Cover Memo

Instructor: John Beswetherick
Course: BUSIT 202
From: Christopher Singleton
Subject: Assignment2 - Creating a Dimensional Model
Date: 01/18/2016

a.

This project is about reviewing an OLTP operational database design and reviewing the OLAP design that was generated from the OLTP Northwind database. This includes finding anything that is missing in the design and/or incorrectly modeled. The purpose of reviewing current design and doing some changes is to be able answer the end users' questions according to their company's business rules. This project also involves extracting, transforming, and loading the data. Reviewing the ETL script is required in order to find what transformations have been done in order to load to the final destination tables.

The most challenging part of the ETL script is to understand the loading of the [Dim Dates] table, especially the two sets of data (two rows) that are combined (union) together, which includes "unknown" or "corrupted" values. However, I finally figured it out what that means and why it is necessary. These values are going to be used when the date is missing or un-known during ETL processing.

From what I submitted on the discussion board from the first assignment, it seems that from the OLTP Northwind database that the [Suppliers] table should not be flattened into the Products table. It belongs in another Data Mart.

Please note all tasks required in this assignment have been completed.

b.

Comments on the Operational Database and the DWNorthwindOrders data warehouse.

Northwind Operational Database (OLTP):

In the Order Details table, [Order ID] should not be the Primary Key, it should be the Foreign Key since table Orders and [Order Details] have a One-to-Many relationship. In the [Order Details] table, Primary Key is missing, so [Order Detail Line ID] as the primary key should be added. Also the [Order Detail Line ID] is formed in order to reduce any conflict with other Data Marts that you may decide to build in the future to satisfy the business needs.

In addition, Shippers and Orders table have a One-to-Many relationship. There should be a Foreign Key constraint in the Orders table. Therefore, the [Shipper ID] should be added to the Order table as a Foreign Key. Due to this One-to-Many relationship, the Company Name can be found by joining these two tables on the common column [Shipper ID], so the column [Ship Via] is not needed (should be eliminated) in the Order table.

Lastly, [Product ID] in the Products table was set as a Primary Key. It should be a Foreign Key since Products and [Order Details] have a One-to-Many relationship.

DWNorthwindOrders Data warehouse Database (OLAP):

It is recommended to rename the [Fact Orders] table to [Fact Sales Orders] because if you decide to create another DataMart for purchasing in the future, you could get the names easily mixed up (name it more specific). As stated above, primary key [Order Detail Line ID] should be added to the [Order Details] table in the OLTP database. Thus, primary key [Order Detail Line ID] should be added to the [Fact Sales Orders] table forming a composite key with the other primary key [Order ID].

Shippers dimension is missing. If a manager wants to know the total quantity shipped by which shipping company within a specific period of time, the current design does not have the dimension to slice the measures. In order to be able to answer this question. [Dim Shippers] dimension table is added, and primary key [Shipper Key] is added to the [Fact Sales Orders] as a foreign key constraint.

Moreover, a fact less fact table should be added to the current design since Employees table and Territories table have Many-to-Many relationship in the OLTP database. So, the [Factless Fact Employee Territories] table is created linking both [Dim Employees] and [Dim Territories] with [Employee Key] and [Territory Key] as foreign keys. Note that [Dim Territories] dimension should be de-normalized, that is, the attribute in the Region table should be flattened in to the dimension [Dim Territories].

There is a similar situation as above between Customers table and [Customer Demographics]. They have Many-to-Many relationship, so [factless Fact Customer Demo] table is added to it with [Customer Key] and [Customer Type Key] as foreign keys bridging the two dimensions. This way, the business questions can be answered by using these dimensions added to the design.

Finally, all the primary keys of each dimension stored in the [Fact Sales Orders] table should be foreign keys instead of primary keys, such as [Customer Key], [Employee Key], [Product Key], [Order Date Key], [Required Date Key], [Shipped Date Key]. Also, in [Dim Customers] dimension table, the data type of [Customer ID] should be integer (int).

Comments on Transformations in the ETL script:

Transformations for the DimCustomer table:

[Company Name] in the source table (Customers) with datatype nvarchar(40) has been converted to nvarchar(100) in the dimensional table, which gives more character space.

City in the source table (Customers) with nvarchar(15) has been converted to nvarchar(50) in the dimensional table, which gives more character space.

Region in the source table (Customers): if it is null, fill column Region with the value of Country, and then convert the datatype from nvarchar(15) to nvarchar(50).

Country in the source table (Customers) with nvarchar(15) has been converted to nvarchar(50) in the dimensional table, which gives more character space.

