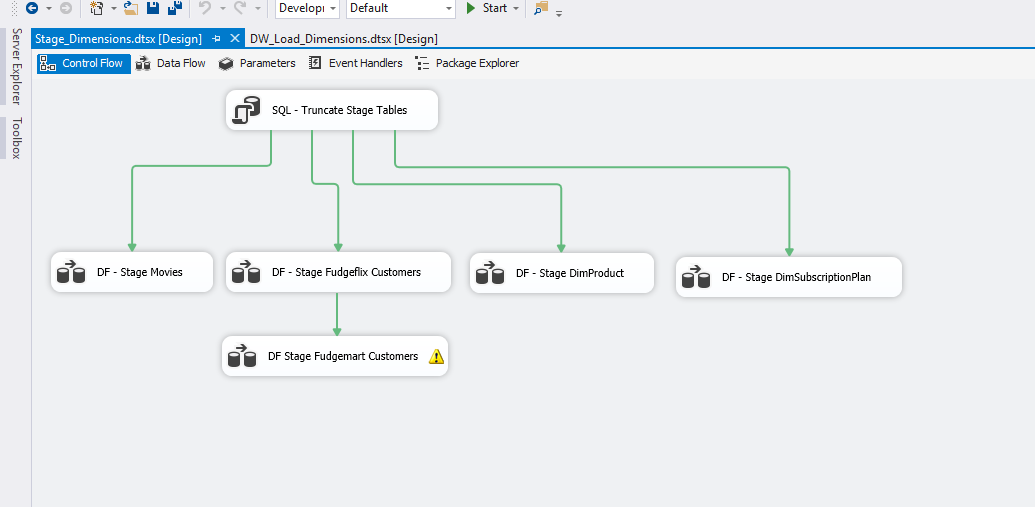
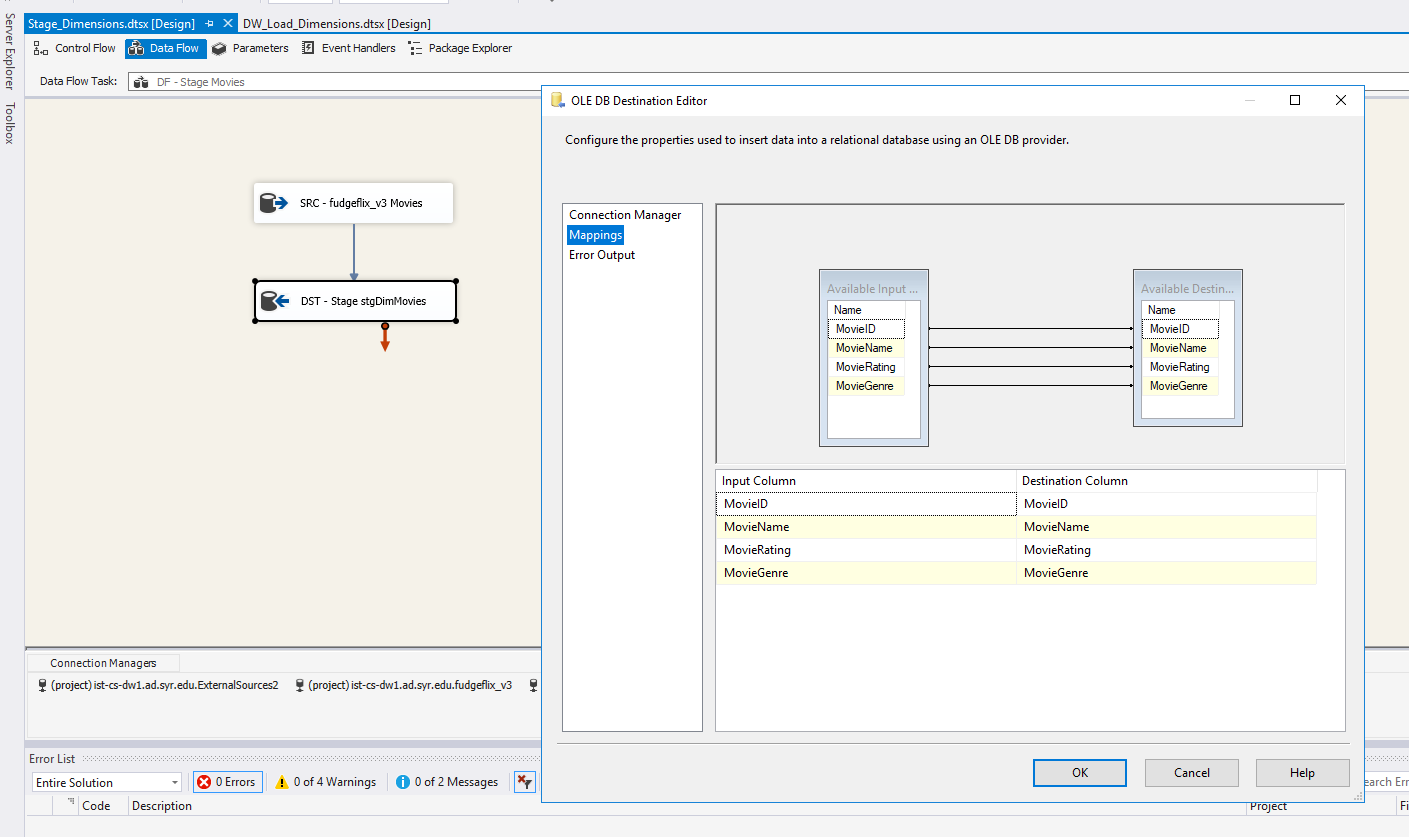
First, we focused on staging all of our dimensional tables. The overall staging package once completed can be seen below:

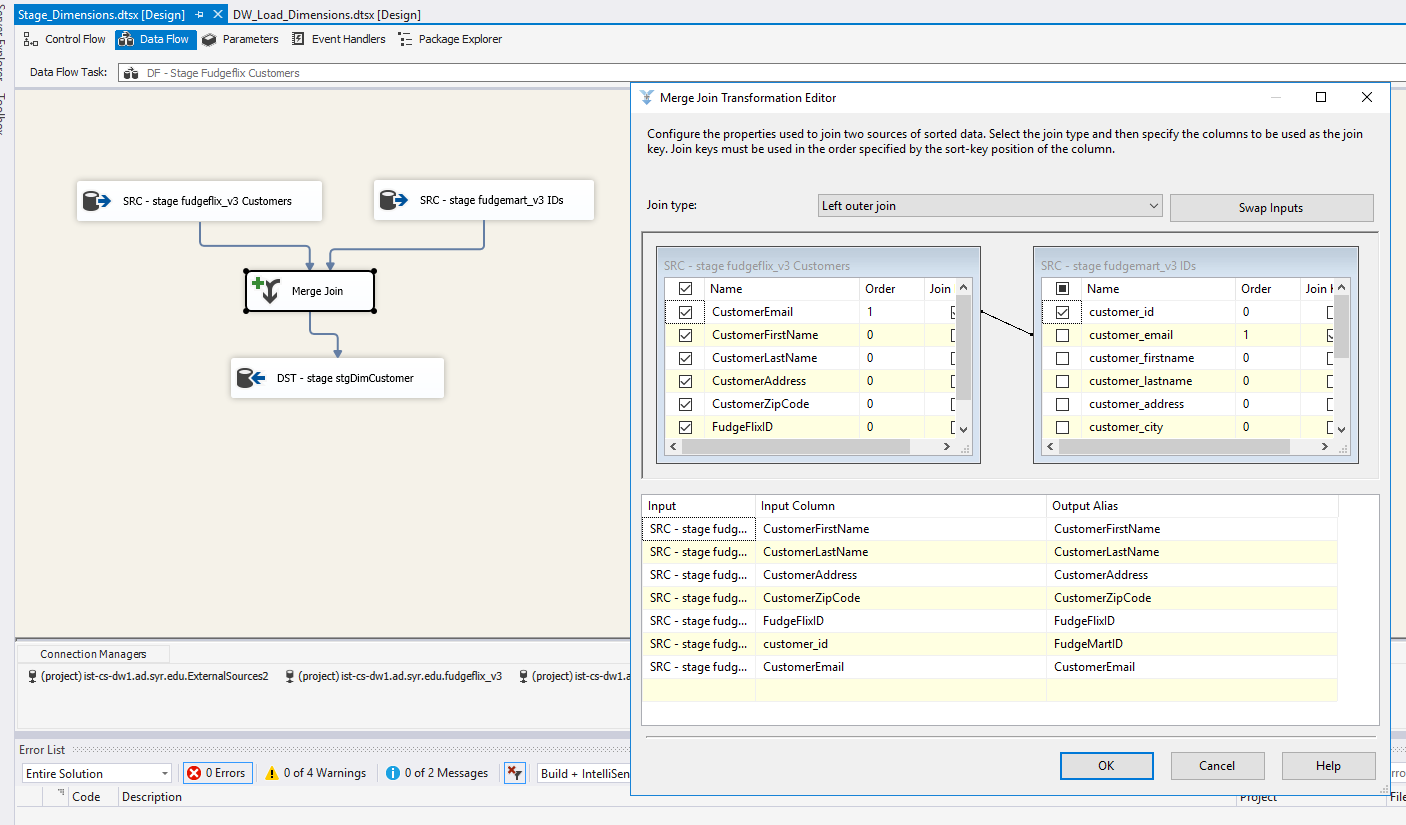


We started with staging DimMovies. The sourcing and mapping of columns was relatively straightforward as seen below:

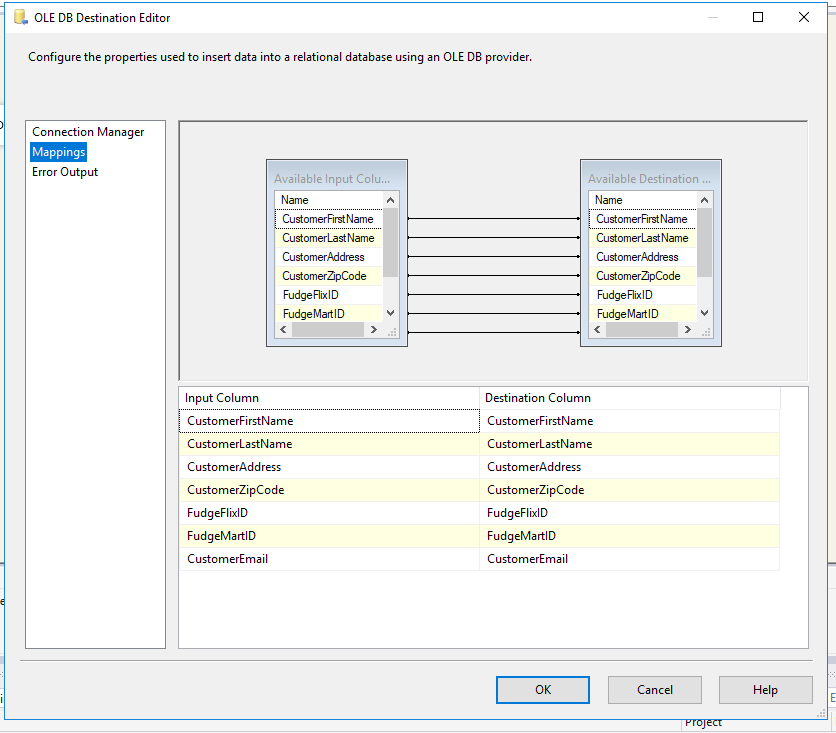


Next was DimCustomers. Because we wanted DimCustomers to contain both Fugdeflix and Fudgemart customers analysis purposes, this required some more involved staging. In the end, we made us of a merge join whereby a left outer join was used to combine records in both tables based on CustomerEmail, their common value. Once the merge join was in place, we could stage the data normally:

**DimCustomer Merge Join**

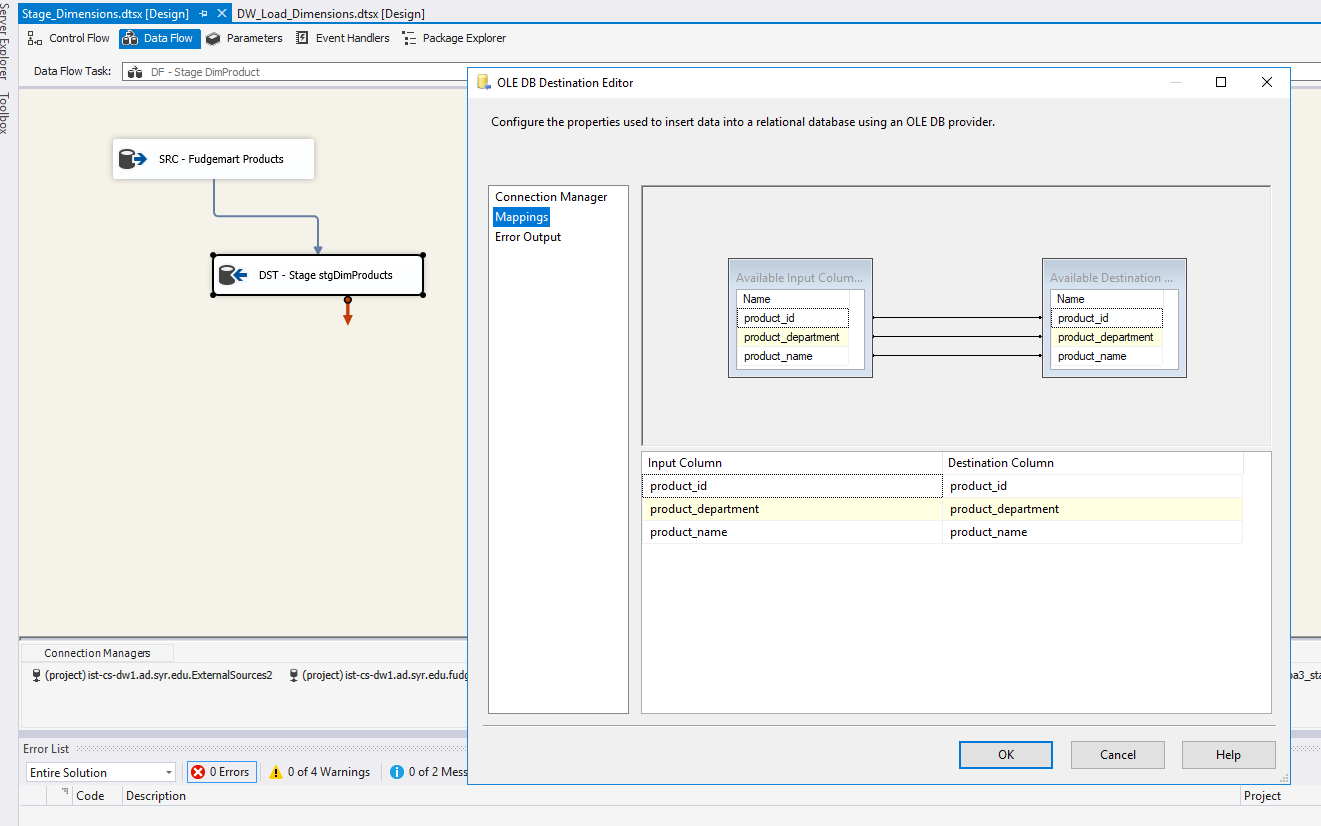


**DimCustomer Mapping**

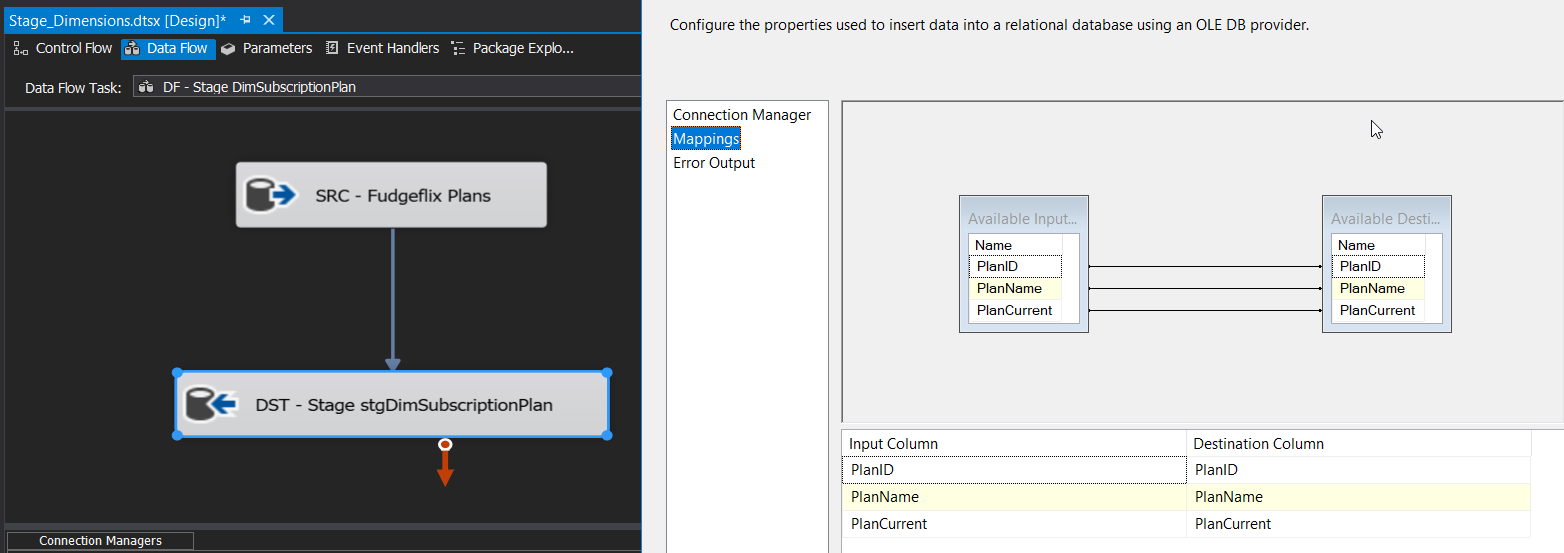


Finally, we staged DimProduct and Dim SubscriptionPlan:

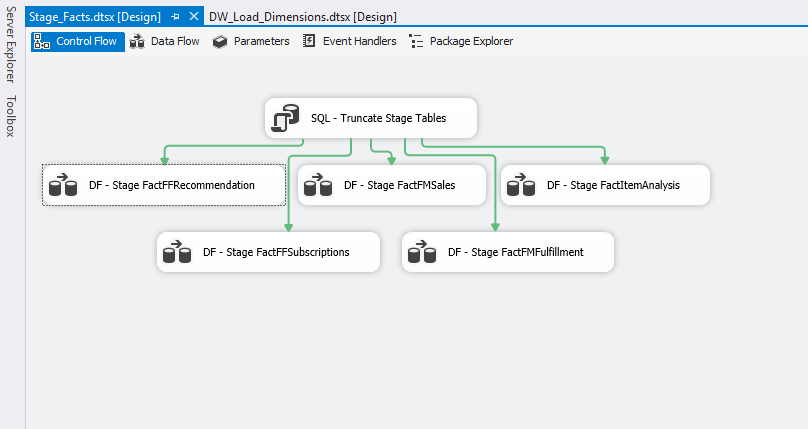
**DimProduct:**

****

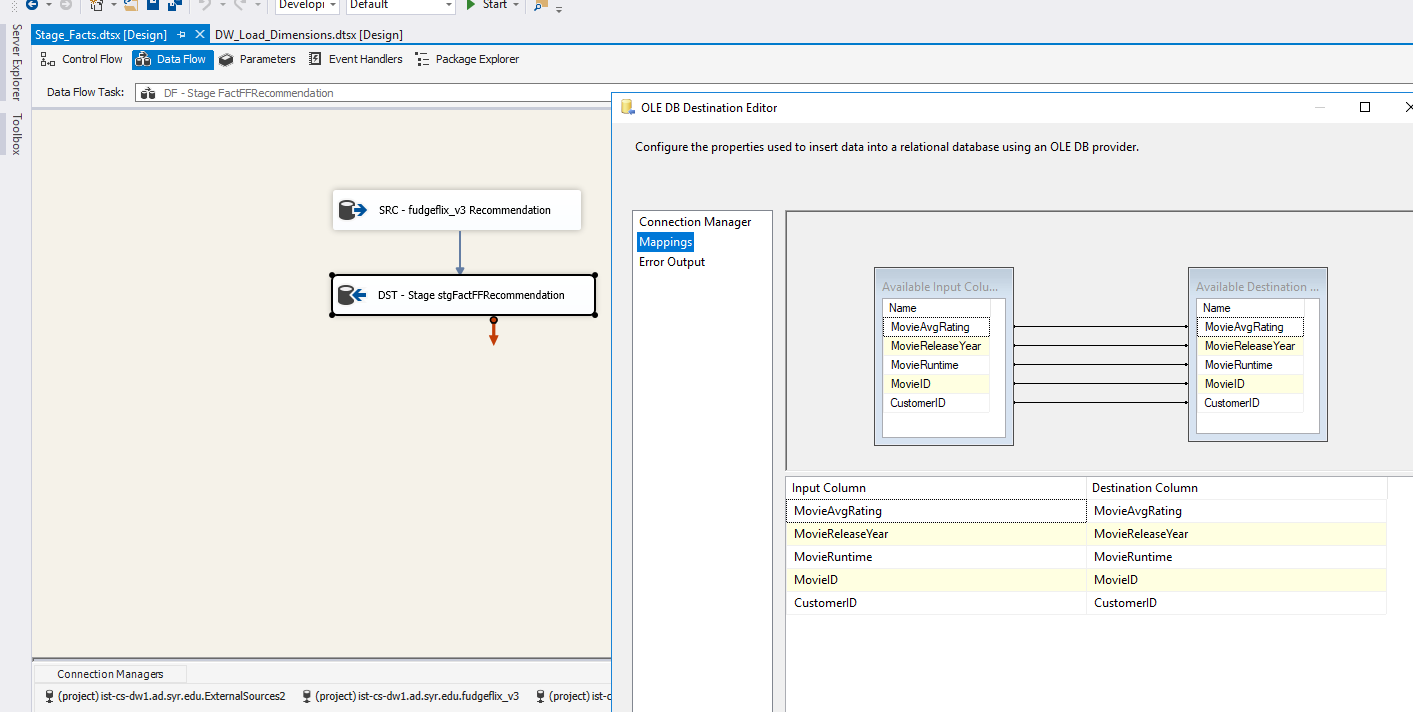
**DimSubscriptionPlan:**



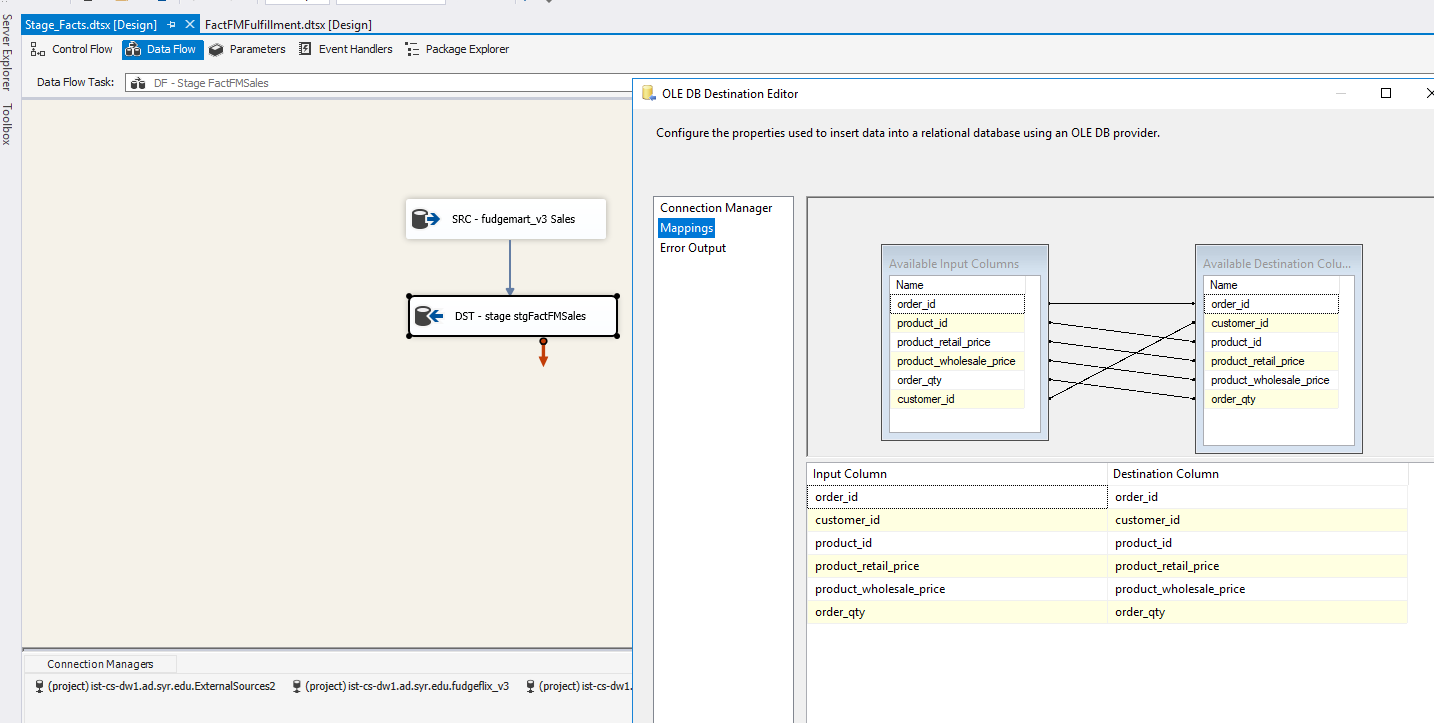
With our dimension tables done, we focused on staging our fact tables:



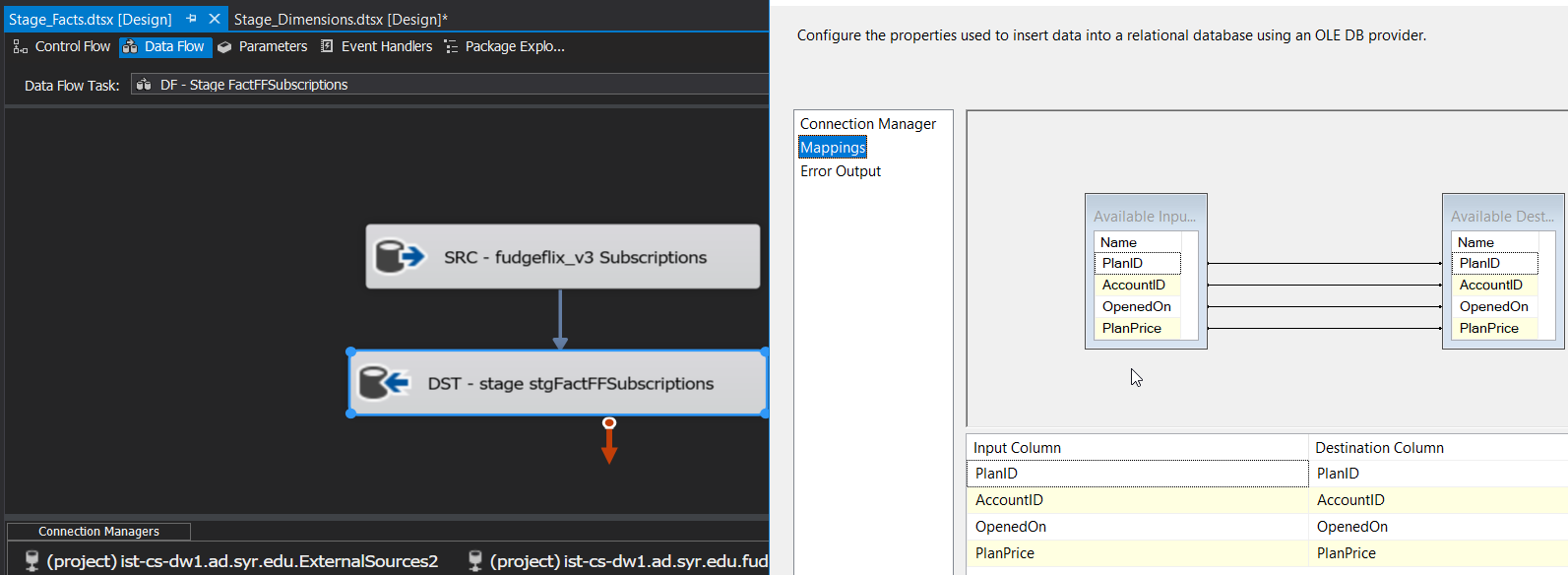
First, we started with FactFFRecommendation:



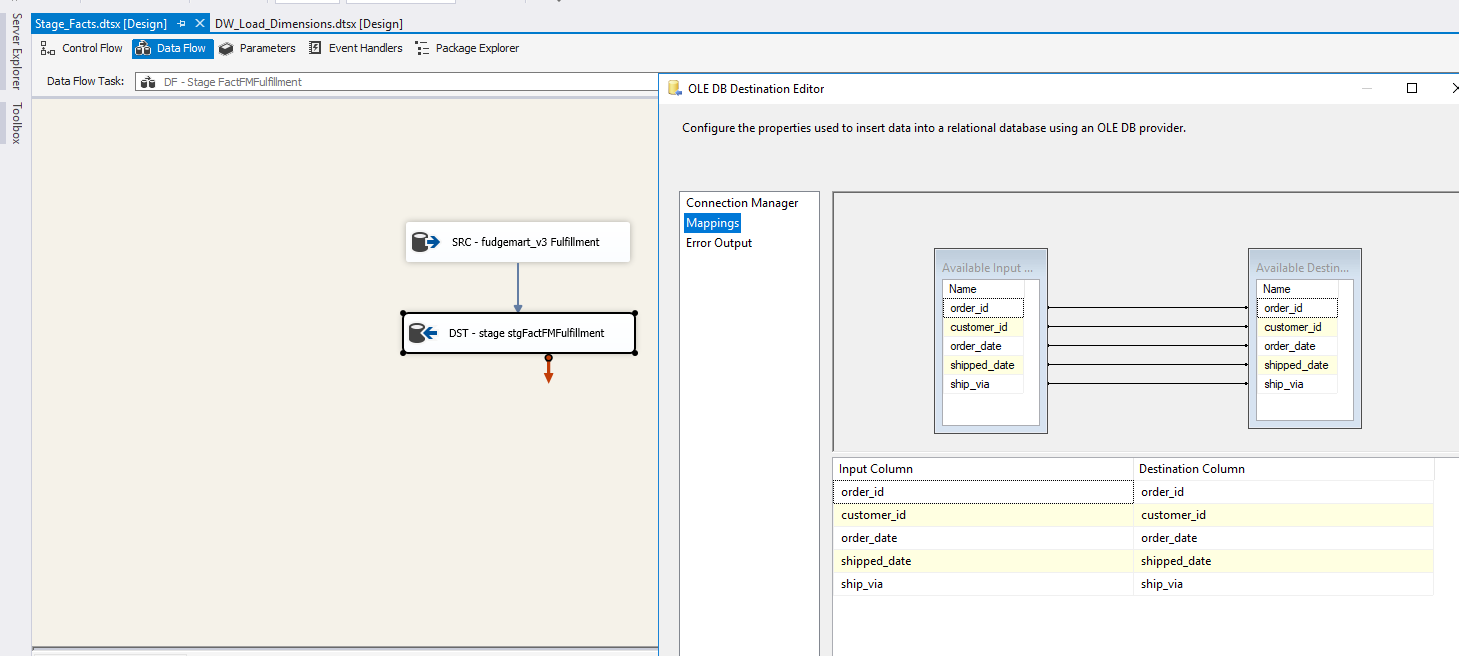
Then FactFMSales:



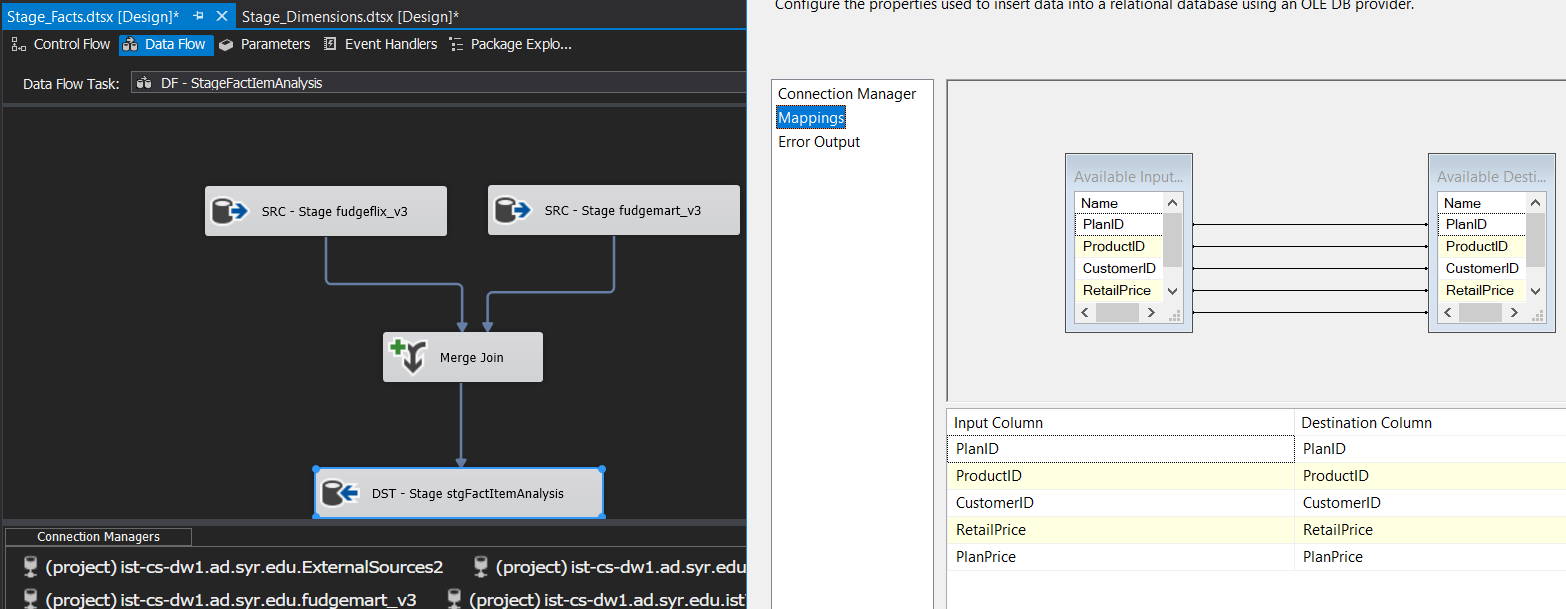
FactFFSubscriptions:



And FactFMFulfillment:

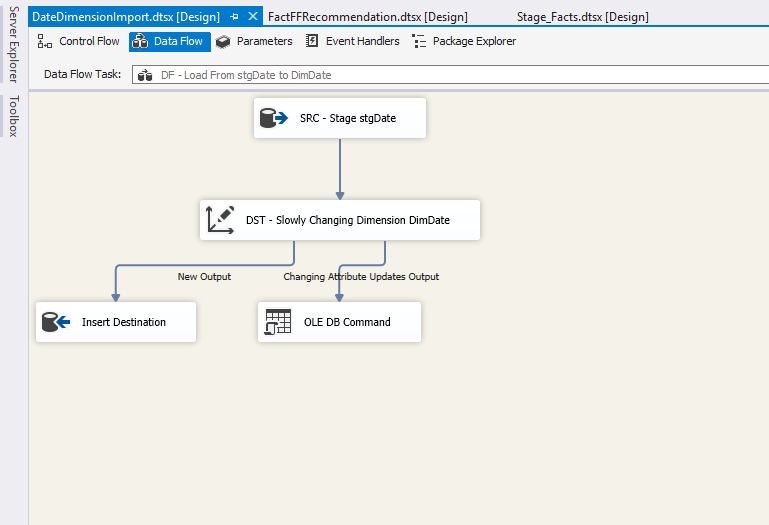


Staging FactItemAnalysis also requires a merge join, this time we use an inner join and merge on the emails:



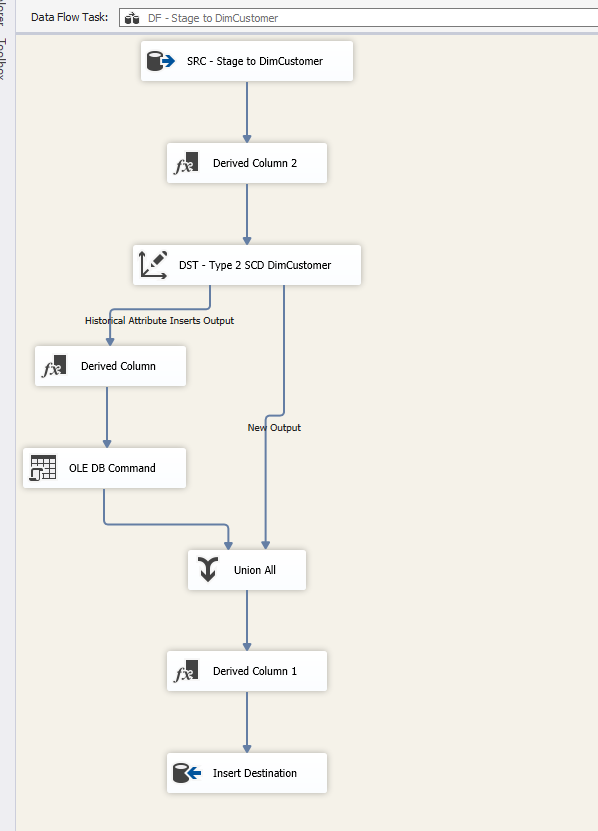
With staging complete, we could move on to loading the data into our data warehouse. To keep things organized, we created separate packages containing the fact table and associated dimension tables that would be loaded.

