This dataset cover data from October 1, 2013 to March 3, 2021, spanning around eight years.
The original dataset contains three files. They are Austin Animal Center Intakes, Austin Animal Center Stray Map, and Austin Animal Center Outcomes.

The original data tables of the data set have been edited, configured, and rearranged to suit the requirements of the project. Hence 7 data tables have been identified:

1. Animal- Stores unique information about each animal, such as ID, name, date of birth, and general attributes.
2. Intakes- Records intake events when an animal is brought into the center. Includes how, when, and why the animal arrived.
3. Outcomes- Records outcome events when an animal leaves the center, including the type (e.g., adoption, transfer)
4. Stray Map- Provides details of animals found as strays, including found location, intake date, and currently in animal center or not.
5. Colors- A table containing standardized color values used to describe animal appearances.
6. Breed- A table of breeds to categorize animals for reporting and filtering.
7. Sex Type- A table that stores standardized values for sex-related data, such as "Neutered Male", "Spayed Female", "Intact Male".


Primary Breed
Breed
Secondary Breed
Color_ID
Color_Name
Is Mixed
BreedNumber
Colors
1
Breed
1
Has
Latitude
Name
Found_Location
Animal_ID
Longitude
Stray_Map_ID
Has
Date_of_Birth
At_AAC
N
Animal_Type
Stray_Map
1
Has
N
Animal
1
Intake_Date
Stray_Map_ID
1
1
Color_ID
Breed_ID
Type
Image_Link
Age
Records
Animal_ID
Outcome_ID
Animal_ID
IntakeDateTime
Records
Intake_ID
1
OuctcomeDateTime
IntakeMonthYear
N
Outcomes
Intakes
1
OutcomeMonthYear
Found_Location
N
OutcomeType
Intake_Type
Has
Has
Age Upon Outcome
Sex Upon Inkate
OutcomeSubType
Intake_Condition
Age upon Intake
Sex Upon Outcome
1
1
Sex_Type
Status
Sex_Type
Sex

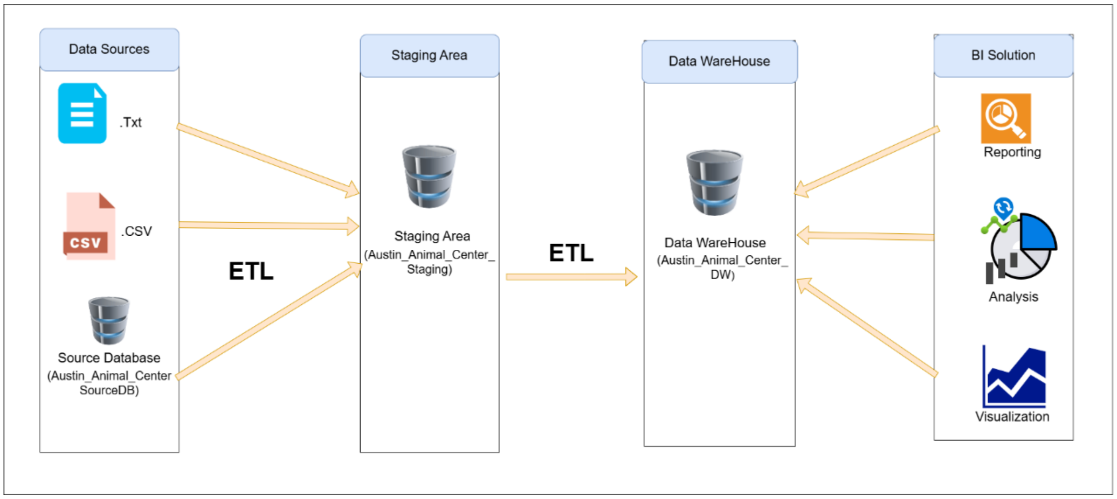
2 Preparation of the Data Sources

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

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

1. .csv
   The Colors data were kept in the csv source type file (Colors.csv).
   This containing standardized color values used to describe animal.
2. .txt
   The Sex type data were saved in a txt source type file (Sex_Type.txt).
   This contains standardized values for sex-related data.
3. Database
   A source database (Austin_Animal_CenterSourceDB) was created by importing the Breed.csv, Animal.csv, Outcome.csv, Stray_Map.csv and Intakes.csv files.
   - Breed - A table of breeds to categorize animals for reporting and filtering.
   - Animal - This table contains Stores unique information about each animal.
   - Outcome -. This table records outcome events when an animal leaves the center
   - Stray_Map-This Provides details of animals found as strays
   - Intakes- Records intake events when an animal is brought into the center.

3 Solution Architecture



The Austin Animal Center is a real-world dataset that contains comprehensive records of animals that entered the animal shelter. Above diagram represents high-level Data Warehousing and Business Intelligence architectural solution for given dataset. There are four main layers.

1. Data Sources:
   The Data Sources layer is the first and most crucial step in any Data Warehousing and Business Intelligence (BI) architecture. Data sources collect data from multiple origins (structured or unstructured) and serve it as input for the ETL (Extract, Transform, Load) process. In Austin Animal Center,
   - Source database (Austin_Animal_CenterSourceDB) it has Animal, Outcome, Intakes, Breed and Stray_Map tables.
   - CSV (Colors.csv)
   - TXT (Sex_Type.txt)

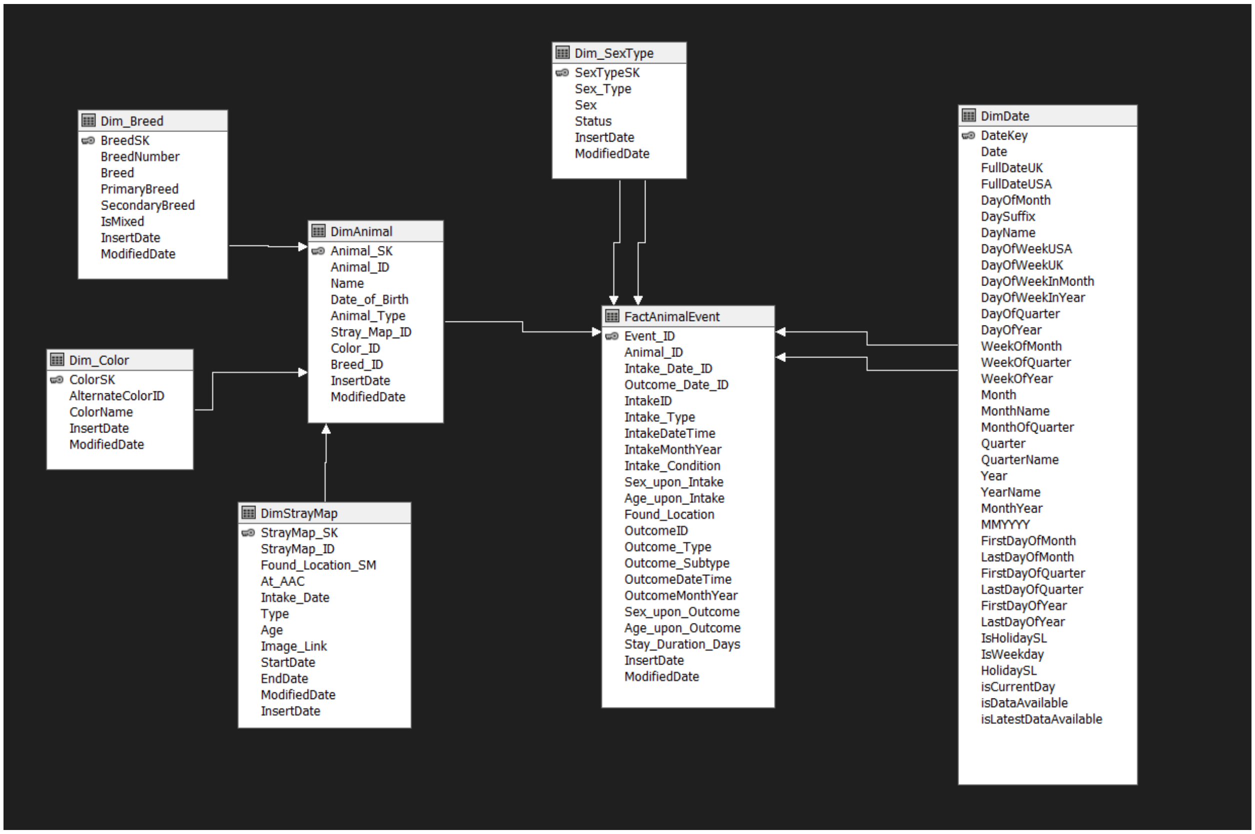
2. Staging Area:
   The Staging Area serves as a temporary workspac**e** where data from various sources is collected and prepared before moving into the data warehouse. It acts as an intermediate layer that helps ensure integrity, consistency, and quality of data before it is permanently stored and used for analysis. In this case Austin_Animal_Center_Staging used as a staging area.

3. Data warehouse:
   A Data Warehouse is a central system used to store, organize, and manage historical **data** from multiple sources. It helps businesses perform reporting and analysis by providing clean, consistent, and structured data for better decision-making. There are six dimension table and one fact table in Austin_Animal_Center_DW.

4. BI solution:
   Business Intelligence (BI) solution refers to the tools and technologies used to analyze and visualize data stored in the data warehouse. It helps organizations to make informed, data-driven decisions. For example, animal center data warehouse can be used to Analyze trends such as adoption rates by breed, intake conditions over time

5. ETL
   ELT stands for Extract, Load, Transform. is a data integration process where data is first extracted from various source systems, then loaded directly into the target system such as a data warehouse, and finally transformed or cleaned within the target system

4 Data Warehouse Design and Development



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

The schema used : Snowflake Schema

The Snowflake Schema has been utilized in the dimensional modeling to reduce redundancy through normalization. As shown, the dimension tables have been normalized, including the DimAnimal and DimStrayMap tables.

It was assumed that the location details, such as the StrayMap and Animal dimensions, would provide greater benefits in categorizing and analyzing the data in various ways. Therefore, the DimBreed and DimDate tables contain hierarchical attributes that describe the respective characteristics.

- Year > >Quater > Month >Day of month for DimDate
- Breed > Primary Breed > secondary Breed > isMixed Type for DimBreed

- **Dimension and Fact Tables**

  Six dimension tables and one fact table was created:

  1. DimStrayMap - The stray map dimension table contains stray map details, with StrayMapSK as the surrogate key.
  2. DimColor - Contains the color details of the animals, with ColorSK as the surrogate key.
  3. DimBreed - Contains breed details of the animals, with BreedSK as the surrogate key.
  4. DimAnimal – The animal dimension table contains animal details, with AnimalSK as the surrogate key.DimStrayMap,DimColor,DimBreed liked using foreign keys.
  5. DimSexType - A categorical value describing the sex of the animal. SexTypeSK (Surrogate Key)
  6. DimDate - This is a common dimension. DateKey is the surrogate key. An SQL script was used to generate the date dimension based on the IntakeDateTime and OutcomeDateTime fields
  7. FactAnimalEvent – Contains all the transactional data related to animal intake and outcome events. It references the dimension tables via foreign keys. The fact table stores the facts and metrics such as StayDurationDays, along with foreign keys linking it to DimAnimal, DimSex_Type, and DimDate.

- **Slowly changing dimensions –**

  It was assumed that certain attributes related to DimStrayMap could change over time, while others would remain historical. For instance, an animal's Age or whether it is At AAC (at the animal center) could change due to different factors (e.g., animal's age changes over time or its status at the center changes). Additionally, the Image_Link could be updated if the animal receives a new image.

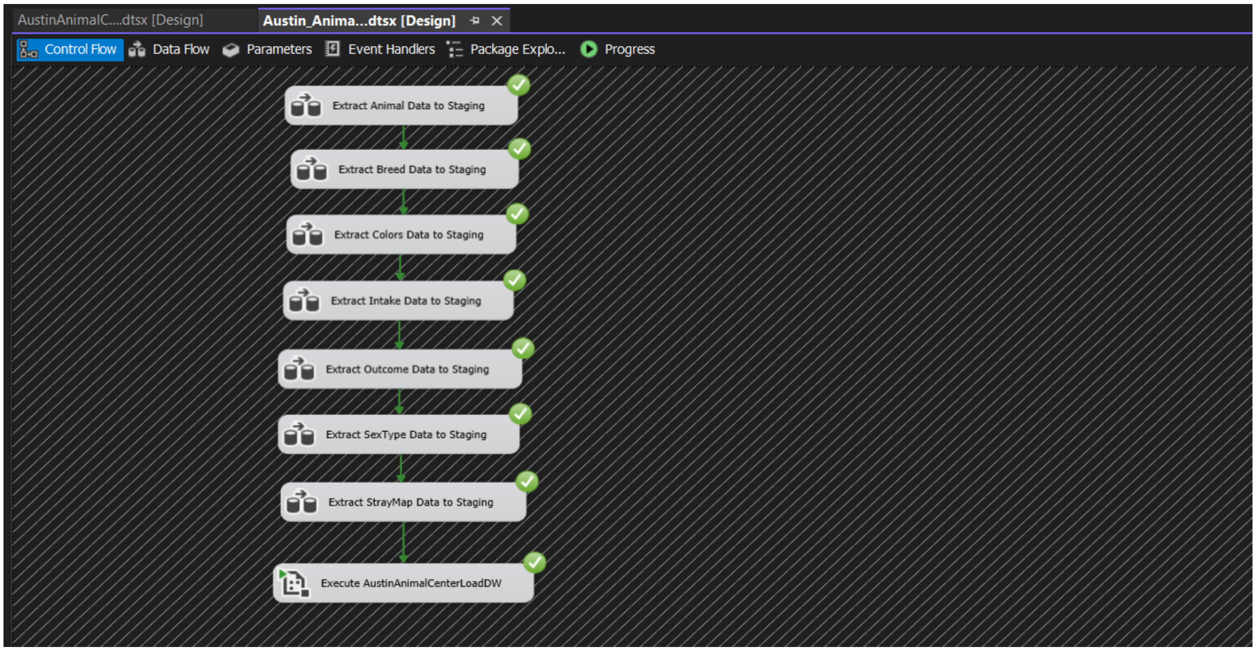
  Changing Attribute: Age, At_AAC and Image_Link

  Historical Attribute: Intake_Date

5 ETL Development

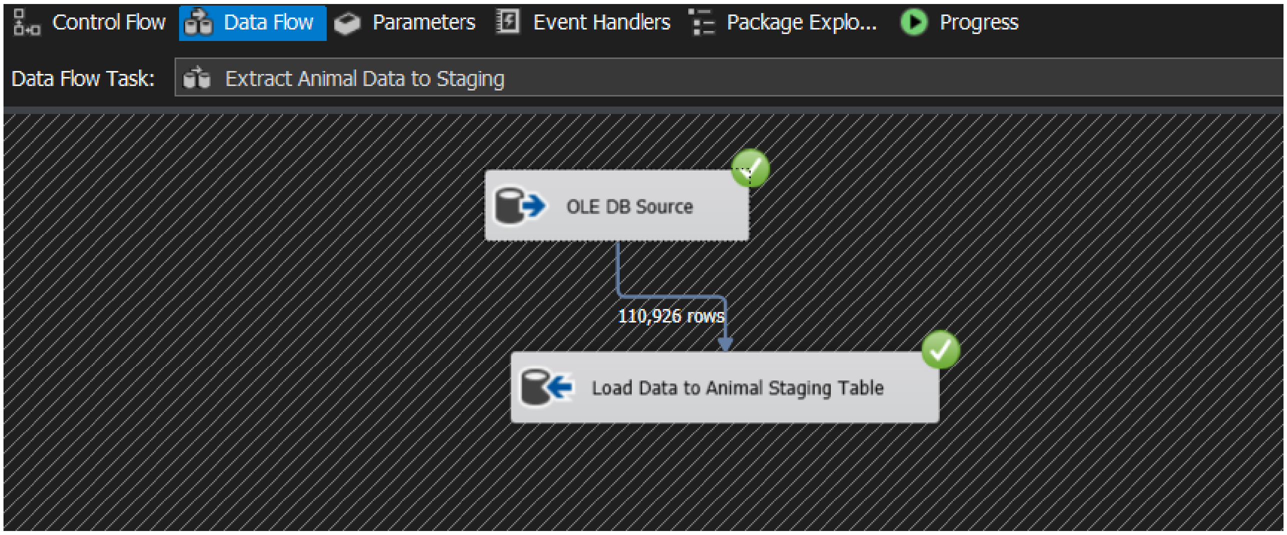
5.1 Extract Data from Source to Staging

Execution order

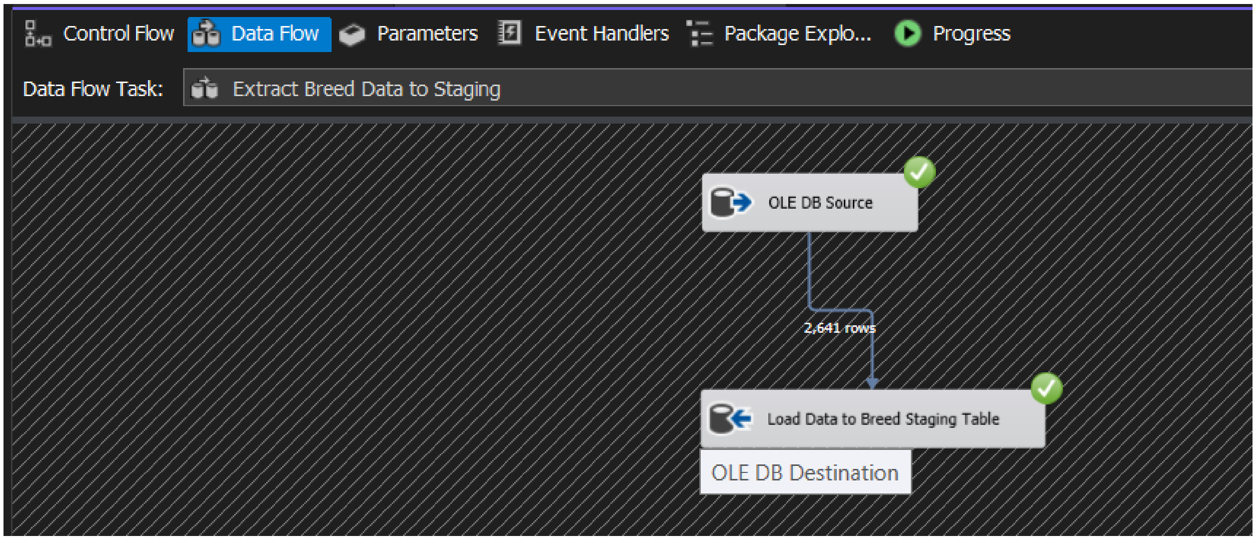

AustinAnimalC....dtsx [Design]
Austin_Anima...dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Extract Animal Data to Staging
Extract Breed Data to Staging
Extract Colors Data to Staging
Extract Intake Data to Staging
Extract Outcome Data to Staging
Extract SexType Data to Staging
Extract StrayMap Data to Staging
Execute AustinAnimalCenterLoadDW

. Individual extractions into the staging database happens as below images:

- Animal data are extracted from a source DB

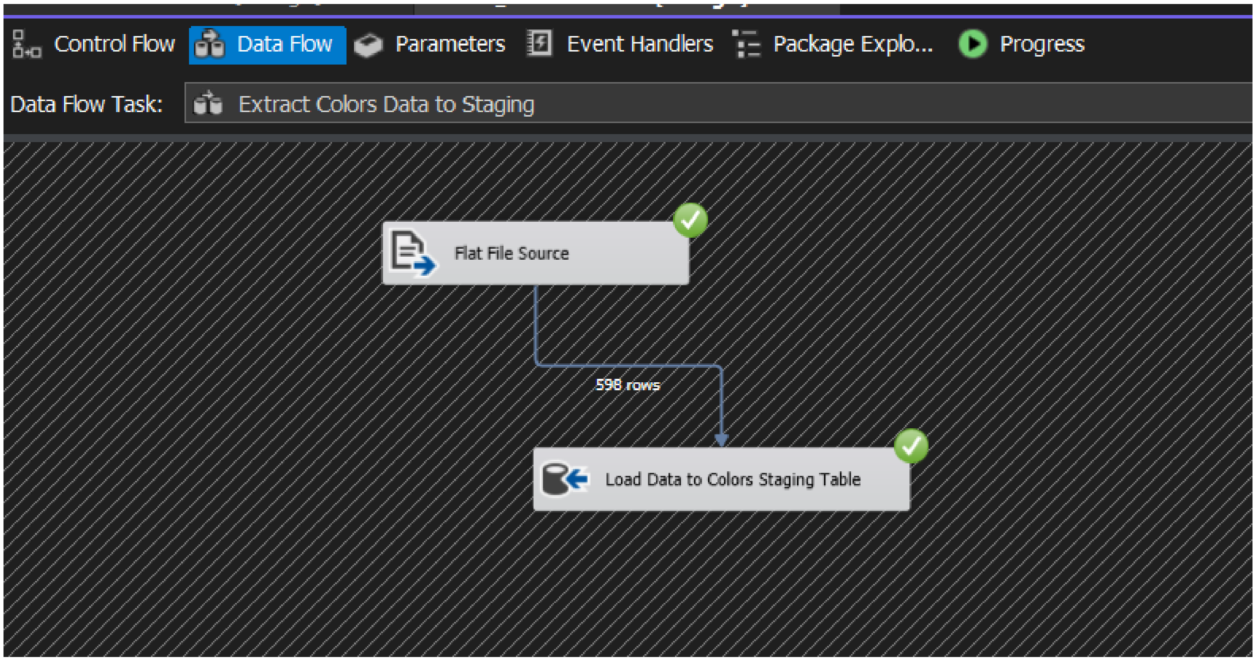


- Breed Data extract from Source DB.