Transformations for the DimEmployees table:

[First Name] and [Last Name] in the source (OLTP) table (Employees) are concatenated as [Employee Name] in the [Dim Employees] table, and then the datatype has been converted to nvarchar(100) of the DWNorthwindOrders database.

[Reports To] column can be null in the Northwind Employees table. If the [Reports To] column is null, then use [Employee ID] as its value for the column [Manager Key] in the [Dim Employees] table of the DWNorthwindOrders Data Warehouse (OLAP).

Likewise, [Manager ID] in the [Dim Employees] table (DWNorthwindOrders OLAP) will be filled with the Northwind (OLTP) [Employees] [Reports To] column data, but If the column is null then it will use [Employees ID] as its value.

Transformations for the DimProducts table:

[Product Name] in the source table (Products) with the datatype nvarchar(40) has been converted to nvarchar(100). [Category Name] in the source table (Categories) with datatype nvarchar(15) has been converted to [Product Category] in the [Dim Products] table with a datatype nvarchar(100), which gives more character space.

[Unit Price] in the source table [Products] with the datatype money, has been converted to the [Product Std Price] in [Dim Products] with the datatype decimal(18,4).

Discontinued in the source table [Products] with the datatype bit, has been converted into the [Product Is Discontinued] in the [Dim Products] table with the datatype nChar(1) with a flag that if Discontinue = 1 then it is ‘T’, else it is ‘F’. This converts the origin datatype from 1 to F and all other values to T.

(Continued)

Transformations for the DimDates table:

To start off with setting up the columns, there is a variable interval declared called @DateInProcess, which gives a start date, end date and loops through iterating each day in order to create the [Date], [Date Name], [Month], [Month Name], Quarter, [Quarter Name], [Year], and [Year Name].

Then the values are created for the @DateInProcess, starting with [Date Name] for the weekday concatenated with a value of nvarchar(20).

Month and Year is given a number with through the variable @DateInProcess while iterating through the loops.

I noticed that when the [Date Name] is specifically put on the outside of the parentheses with Month or Quarter inside the parentheses, then it will give it in character format value (nvarchar(20)). Take for instance: `DateName( quarter, @DateInProcess )` Means: Display the quarter in character format value using the variable @DateInProcess, this will give the quarter name based on the [Date Name] value. In other words, the [Date Name] value understands the quarter, then it will give it as a number in character format. The [Month Name] also uses the [Date Name] value in order to display in character format also.

The [Quarter Name] column is displayed by concatenating Q with the added [Date Name] format with quarter and the added year, all together with a value of nvarchar(50).

The [Year Name] column is displayed by converting the year while using the @DateInProcess variable with a value of nvarchar(50).

Next is populating the data into the DWNorthwindOrders data warehouse [Dim Dates] columns using insert while iterating through the loops on each row. During this time missing values are configured with conversions to populate the [Date Key], [Date], [Date Name], [Month], [Month Name], [Quarter], [Quarter Name], [Year], and [Year Name] with -1 for Unknown data or -2 for corrupt data.

