MEXIA Technical Skill Assessment – Jordan Machan

Friday, February 2, 2018

Author: Jordan Machan

# Part 1: ETL Design

I have broken down the ETL into 5 phases:

1. Database Design – create all of the applicable database objects for this project
2. Process the Stats Canada flat files – I have not built a SSIS package for this
3. Mexia Age – package for processing the derived column Age
4. Mexia FSA – package for processing the derived column FSA
5. Mexia SCD – package for processing slowly changing dimensions (both SCD1 and SCD2)

## Database Design

Having the documentation in hand I was able to determine what database objects that I needed to create. These database objects are as follows:

Databases

1. MexiaCRM – contains the CRM Customer table
2. MexiaSource – contains both staging and source tables
3. MexiaTarget – contains the DimCustomer table

Database MexiaCRM contains the following objects:

* Dbo.Customer – CRM Customer table

Database MexiaSource contains the following objects:

* Dbo.AgeGroup – this table consists of a 120 rows (i.e. Age from 0 to 119 with an applicable AgeGroupID) used for lookup purposes
* Dbo.Customer/Stage – this table in addition to the dbo.Customer\_Stage are the tables where transformation of data and loading of data occurs
* Dbo.ETL – this table keeps track of when the etl packages were last run for the Customer table
* Dbo.SC\_Area – this table contains Statistics Canada Area Information
* Dbo.SC\_Mapping – this table contains Statistics Canada Mapping relational information between FSA and Area
* Dbo.SC\_SurveyResults – this table contains Statistics Canada Survey Results
* Dbo.vw\_CRM\_Customer – this view contains CRM Customers that have an updated date greater than the etl date
* Dbo.vw\_Customer – not sure if it is needed
* Dbo.vw\_SC\_Mapping – this view returns a result set of the most recent survey for a given fsa and survey results
* Dbo.vw\_STG\_Customer – this view contains STG Customers that have an updated date greater than the etl date
* Dbo.vw\_unit\_test\_01 – a view used specifically for displaying the results after running unit test 01
* Dbo.vw\_unit\_test\_02 – a view used specifically for displaying the results after running unit test 02
* Dbo.vw\_unit\_test\_03 – a view used specifically for displaying the results after running unit test 03
* Dbo.vw\_unit\_test\_04 – a view used specifically for displaying the results after running unit test 04

Database MexiaCRM contains the following objects:

* Dbo.DimCustomer – Customer Dimension table

Preliminary Steps

All of the database and database objects were created using scripts.

The main script to drop, create and populate is called mexia\_big\_kahuna.bat

The steps are as follows:

1. Create the mexia support tables
2. Drop and create the customer tables
3. Drop and create the customer views
4. Insert into the CRM customer table – seed this table with some data
5. Update the CRM customer table – to come close to the unit test cases
6. Drop and create the sc tables (i.e. sc aka Statistics Canada)
7. Drop and create the sc views
8. Insert into the sc tables

A Picture is worth a thousand words

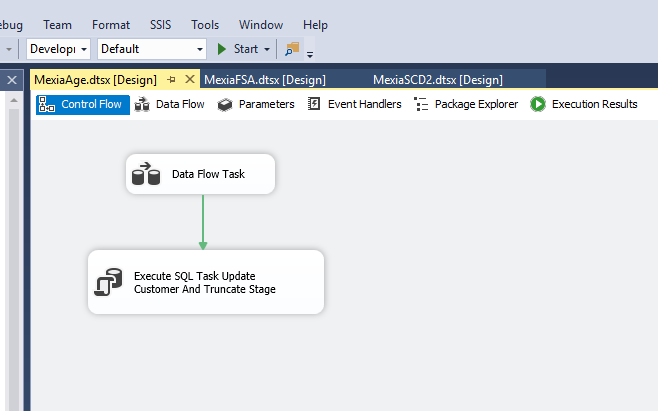
In order to communicate more effectively in what I did for this technical skill assessment I am going to paste a couple of SSIS Package Diagrams.

These diagrams will represent the process flow.