Data Flow Task: Extract Breed Data to Staging

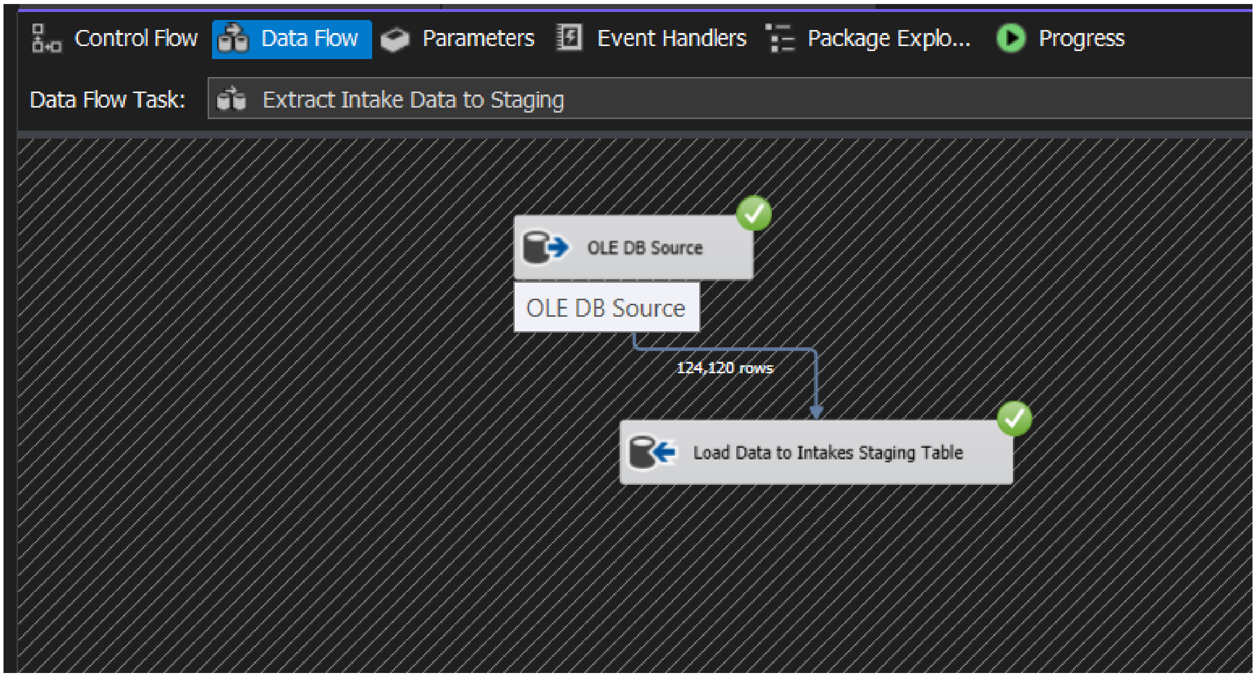
- Colors data extract from flat file source(.csv).



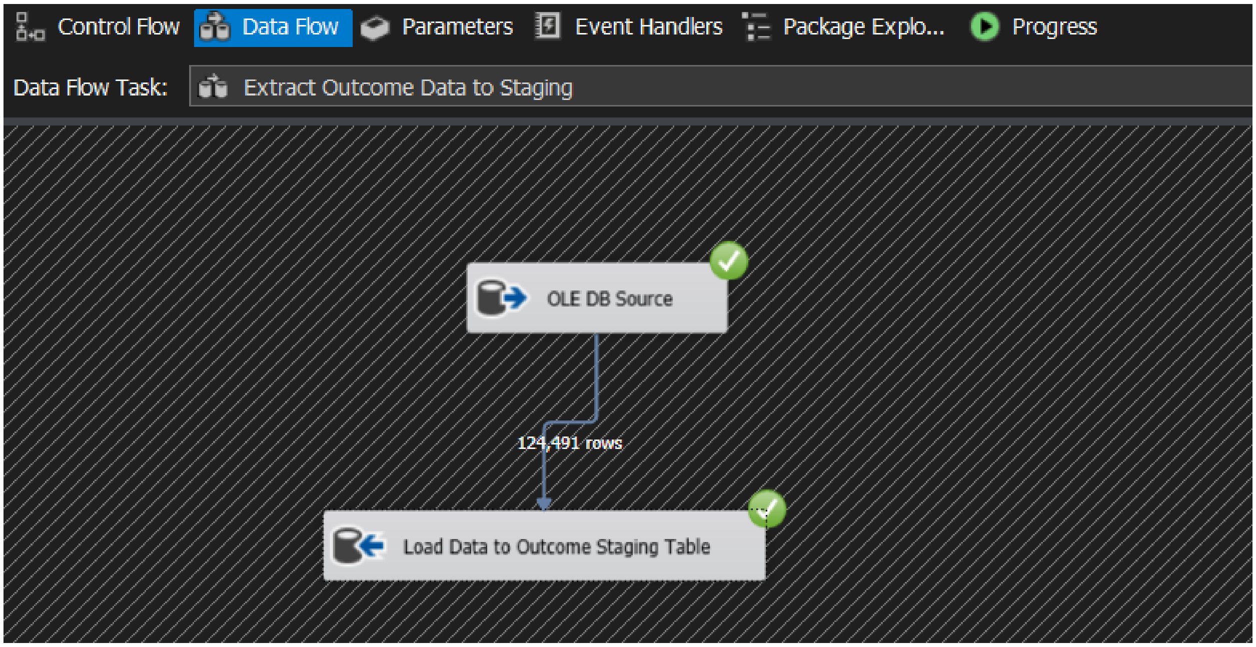
Data Flow Task: Extract Colors Data to Staging

-

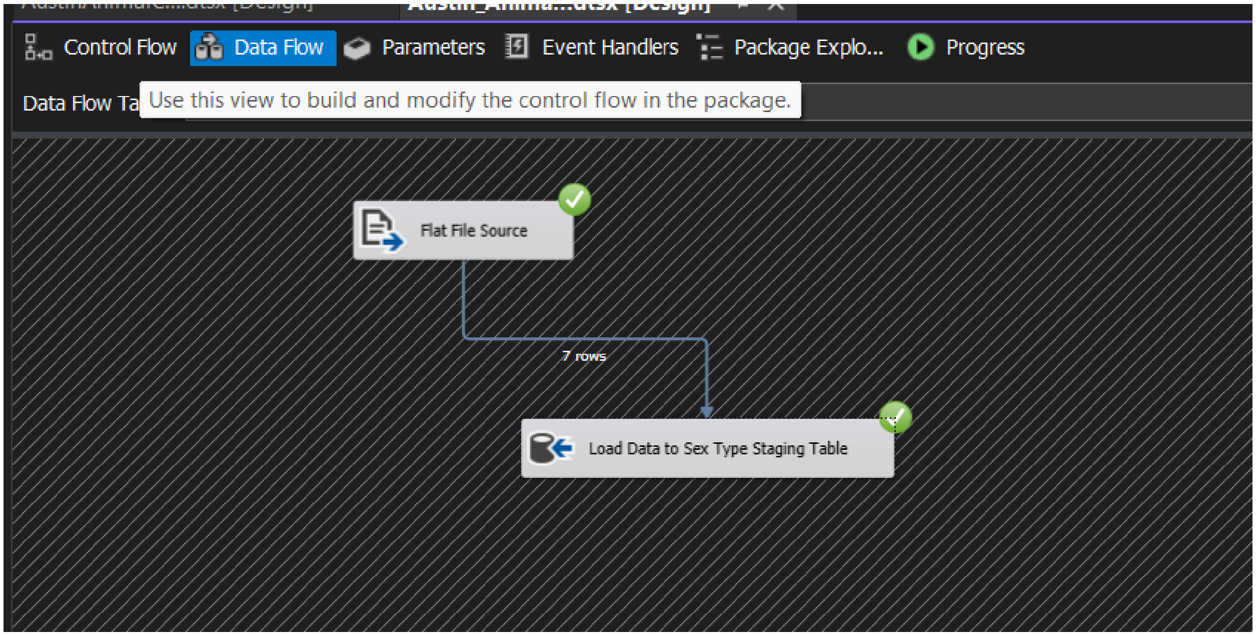
- Intakes data extract from source DB



- Outcome data extract from source DB .

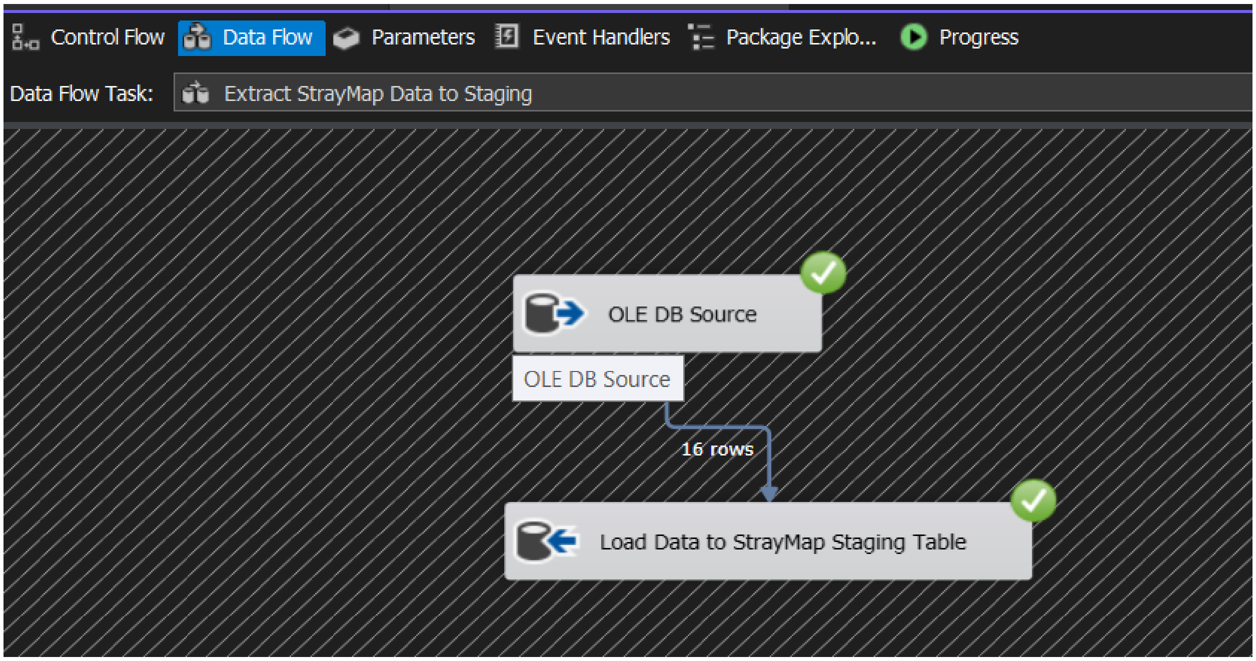


- Sex Type data extract from flat file source(.txt)



Use this view to build and modify the control flow in the package.

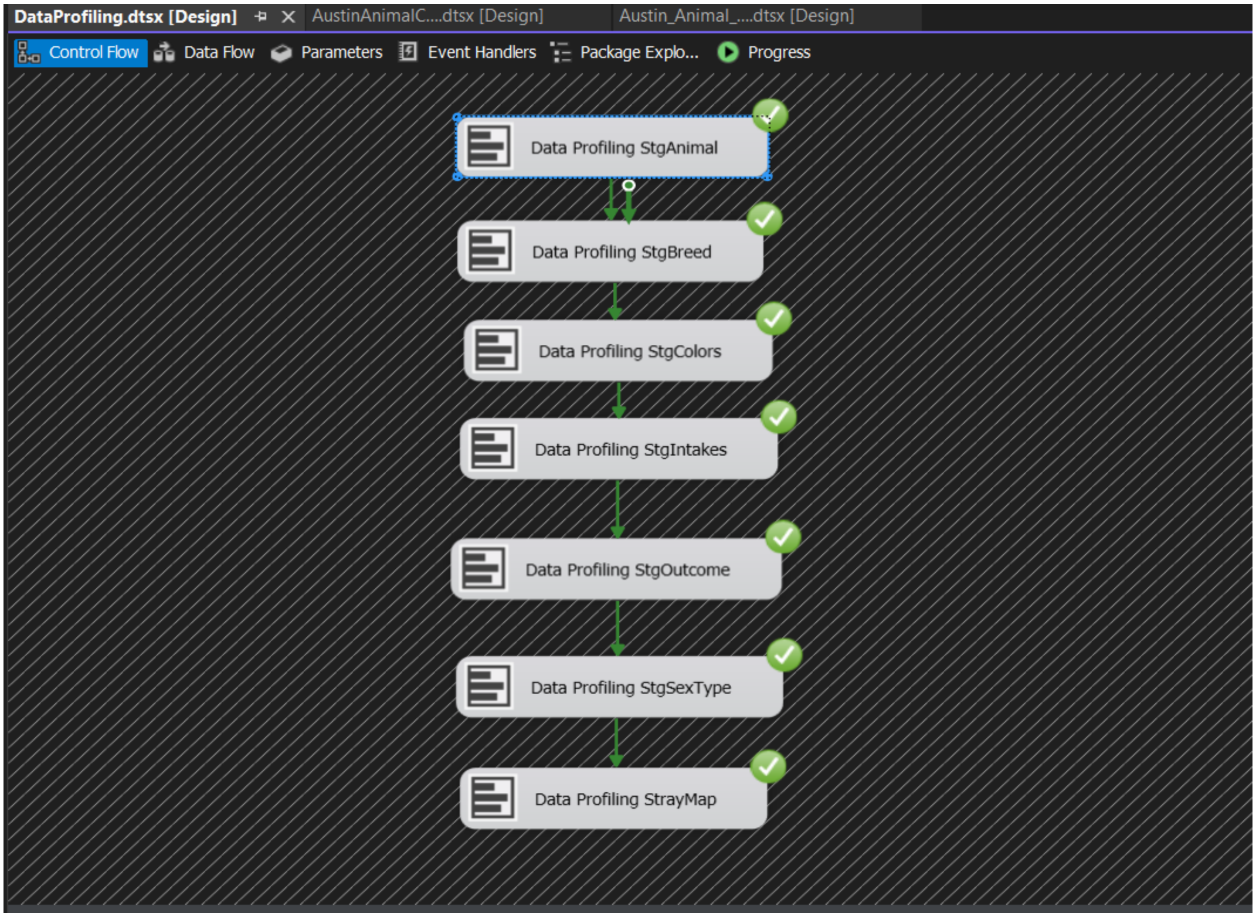
- Stray Map data extract from source DB



Data Flow Task: Extract StrayMap Data to Staging

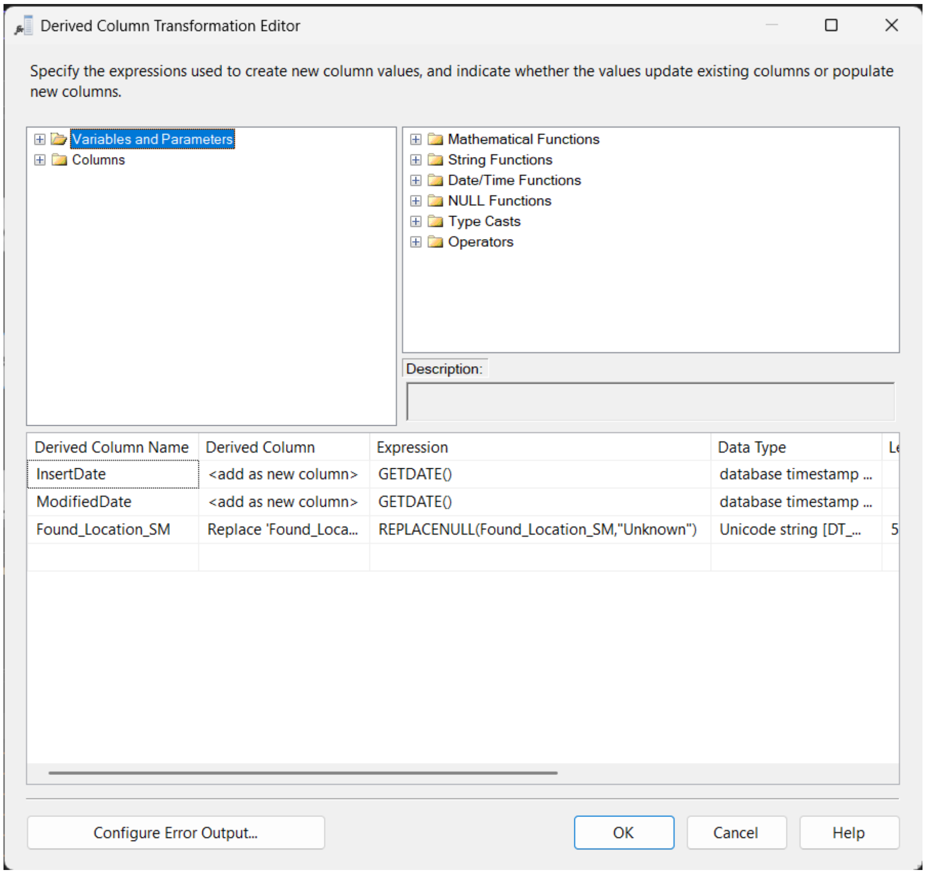
-

5.1.1 Data Profiling Package


DataProfiling.dtsx [Design]
AustinAnimalC....dtsx [Design]
Austin_Animal_....dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Profiling StgAnimal
Data Profiling StgBreed
Data Profiling StgColors
Data Profiling StgIntakes
Data Profiling StgOutcome
Data Profiling StgSexType
Data Profiling StrayMap

5.2 Transforming the Staged Data Before They Loaded in to Data Warehouse

- In the Stray Map table, the NULL values in the 'Found Location Data' field were identified and replaced. Insertions of the modified and insert date are also assigned to obtain the system dates during insertion and modification.

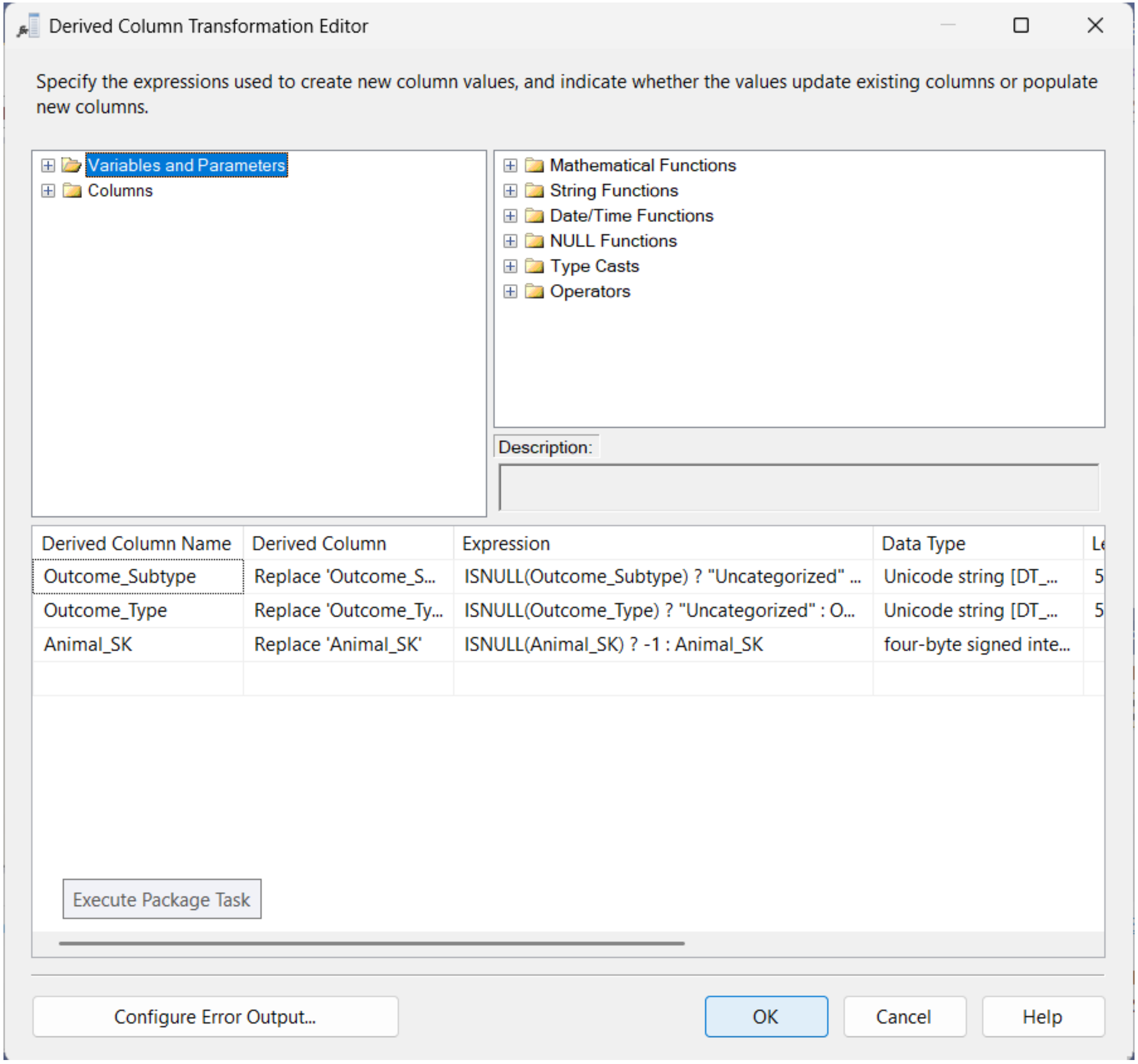


Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

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 ...	
Found_Location_SM	Replace 'Found_Loca...	REPLACENULL(Found_Location_SM,"Unknown")	Unicode string [DT_...	5

-

- In FactAnimalEvent table, the Null values in outcome Sub type were Identified and replaced.



Derived Column Transformation Editor

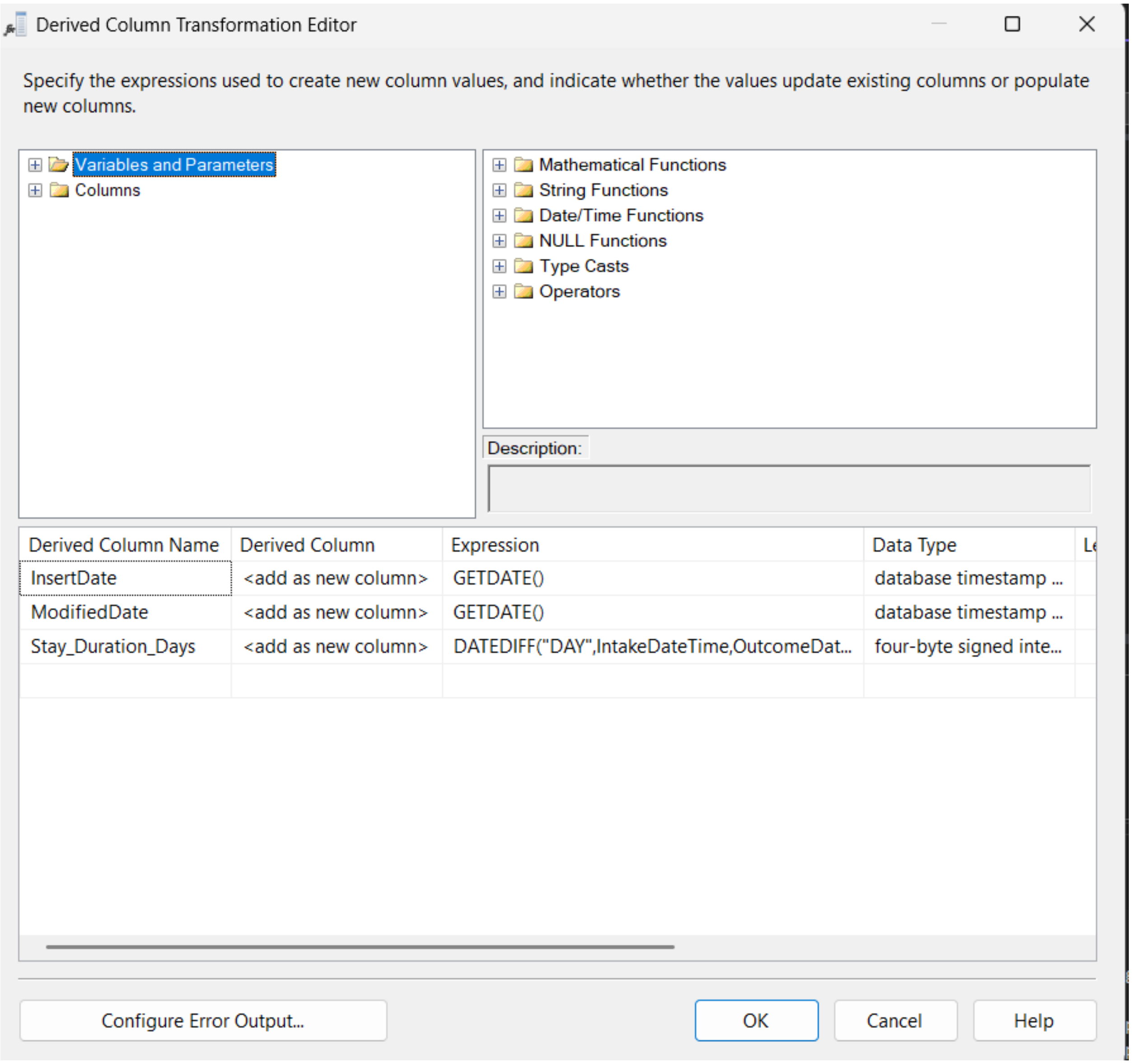
Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.

Derived Column Name	Derived Column	Expression	Data Type	Le
Outcome_Subtype	Replace 'Outcome_S...	ISNULL(Outcome_Subtype) ? "Uncategorized" ...	Unicode string [DT_...	5
Outcome_Type	Replace 'Outcome_Ty...	ISNULL(Outcome_Type) ? "Uncategorized" : O...	Unicode string [DT_...	5
Animal_SK	Replace 'Animal_SK'	ISNULL(Animal_SK) ? -1 : Animal_SK	four-byte signed inte...	

- 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.

- In FactAnimalEvent table, derived columns have been used to add the insertion and modified date columns.
  The duration of an animal's stay at the center was calculated by finding the difference between the intake date and the outcome date



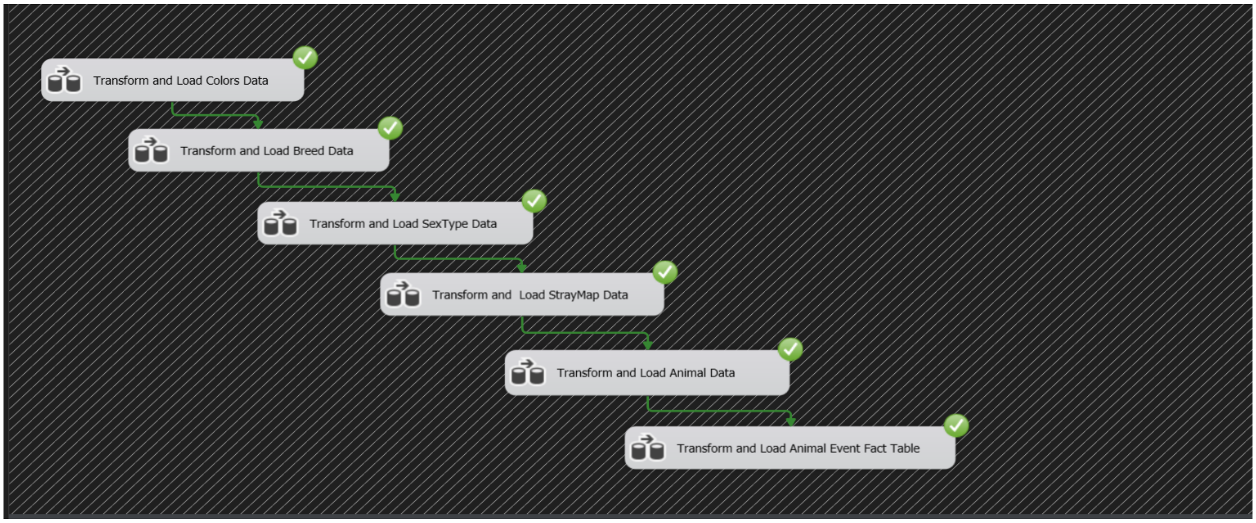
Derived Column Transformation Editor
Specify the expressions used to create new column values, and indicate whether the values update existing columns or populate new columns.
Variables and Parameters
Columns
Mathematical Functions
String Functions
Date/Time Functions
NULL Functions
Type Casts
Operators
Description:
Configure Error Output...
OK
Cancel
Help

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 ...	
Stay_Duration_Days	<add as new column>	DATEDIFF("DAY",IntakeDateTime,OutcomeDat...	four-byte signed inte...	

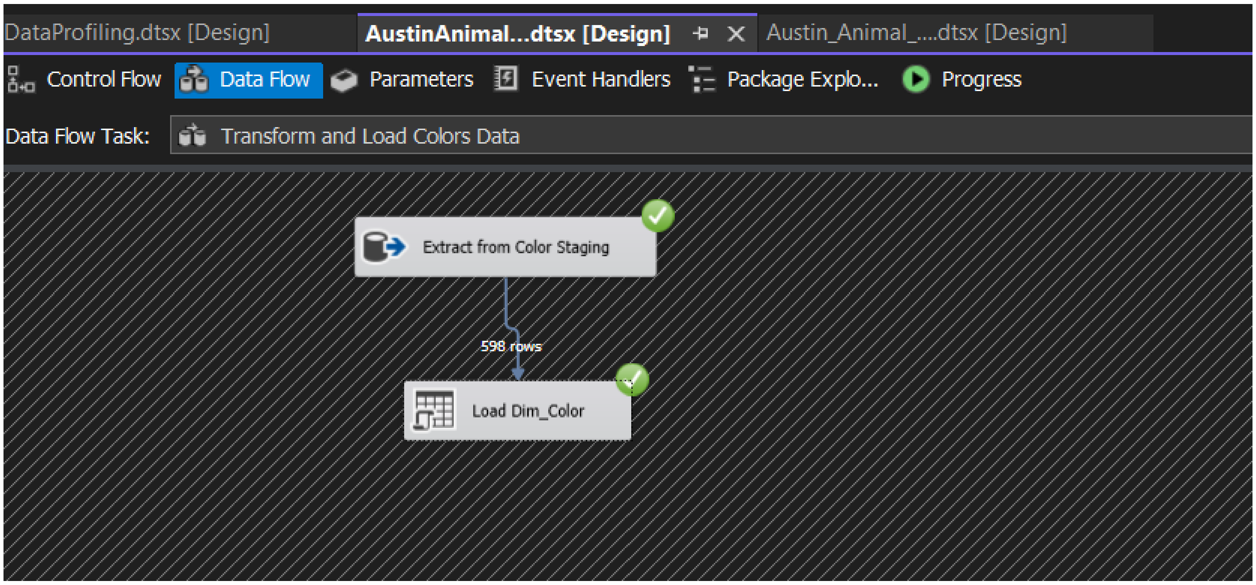
-

5.3 Loading the Transformed Data into the Data Warehouse

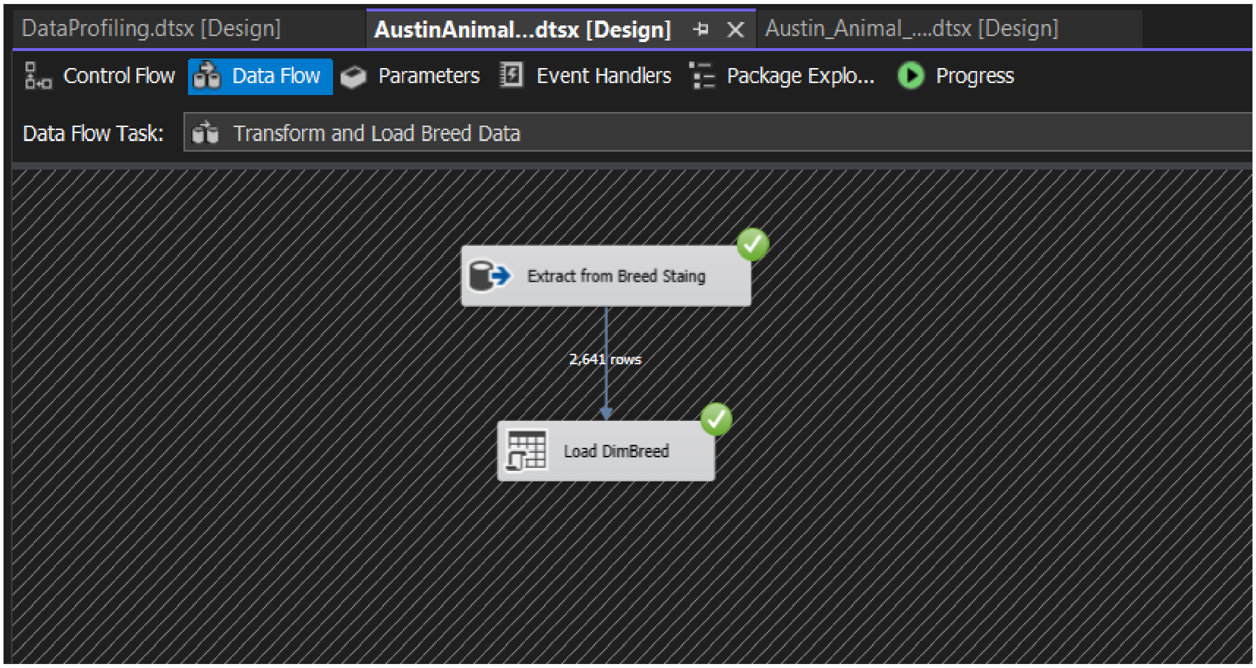
- Order of Execution


Transform and Load Colors Data
Transform and Load Breed Data
Transform and Load SexType Data
Transform and Load StrayMap Data
Transform and Load Animal Data
Transform and Load Animal Event Fact Table

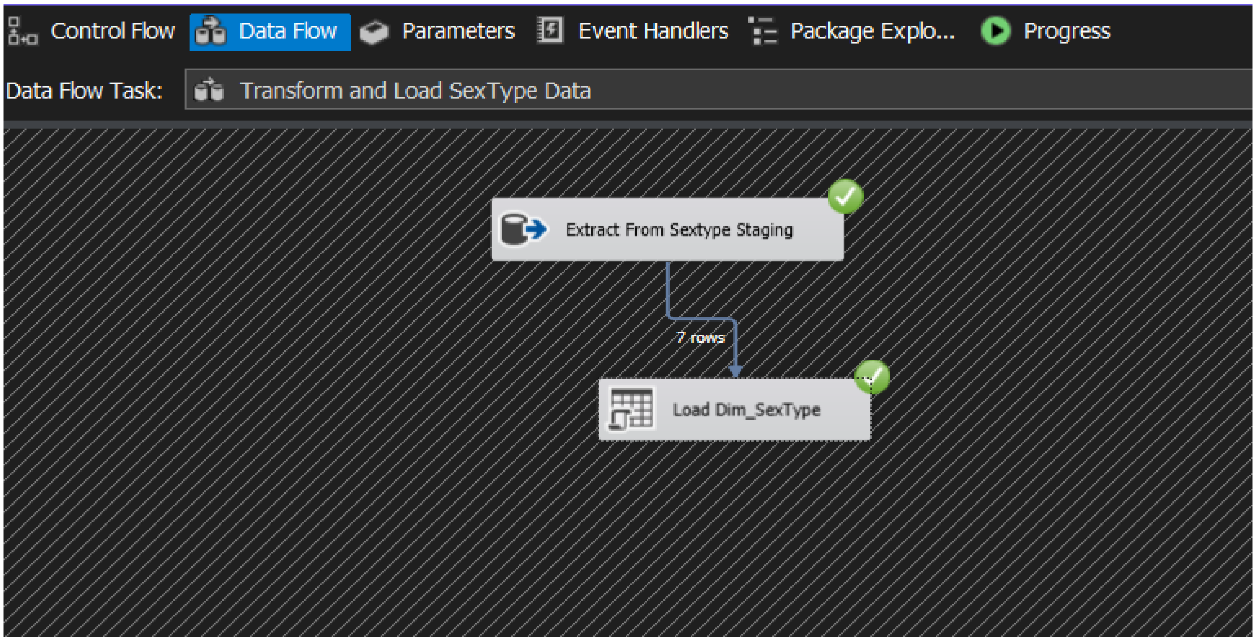
- Loading color staging data in to data warehouse Dim Color table.


DataProfiling.dtsx [Design]
AustinAnimal...dtsx [Design]
Austin_Animal_....dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load Colors Data
Extract from Color Staging
598 rows
Load Dim_Color

- Loading breed staging data in to data warehouse Dim Breed table.

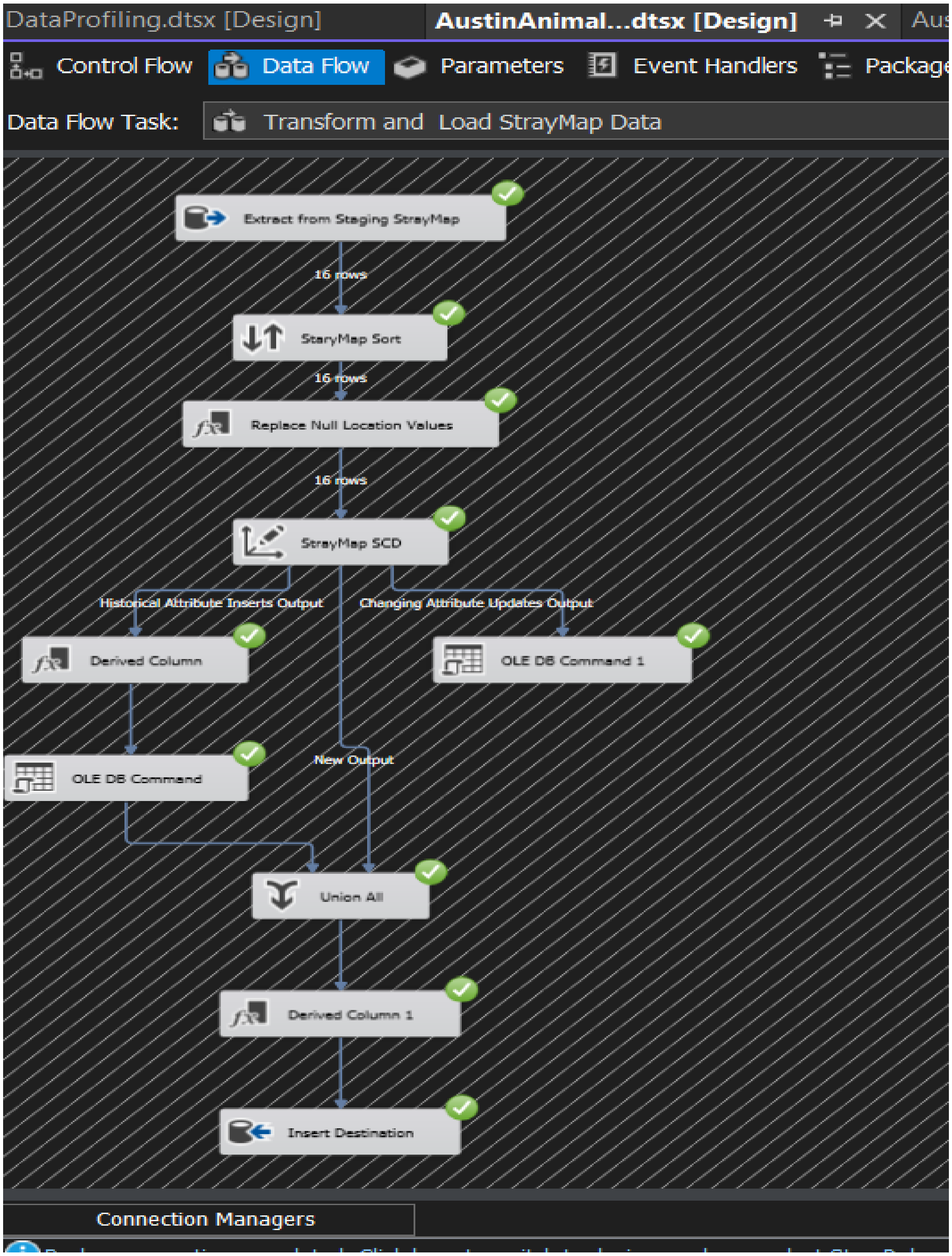

DataProfiling.dtsx [Design]
AustinAnimal...dtsx [Design]
Austin_Animal_....dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load Breed Data
Extract from Breed Staing
2,641 rows
Load DimBreed

- Loading sex type staging data in to data warehouse Dim SexType table.



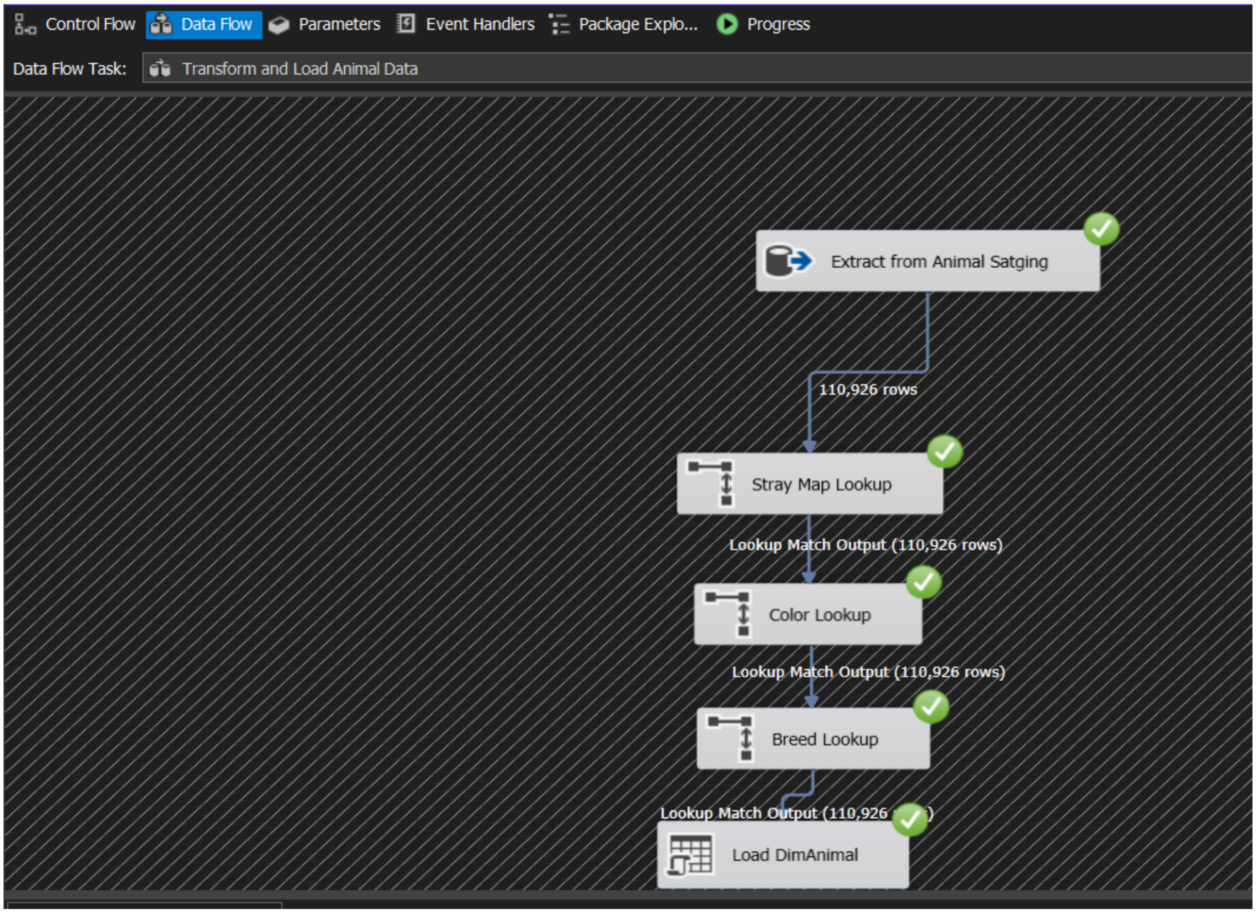
-

- Loading the transformed Stray Map Staging data as a slowly changing dimension into the data warehouse DimStrayMap.



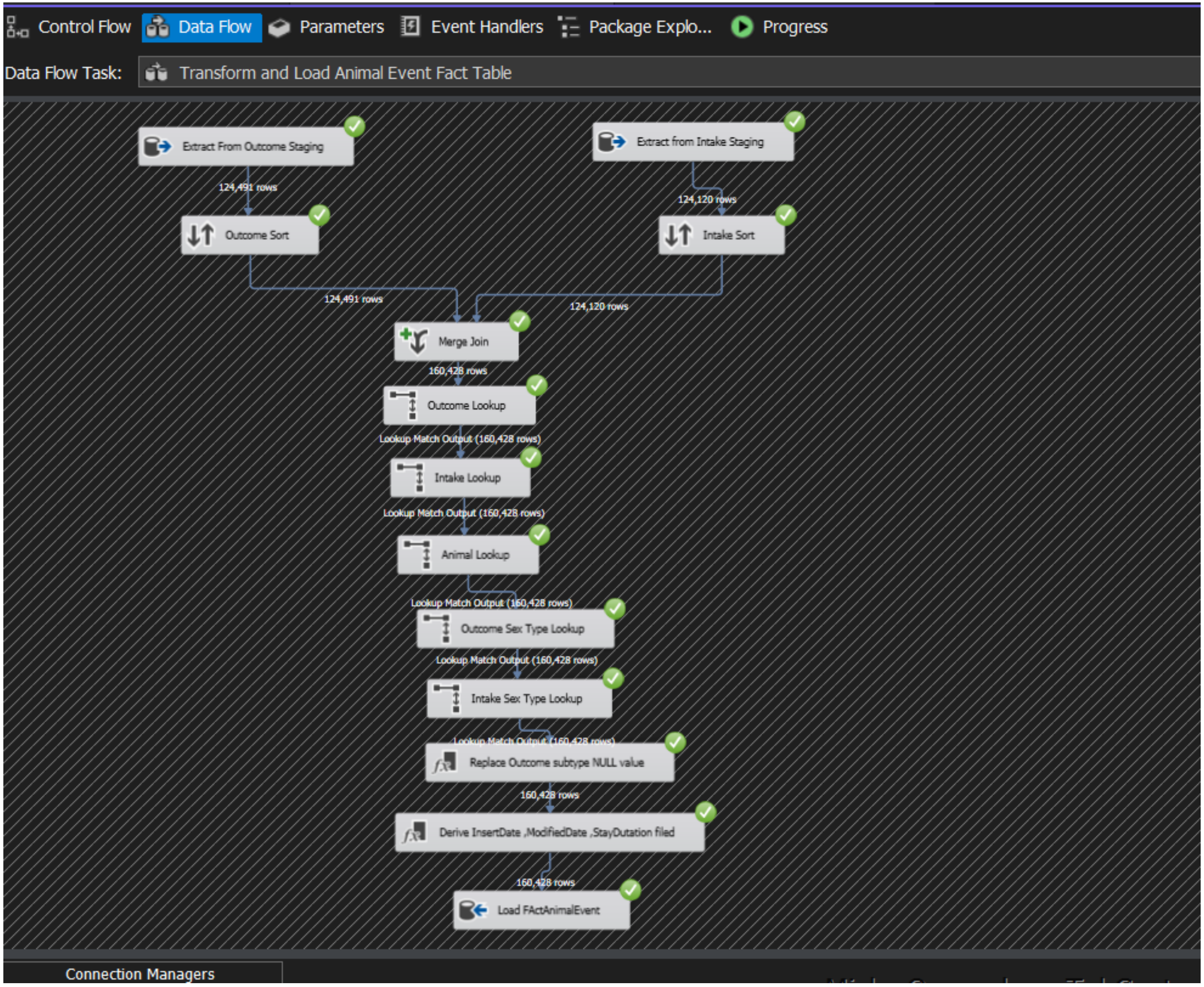
-

- Loading animal staging data in to data warehouse Dim Animal table and Lookups and merge Join have been used to create foreign key references between table.


Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load Animal Data
Extract from Animal Satging
110,926 rows
Stray Map Lookup
Lookup Match Output (110,926 rows)
Color Lookup
Lookup Match Output (110,926 rows)
Breed Lookup
Lookup Match Output (110,926)
Load DimAnimal

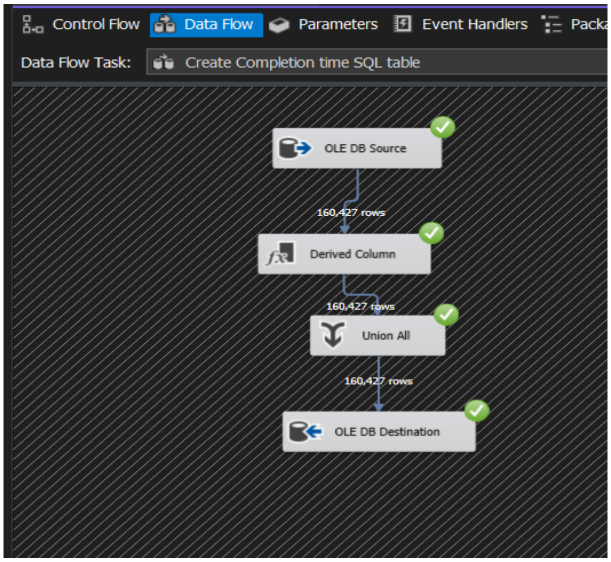
-

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

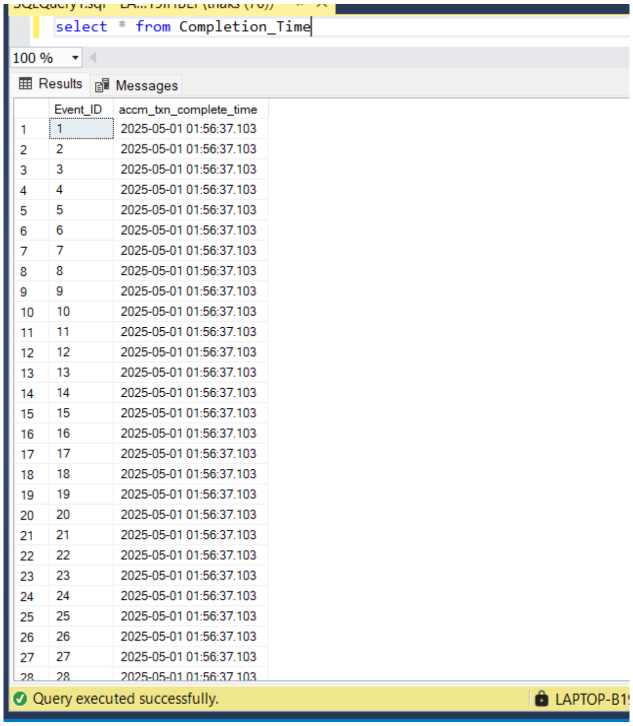

Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load Animal Event Fact Table
Extract From Outcome Staging
124,491 rows
Outcome Sort
124,491 rows
Extract from Intake Staging
124,120 rows
Intake Sort
124,120 rows
Merge Join
160,428 rows
Outcome Lookup
Lookup Match Output (160,428 rows)
Intake Lookup
Lookup Match Output (160,428 rows)
Animal Lookup
Lookup Match Output (160,428 rows)
Outcome Sex Type Lookup
Lookup Match Output (160,428 rows)
Intake Sex Type Lookup
Lookup Match Output (160,428 rows)
Replace Outcome subtype NULL value
160,428 rows
Derive InsertDate ,ModifiedDate ,StayDutation filed
160,428 rows
Load FActAnimalEvent
Connection Managers

6 ETL Development – Accumulating Fact Tables

6.1 Creating Completion Time SQL Table



Data Flow Task: Create Completion time SQL table

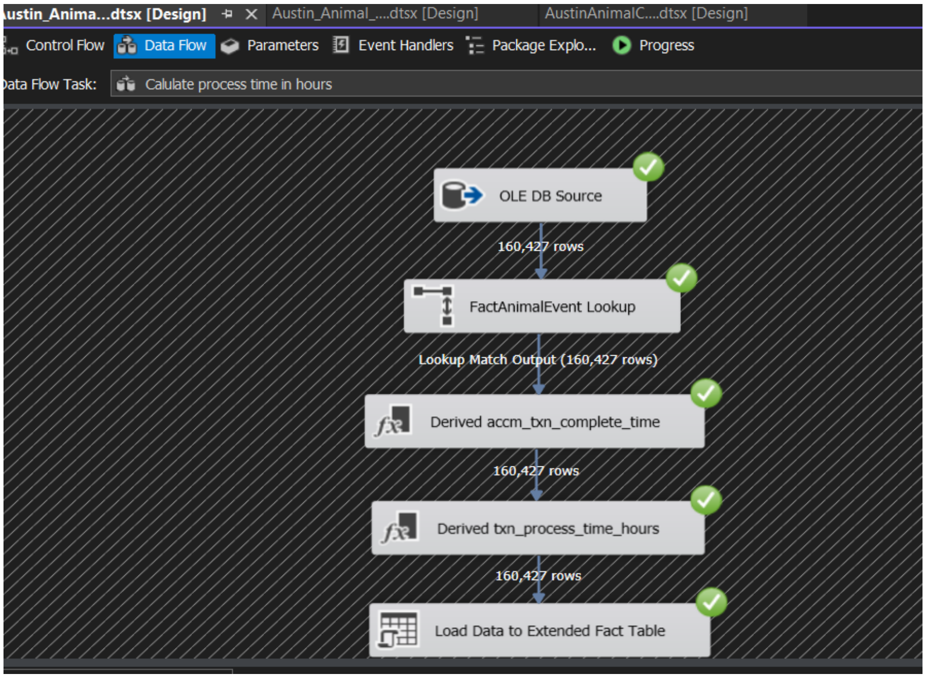


```sql
select * from Completion_Time
```

	Event_ID	accm_txn_complete_time
1	1	2025-05-01 01:56:37.103
2	2	2025-05-01 01:56:37.103
3	3	2025-05-01 01:56:37.103
4	4	2025-05-01 01:56:37.103
5	5	2025-05-01 01:56:37.103
6	6	2025-05-01 01:56:37.103
7	7	2025-05-01 01:56:37.103
8	8	2025-05-01 01:56:37.103
9	9	2025-05-01 01:56:37.103
10	10	2025-05-01 01:56:37.103
11	11	2025-05-01 01:56:37.103
12	12	2025-05-01 01:56:37.103
13	13	2025-05-01 01:56:37.103
14	14	2025-05-01 01:56:37.103
15	15	2025-05-01 01:56:37.103
16	16	2025-05-01 01:56:37.103
17	17	2025-05-01 01:56:37.103
18	18	2025-05-01 01:56:37.103
19	19	2025-05-01 01:56:37.103
20	20	2025-05-01 01:56:37.103
21	21	2025-05-01 01:56:37.103
22	22	2025-05-01 01:56:37.103
23	23	2025-05-01 01:56:37.103
24	24	2025-05-01 01:56:37.103
25	25	2025-05-01 01:56:37.103
26	26	2025-05-01 01:56:37.103
27	27	2025-05-01 01:56:37.103
28	28	2025-05-01 01:56:37.103

Query executed successfully.

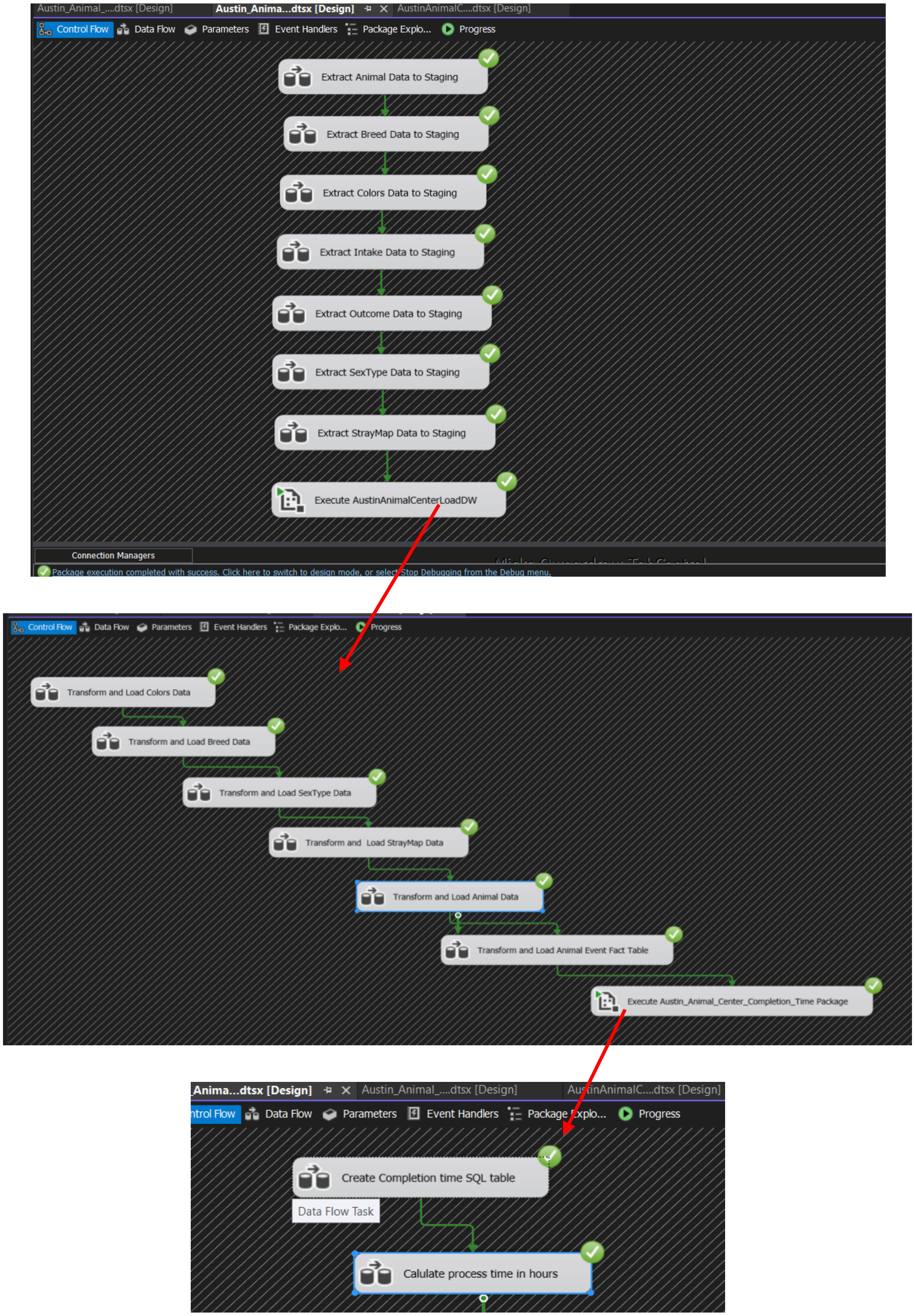
6.2 Update Fact Table with Calculated Process Time


ustin_Anima...dtsx [Design]
Austin_Animal_....dtsx [Design]
AustinAnimalC....dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
OLE DB Source
160,427 rows
FactAnimalEvent Lookup
Lookup Match Output (160,427 rows)
Derived accm_txn_complete_time
160,427 rows
Derived txn_process_time_hours
160,427 rows
Load Data to Extended Fact Table

ata Flow Task: Calulate process time in hours

7 Overall Execution Flow of the Total Solution

The proper flow of execution is as follows: first, execute the data staging; then, load the staged data into the data warehouse; and finally, update the fact table with the process time.


Austin_Animal_....dtsx [Design]
Austin_Anima...dtsx [Design]
AustinAnimalC....dtsx [Design]
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Extract Animal Data to Staging
Extract Breed Data to Staging
Extract Colors Data to Staging
Extract Intake Data to Staging
Extract Outcome Data to Staging
Extract SexType Data to Staging
Extract StrayMap Data to Staging
Execute AustinAnimalCenterLoadDW
Connection Managers
Transform and Load Colors Data
Transform and Load Breed Data
Transform and Load SexType Data
Transform and Load StrayMap Data
Transform and Load Animal Data
Transform and Load Animal Event Fact Table
Execute Austin_Animal_Center_Completion_Time Package
Anima...dtsx [Design]
Austin_Animal_....dtsx [Design]
AustinAnimalC....dtsx [Design]
ntrol Flow
Create Completion time SQL table
Data Flow Task
Calulate process time in hours

Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.