First, we needed to load DimDates:

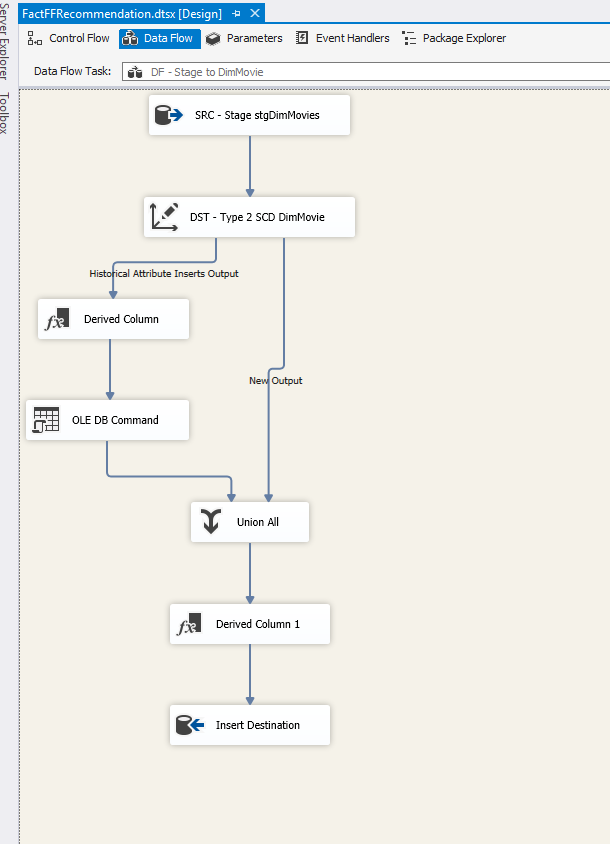


Once DimDate was loaded, we could then focus on our other tables. Once again, we worked on loading our dimension tables:

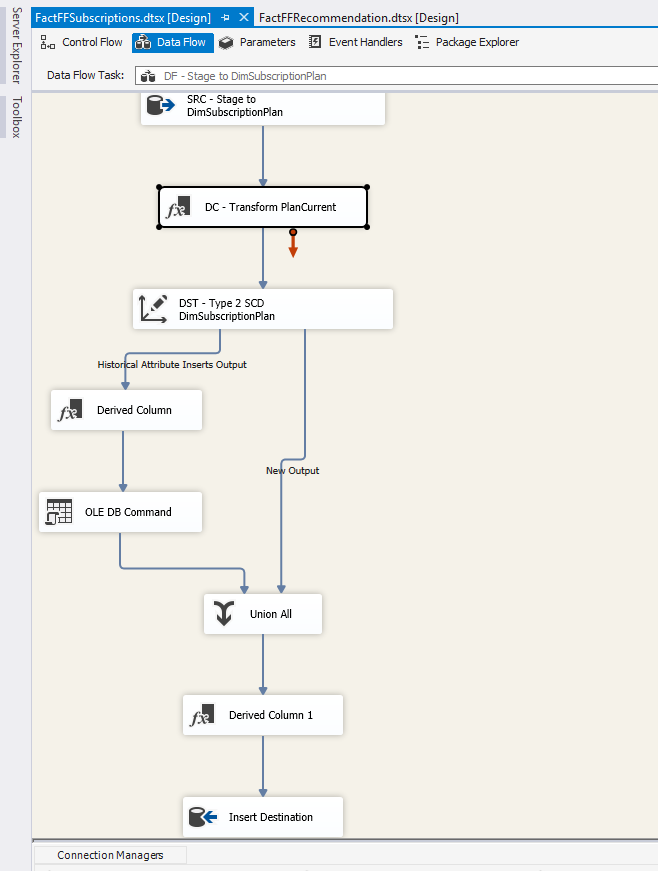
DimCustomer:



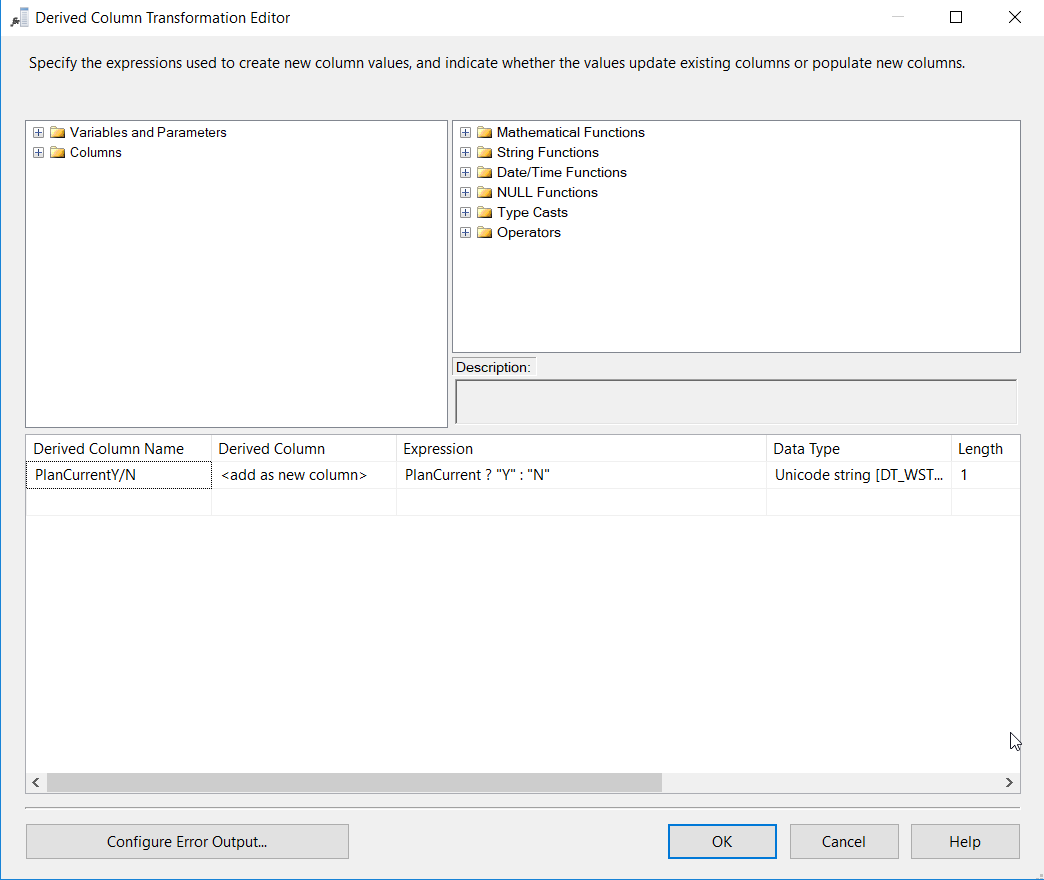
DimMovie:



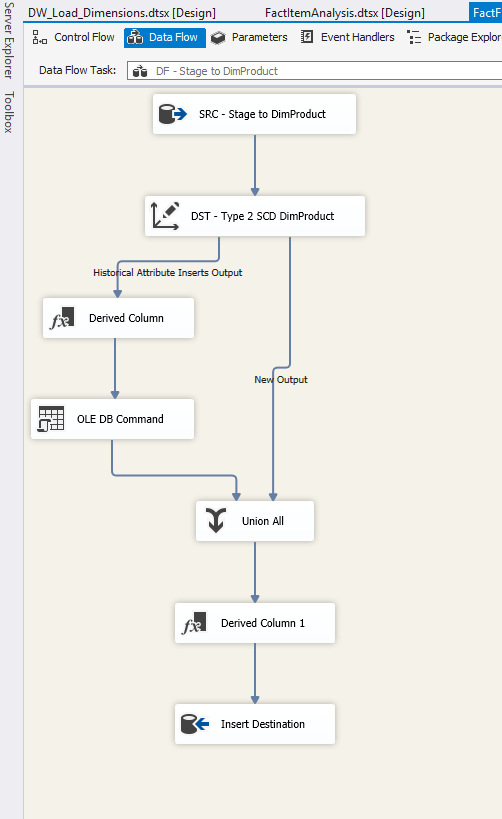
DimSubscriptionPlan:



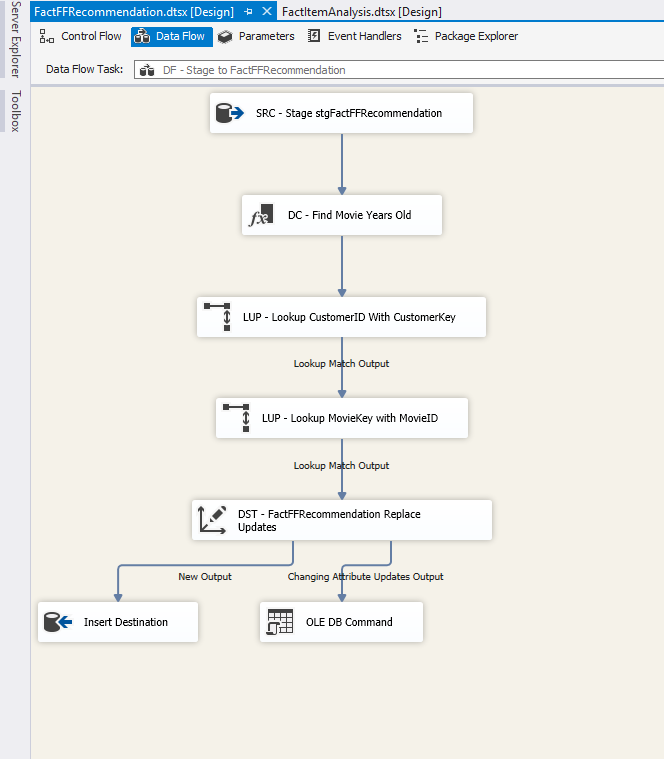
Note that in DimSubscriptionPlan we used derived column to set IsCurrentPlan to accept either Y or N:



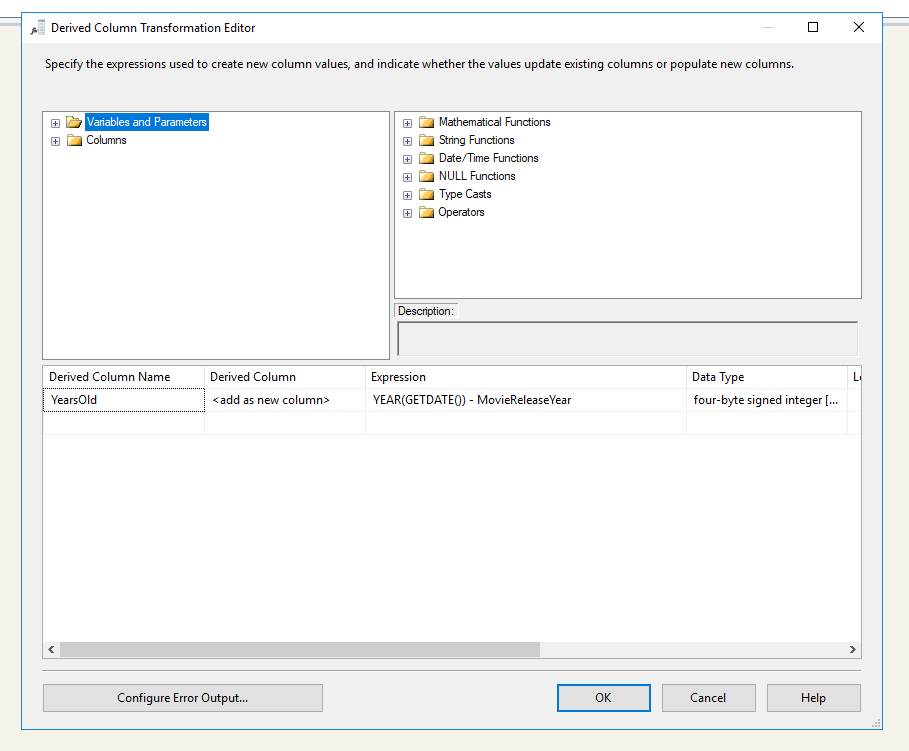
And finally DimProduct:



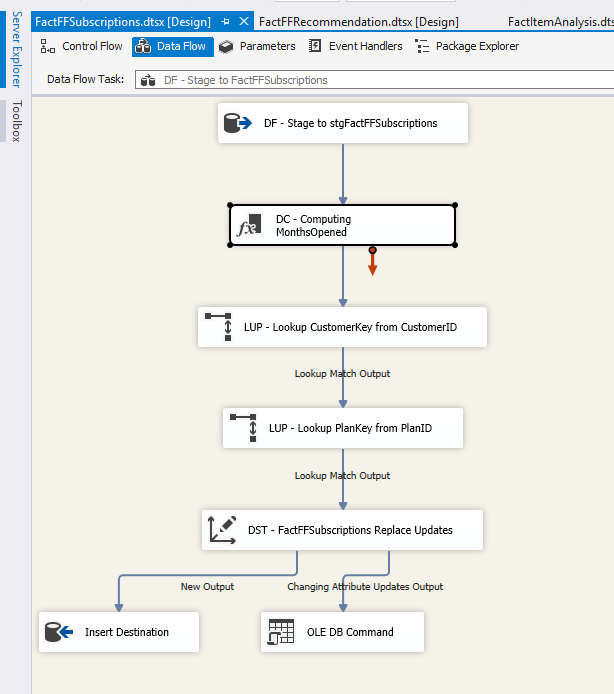
Finally, we could load our fact tables, starting with FactFFRecommendation:



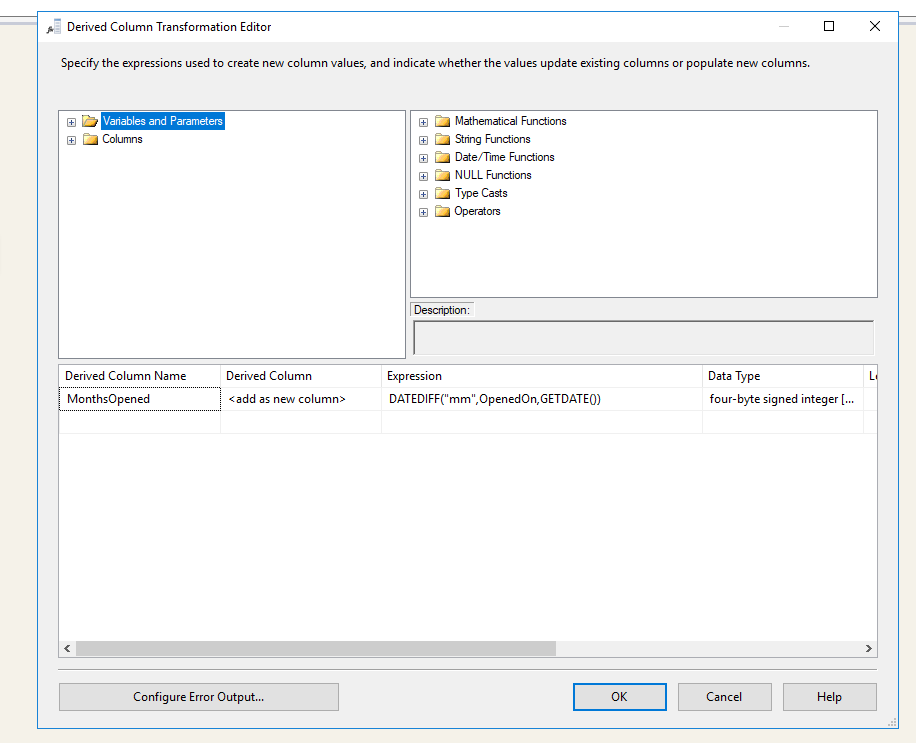
Note the use of the derived column. In this case, we wanted to get how old the movie was by using the GETDATE function:



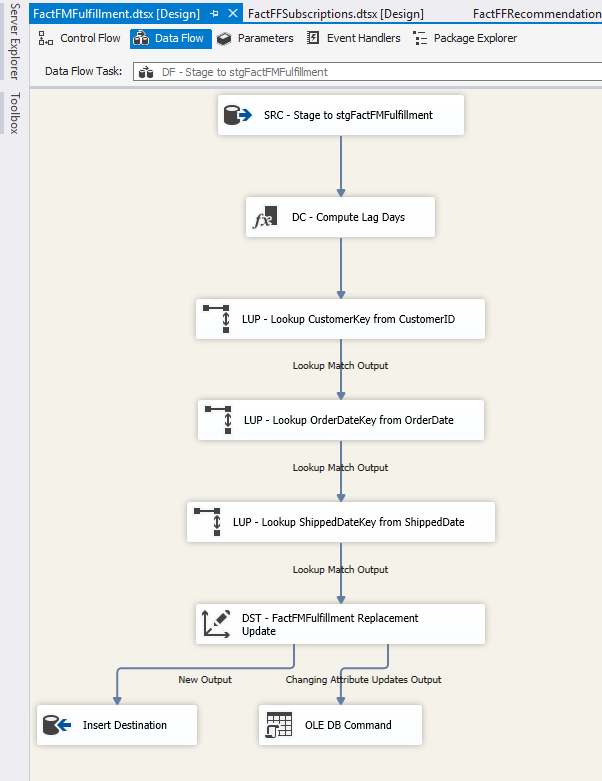
Moving forward, we began to load FactFFSubscriptions.