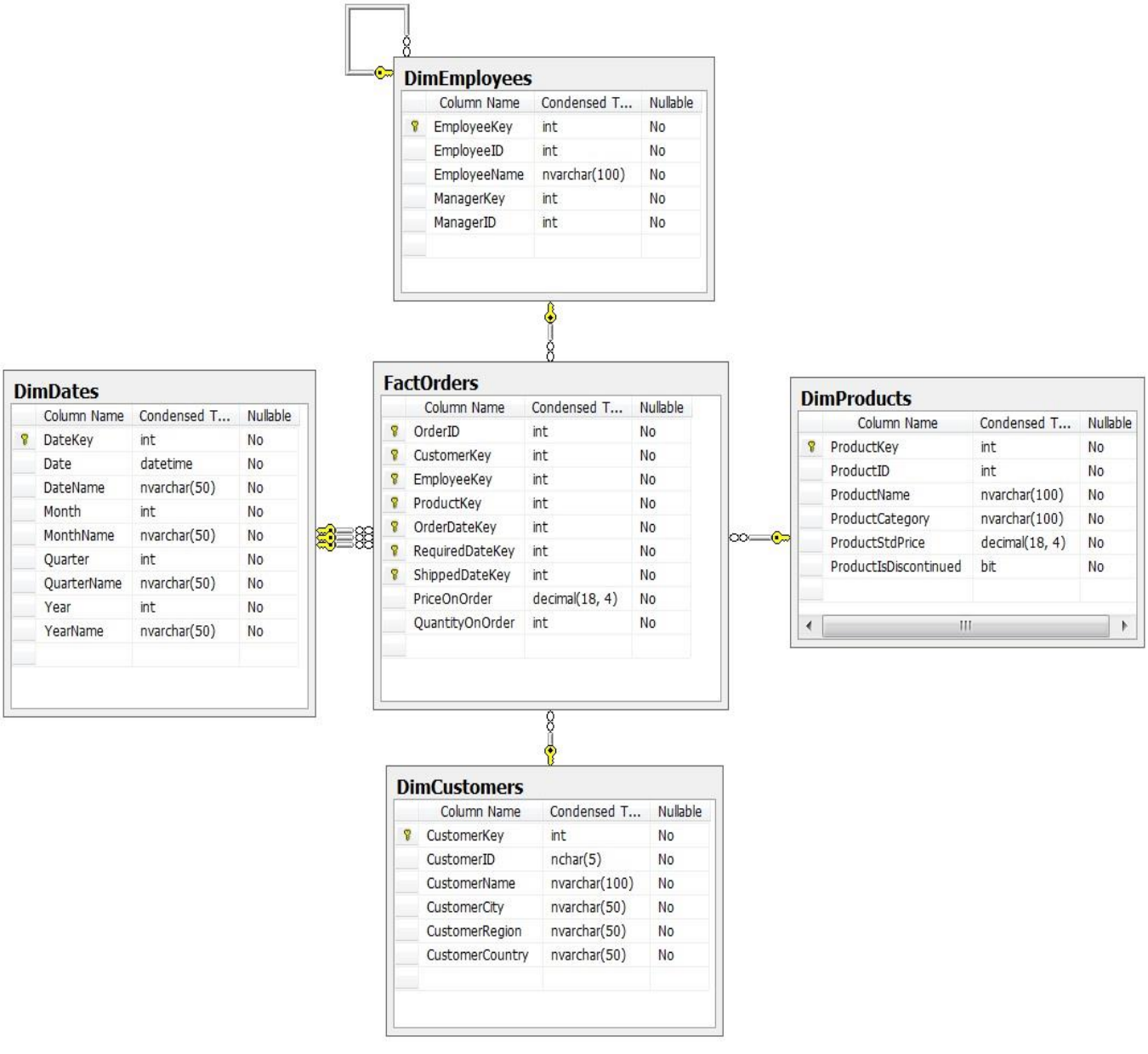
Transformations for the FactOrders table:

Using the source data in the [Order Details] table (Northwind-OLTP), the [Unit Price] is summed and transferred into the [Price On Order] column of the [Fact Orders] table (DWNorthwindOrders-OLAP). Likewise, the quantity is converted and summed while being transferred from the [Order Details] table (Northwind-OLTP) to the [Fact Orders] table (DWNorthwindOrders-OLAP) [Quantity On Order] column.

In the Orders table (Northwind-OLTP), if any of the cell's [Order Date], [Required Date] and/or [Shipped Date] are null, then the date will be will be replaced with `'1900-01-01 00:00:00.000'` in the same named columns of the (DWNorthwindOrders-OLAP) [Fact Orders] table.

Note: At the end of all the transformations, the foreign keys are added to the [Fact Orders] and a foreign key is also added to the [Dim Employees] table to allow easy access (continuity) of all the data that has just been populated.

c. **Original design of DWNorthwindOrders.**



d. **Updated design of DWNorthwindOrders.**

DWNorthwindOrders (OLAP Suggested Revisions)

Added the [Dim Customer Demo] table then created the interlinking [Factless Fact Customer Demo] table, then created the [Dim Territories] interlinking [Factless Fact Employee Territories] table and then finally added the [Dim Shippers] table to the DWNorthwindOrders data warehouse. Please note that the [Factless Fact Customer Demo] and [Factless Fact Employee Territories] interlinking tables will satisfy the many to many relationship requirements. Changed [Fact Orders] to [Fact Sales Orders] with [Order ID] and [Order Detail Line ID] as primary keys (renaming the [Fact Orders], which makes it more specific and reduced a less possible chance of confusion with other Data Marts in the future. Foreign keys in [Fact Sales Order] includes [Customer Key], [Employee Key], [Product Key], [Order Date Key], [Required Date Key], [Shipped Date Key] and [Shipper Key]. Changed [Price on Order] to the decimal datatype in the table creation script, instead of “money” (decimal is the standard industry datatype). Added necessary columns for the newly created tables to reflect what is needed to transfer the data in each new table that was added.