Assumptions:

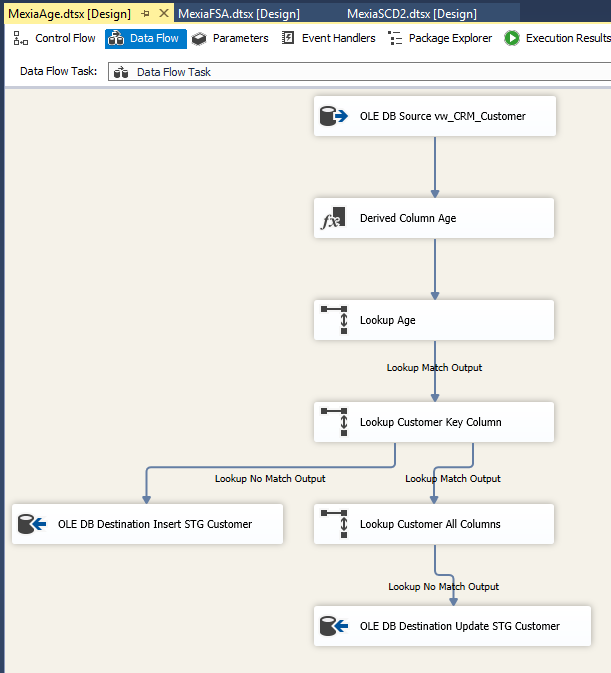
All of the support tables have been created and populated such as the AgeGroup and ETL. All of the sc tables have been created and populated.

## MexiaAge Package



There are 2 MexiaAge Control Flow tasks:

1. Data Flow Task
2. Execute SQL Task Update Customer And Truncate Stage



The Data Flow Task consists of the following steps:

1. OLE DB Source vw\_CRM\_Customer – uses the vw\_CRM\_Customer – give me only the CRM Customers that have not been processed since the last time we ran the ETL
2. Derived Column Age – since the CRM Customer table does not have an Age column we derive this column based on a datediff calculation
3. Lookup Age – since there is an AgeGroup component to this exercise we look for the AgeGroup associated with the derived Age column
4. Lookup Customer Key Column – we compare the CRM customer to the STG customer and if the lookup doesn’t find anything then the crm customer, age and agegroupid are inserted into the stg customer table
5. OLE DB Destination Insert STG Customer – self explanatory
6. Lookup Customer All Columns – if the key matches then lookup or compare all other columns to see if the data is different. If the data is different then update the stg customer columns with the crm customer data along with the age and agegroupid information.
7. OLE DB Destination Update STG Customer – self explanatory

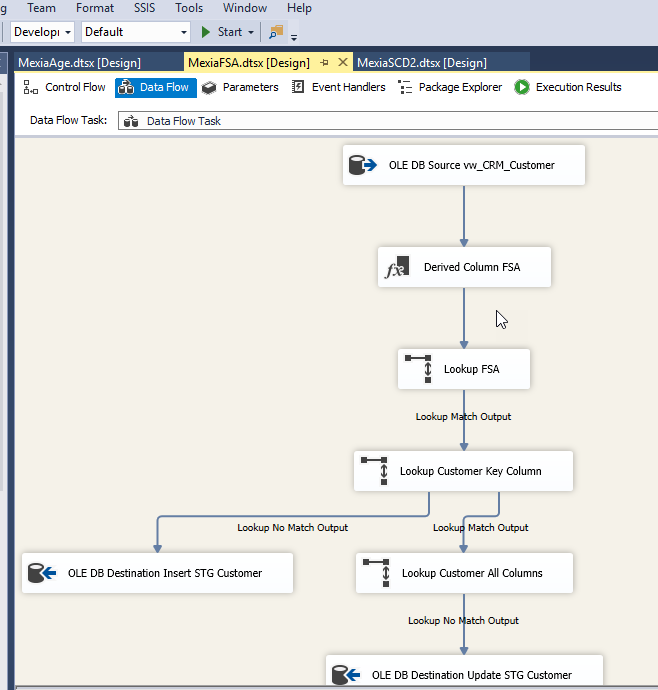
In summary, this etl package inserts and/or updates the stg customer table with crm customer data plus populates the age and agegroupid derived columns.

## MexiaFSA Package

There are 2 MexiaFSA Control Flow tasks:

1. Data Flow Task
2. Execute SQL Task Update Customer And Truncate Stage

This package is similar in functionality to mexiaage.



The Data Flow Task consists of the following steps:

1. OLE DB Source vw\_CRM\_Customer – uses the vw\_CRM\_Customer – give me only the CRM Customers that have not been processed since the last time we ran the ETL
2. Derived Column FSA – since the CRM Customer table does not have an FSA column we derive this column based on a extraction of the 3 left most characters of the postal code
3. Lookup FSA – since there is an FSA/Survey Result component to this exercise we look for the Survey Result associated with the derived FSA column
4. Lookup Customer Key Column – we compare the CRM customer to the STG customer and if the lookup doesn’t find anything then the crm customer, fsa, sc\_areaid, survey year, average information are inserted into the stg customer table
5. OLE DB Destination Insert STG Customer – self explanatory
6. Lookup Customer All Columns – if the key matches then lookup or compare all other columns to see if the data is different. If the data is different then update the stg customer columns with the crm customer data along with the fsa, sc\_areaid, survey year, average information.
7. OLE DB Destination Update STG Customer – self explanatory

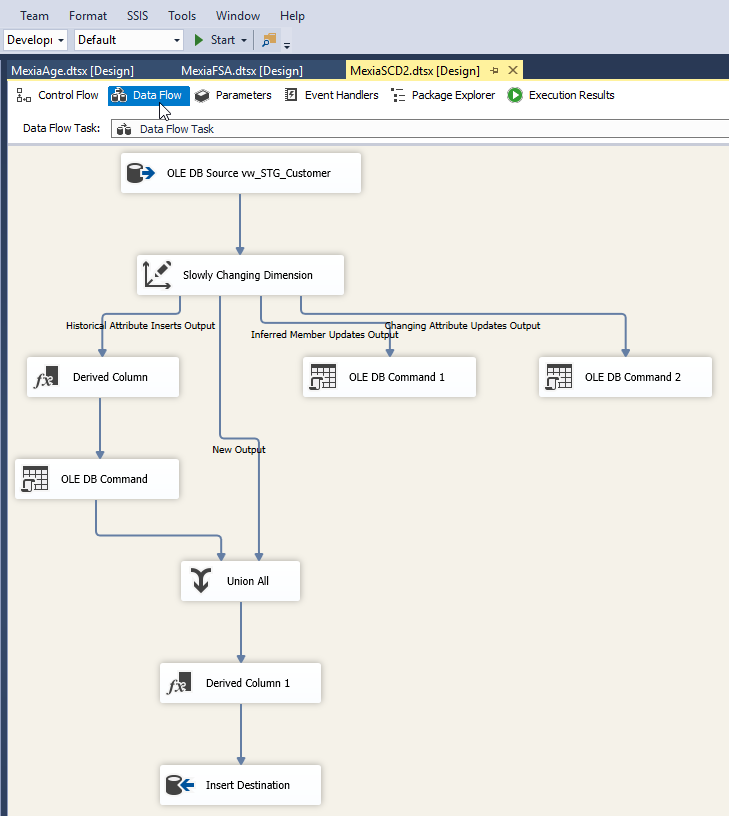
In summary, this etl package inserts and/or updates the stg customer table with crm customer data plus populates the fsa, sc\_areaid, survey year, average information columns.

## MexiaSCD2 Package

There are 2 MexiaSCD2 Control Flow tasks:

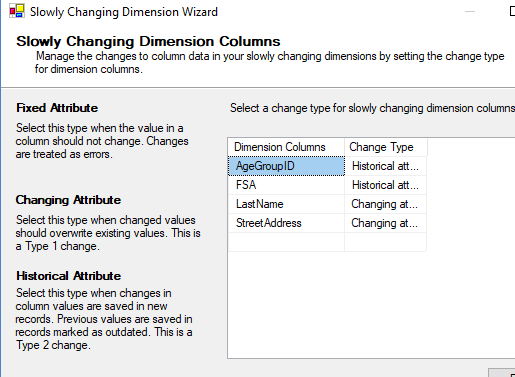
1. Data Flow Task
2. Execute SQL Task Update ETL With Max Updated Date

This package is the slowly changing dimension package. After the data flow task completes a sql task is executed that updates the etl table with the max updated date from the stg customer table.



The Data Flow Task consists of the following steps:

1. OLE DB Source vw\_STG\_Customer – uses the vw\_STG\_Customer – give me only the STG Customers that have not been processed since the last time we ran the ETL
2. Slowly Changing Dimension



Historical Attributes are: AgeGroupID and FSA

Changing Attributes are: LastName and StreetAddress – these attributes were selected to satisfy unit test use case requirements

1. Derived Column->OLE DB Command->Union All->Derived Column 1->Insert Destination
2. New Output->Union All->Derived Column 1->Insert Destination
3. Changing Attribute Updates Output->OLE DB Command 2

In summary, this etl package inserts and/or updates the DimCustomer table with stg customer data plus compares certain columns for SCD1 and SCD2 changes.

After the Data Flow Task executes the ETL table ETL\_DATE is updated with the max updated date from the stg customer table.

## Unit Tests

For each unit test the following needs to happen:

1. Modify data on the CRM Customer table.
2. Run MexiaAge
3. Run MexiaFSA
4. Run MexiaSCD2
5. Query the DimCustomer table to determine if actual results are what are expected

### Unit\_test\_01.sql

USE [MexiaCRM]

GO

update Customer set PostalCode = 'R3C 1X7'

,updateddate = getdate()

where CustomerNumber = 1;

go

--run MexiaAge.dtsx

--run MexiaFSA.dtsx

--run MexiaSCD2.dtsx

--run select \* from vw\_unit\_test\_01



### Unit\_test\_02.sql

USE [MexiaCRM]

GO

update Customer set BirthDate = CONVERT(VARCHAR, '1958-08-30', 121)

,updateddate = getdate()

where CustomerNumber = 3;

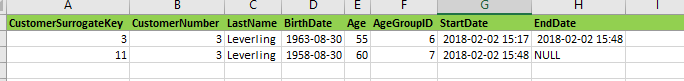
go

--run MexiaAge.dtsx

--run MexiaFSA.dtsx

--run MexiaSCD2.dtsx

--run select \* from vw\_unit\_test\_02



### Unit\_test\_03.sql

USE [MexiaCRM]

GO

update Customer set BirthDate = CONVERT(VARCHAR, '1948-02-19', 121)

,city = 'Brandon'

,postalcode = 'R7A 7R2'

,updateddate = getdate()

where CustomerNumber = 2;

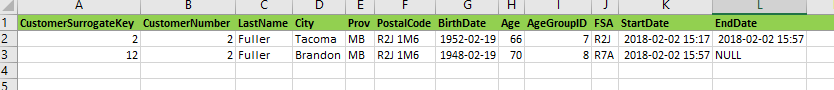
go

--run MexiaAge.dtsx

--run MexiaFSA.dtsx

--run MexiaSCD2.dtsx

--run select \* from vw\_unit\_test\_03



### Unit\_test\_04.sql

USE [MexiaCRM]

GO

update Customer set LastName = 'Smith'

,StreetAddress = '678 Waverley'

,updateddate = getdate()

where CustomerNumber = 4;

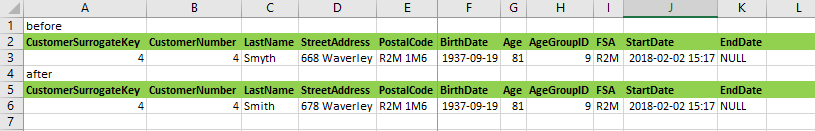
go

--run MexiaAge.dtsx

--run MexiaFSA.dtsx

--run MexiaSCD2.dtsx

--run select \* from vw\_unit\_test\_04



# Processing Requirements

|  |  |  |
| --- | --- | --- |
|  | Processing Requirements Description | Comments |
|  | 1. The “type I” attributes (e.g., name, address) for the current record in the dimension must be updated to stay synchronized with changes in the source CRM system. | done |
|  | 1. A new surrogate key must be assigned whenever a new row is added to the dimension (either due to a net new customer or a type II change). | done |
|  | 1. Changes in the FSA and/or age group should trigger a type II change in the dimension. If the FSA and age group were to change simultaneously (on the same overnight run), only one type II change should be triggered. | done |
|  | 1. The source system does not change its data when someone experiences a birthdate, so we need to consider the updating of age and age group elements of existing records in our dimension as part of our design. | done |
|  | 1. We also need to design an independent process for Stats Can data updates (separate from the daily run) that we can run on demand once every 5 years. This process will loads the raw Stats Canada data from flat files and updates attributes within the dimension for the all of the current customer records. | Not done |

 = key requirement

Additional Notes:

* I did not add a Current Record Flag column. Ran out of time. However, rows where the EndDate is null represents the current record/row.
* I did not put a million rows through this etl process. I am sure that there are ways to improve the performance of these etl process. The bottom line is that these packages work.
* I have zipped all of the MEXIA objects associated with this exercise for review purposes.