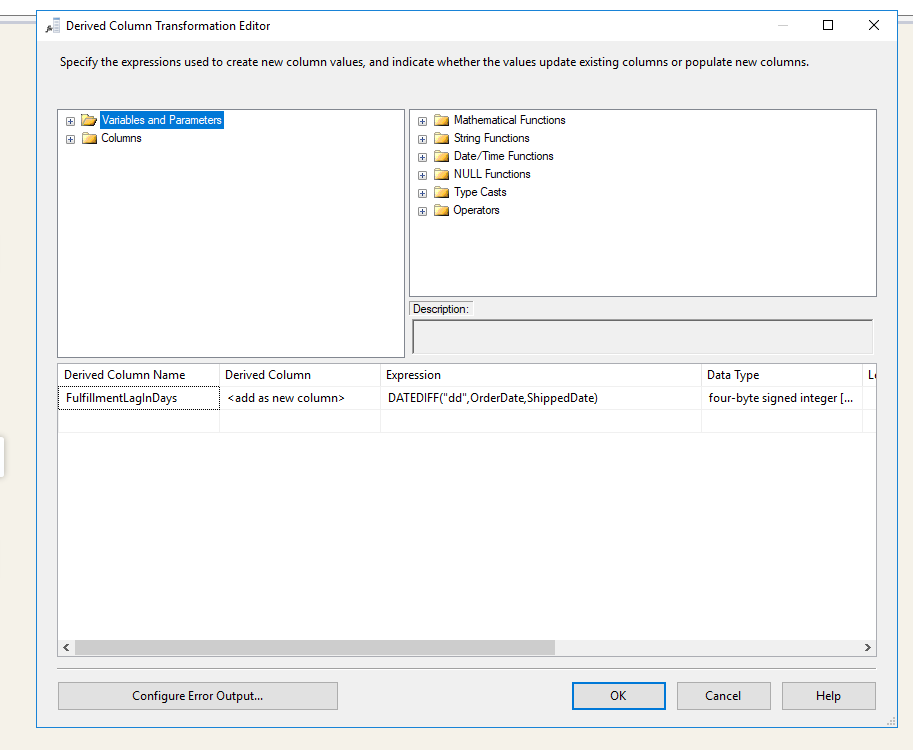
Again, we made use of derived columns to calculate the number of months a Fudgeflix account has been opened.



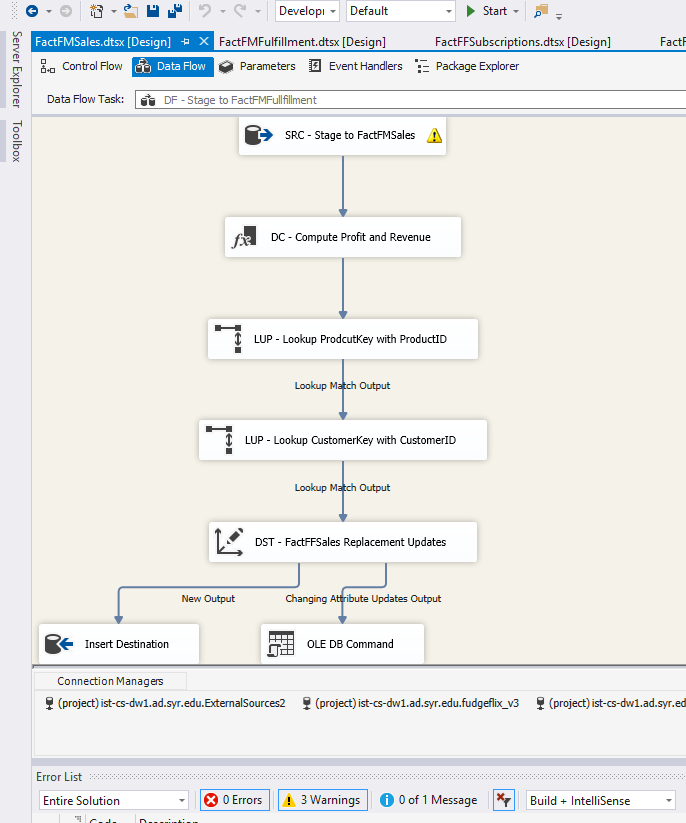
Then, we began loading FactFMFulfillment



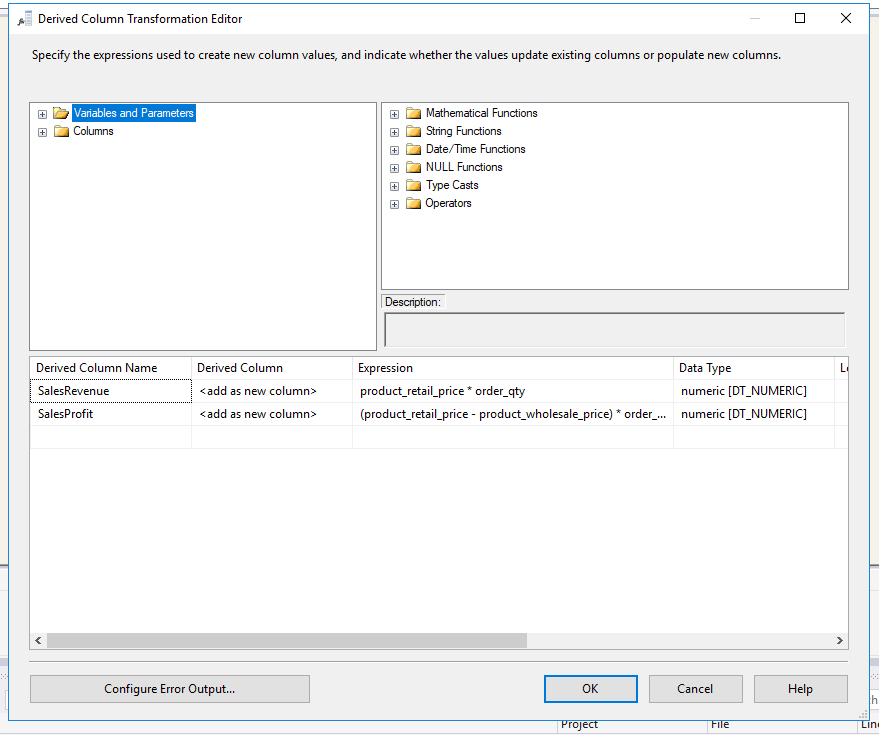
To determine the lag in days between an order place and an order shipped, we made use of a derived column and DATEDIFF:



We then moved on to load FactFMSales:



We created a derived column to calculate two values. First, we needed to determine sales revenue by multiplying product\_retail\_price by order\_qty. Then, we needed to determine SalesProfit by subtracting product\_retail\_price from product\_wholesale\_price and then multiplying that by order\_qty.



Finally we load FactItemAnalysis:

