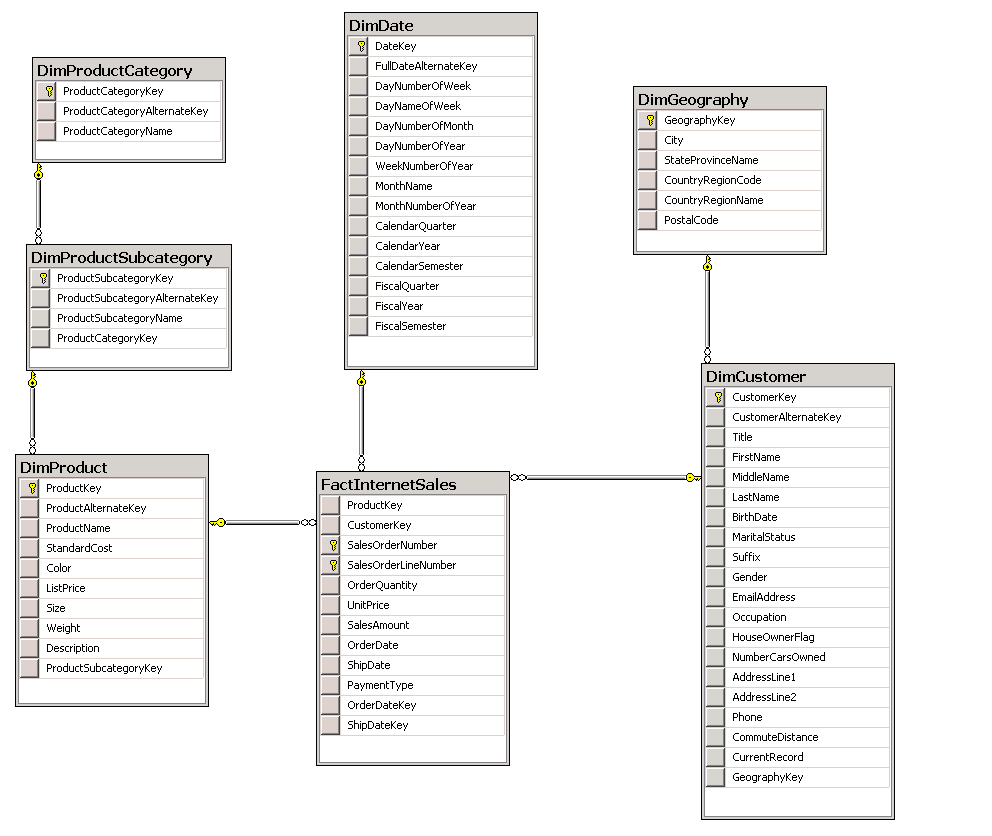
# SSIS Project – Part 1

1. Create and SSIS project “Staging\_DW\_Init”. Using the scripts provided below, create two packages, each should create one database: Staging and DW database. Please take time to review the scripts and understand what they do. Use “Execute SQL Task” to send these scripts to the server. You are free to decide how may SQL tasks you need to create for each package. Execute package.
2. Using the SQL Server Management Studio, review the two databases created. For the DW, create a database diagram. It should look like this:



1. Create another SSIS Project, called “Internet Sales Project”. This project should contain two packages: “Extract Customers” and “Extract Internet Sales”. The source DB on the the localhost server is InternetSales, the destination would be the staging DB that you created in step 1.
   1. “Extract Customers” should load the first 1000 customers into the staging Customers table. No transformation required.
   2. “Extract Internet Sales” should use this query, but load only first 1000 order lines into the staging DB. In the data flow, include a derived column that would calculate the sales amount

SELECT

od.ProductKey,

oh.CustomerKey,

od.SalesOrderNumber,

od.SalesOrderLineNumber,

od.OrderQuantity,

od.UnitPrice,

oh.OrderDate,

oh.ShipDate,

ISNULL(pt.PaymentTypeName, 'Other') PaymentType

FROM SalesOrderDetail od

JOIN SalesOrderHeader oh ON od.SalesOrderNumber = oh.SalesOrderNumber

JOIN PaymentTypes pt ON oh.PaymentType = pt.PaymentTypeKey

1. Make sure to send a notification email about success or failure for each of the two packages, created in step 3.

USE master

GO

-- Create the Staging database

CREATE DATABASE SsisProjectStaging

GO

-- Create staging tables

USE SsisProjectStaging

GO

-- Customers

CREATE TABLE [dbo].[Customers](

[CustomerBusinessKey] [nvarchar](15) NOT NULL,

[Title] [nvarchar](8) NULL,

[FirstName] [nvarchar](50) NULL,

[MiddleName] [nvarchar](50) NULL,

[LastName] [nvarchar](50) NULL,

[BirthDate] [date] NULL,

[MaritalStatus] [nchar](1) NULL,

[Suffix] [nvarchar](10) NULL,

[Gender] [nvarchar](1) NULL,

[EmailAddress] [nvarchar](50) NULL,

[Occupation] [nvarchar](100) NULL,

[HouseOwnerFlag] [nchar](1) NULL,

[NumberCarsOwned] [tinyint] NULL,

[AddressLine1] [nvarchar](120) NULL,

[AddressLine2] [nvarchar](120) NULL,

[City] [nvarchar](30) NULL,

[StateProvinceName] [nvarchar](50) NULL,

[CountryRegionCode] [nvarchar](3) NULL,

[CountryRegionName] [nvarchar](50) NULL,

[PostalCode] [nvarchar](15) NULL,

[Phone] [nvarchar](20) NULL,

[CommuteDistance] [nvarchar](15) NULL

)

-- InternetSales

CREATE TABLE [dbo].[InternetSales](

[ProductBusinessKey] [nvarchar](25) NOT NULL,

[CustomerBusinessKey] [int] NOT NULL,

[SalesOrderNumber] [nvarchar](20) NOT NULL,

[SalesOrderLineNumber] [tinyint] NOT NULL,

[OrderQuantity] [smallint] NULL,

[UnitPrice] [money] NULL,

[SalesAmount] [money] NULL,

[OrderDate] [datetime] NULL,

[ShipDate] [datetime] NULL,

[PaymentType] [varchar](15) NULL,

[ProductName] [nvarchar](50) NOT NULL,

[StandardCost] [money] NULL,

[Color] [nvarchar](15) NOT NULL,

[ListPrice] [money] NULL,

[Size] [nvarchar](50) NULL,

[Weight] [float] NULL,

[Description] [nvarchar](400) NULL,

[ProductSubcategoryBusinessKey] [int] NULL,

[ProductSubcategoryName] [nvarchar](50) NOT NULL,

[ProductCategoryBusinessKey] [int] NULL,

[ProductCategoryName] [nvarchar](50) NOT NULL

)

GO

Use master

GO

-- Create the AWDataWarehouse database

CREATE DATABASE SsisProjectDataWarehouse

GO

-- Create dimension tables

USE SsisProjectDataWarehouse

GO

-- Product

CREATE TABLE [dbo].[DimProduct](

[ProductKey] [int] IDENTITY(1,1) NOT NULL PRIMARY KEY CLUSTERED,

[ProductAlternateKey] [nvarchar](25) NULL,

[ProductName] [nvarchar](50) NOT NULL,

[StandardCost] [money] NULL,

[Color] [nvarchar](15) NOT NULL,

[ListPrice] [money] NULL,

[Size] [nvarchar](50) NULL,

[Weight] [float] NULL,

[Description] [nvarchar](400) NULL

)

GO

CREATE TABLE [dbo].[DimCustomer](

[CustomerKey] [int] IDENTITY(1,1) NOT NULL PRIMARY KEY CLUSTERED,

[CustomerAlternateKey] [nvarchar](15) NOT NULL,

[Title] [nvarchar](8) NULL,

[FirstName] [nvarchar](50) NULL,

[MiddleName] [nvarchar](50) NULL,

[LastName] [nvarchar](50) NULL,

[BirthDate] [date] NULL,

[MaritalStatus] [nchar](1) NULL,

[Suffix] [nvarchar](10) NULL,

[Gender] [nvarchar](1) NULL,

[EmailAddress] [nvarchar](50) NULL,

[Occupation] [nvarchar](100) NULL,

[HouseOwnerFlag] [nchar](1) NULL,

[NumberCarsOwned] [tinyint] NULL,

[AddressLine1] [nvarchar](120) NULL,

[AddressLine2] [nvarchar](120) NULL,

[CountryRegionCode] [nvarchar](3) NULL,

[Phone] [nvarchar](20) NULL,

[CommuteDistance] [nvarchar](15) NULL,

[CurrentRecord] bit

)

GO

-- Create Internet Sales fact table

CREATE TABLE [dbo].[FactInternetSales](

[ProductKey] [int] NOT NULL REFERENCES [dbo].[DimProduct] ([ProductKey]),

[CustomerKey] [int] NOT NULL REFERENCES [dbo].[DimCustomer] ([CustomerKey]),

[SalesOrderNumber] [nvarchar](20) NOT NULL,

[SalesOrderLineNumber] [tinyint] NOT NULL,

[OrderQuantity] [smallint] NOT NULL,

[UnitPrice] [money] NOT NULL,

[SalesAmount] [money] NOT NULL,

[OrderDate] [datetime] NULL,

[ShipDate] [datetime] NULL,

[PaymentType] [nvarchar](15) NULL,

CONSTRAINT [PK\_FactInternetSales\_SalesOrderNumber\_SalesOrderLineNumber] PRIMARY KEY CLUSTERED

(

[SalesOrderNumber] ASC,

[SalesOrderLineNumber] ASC

)

)

GO

-- IMPLEMENT PRODUCT CATEGORY SNOWFLAKE TO CREATE A NATURAL HIERARCHY

--Create DimProductCategory table

CREATE TABLE [dbo].[DimProductCategory](

[ProductCategoryKey] [int] IDENTITY(1,1) NOT NULL PRIMARY KEY CLUSTERED,

[ProductCategoryAlternateKey] [int] NULL,

[ProductCategoryName] [nvarchar](50) NOT NULL

)

GO

-- Create DimProductSubcategory table

CREATE TABLE [dbo].[DimProductSubcategory](

[ProductSubcategoryKey] [int] IDENTITY(1,1) NOT NULL PRIMARY KEY CLUSTERED,

[ProductSubcategoryAlternateKey] [int] NULL,

[ProductSubcategoryName] [nvarchar](50) NOT NULL,

[ProductCategoryKey] [int] NULL REFERENCES [dbo].DimProductCategory(ProductCategoryKey)

)

GO

--Add foreign key to DimProductSubcategory

ALTER TABLE [dbo].DimProduct

ADD ProductSubcategoryKey int NULL REFERENCES [dbo].DimProductSubcategory(ProductSubcategoryKey)

GO

-- IMPLEMENT GEOGRAPHY SNOWFLAKE TO SHARE A HIERARCHY ACROSS DIMENSIONS

-- Create DimGeography Table

CREATE TABLE [dbo].[DimGeography](

[GeographyKey] [int] IDENTITY(1,1) NOT NULL PRIMARY KEY CLUSTERED,

[City] [nvarchar](30) NULL,

[StateProvinceName] [nvarchar](50) NULL,

[CountryRegionCode] [nvarchar](3) NULL,

[CountryRegionName] [nvarchar](50) NULL,

[PostalCode] [nvarchar](15) NULL,

)

GO

-- Modify DimCustomer table to add foreign key to DimGeography

ALTER TABLE [dbo].DimCustomer

ADD GeographyKey int NULL REFERENCES [dbo].DimGeography(GeographyKey)

GO

-- Create DimDate dimension table

CREATE TABLE [dbo].[DimDate](

[DateKey] [int] NOT NULL PRIMARY KEY CLUSTERED,

[FullDateAlternateKey] [date] NOT NULL,

[DayNumberOfWeek] [tinyint] NOT NULL,

[DayNameOfWeek] [nvarchar](10) NOT NULL,

[DayNumberOfMonth] [tinyint] NOT NULL,

[DayNumberOfYear] [smallint] NOT NULL,

[WeekNumberOfYear] [tinyint] NOT NULL,

[MonthName] [nvarchar](10) NOT NULL,

[MonthNumberOfYear] [tinyint] NOT NULL,

[CalendarQuarter] [tinyint] NOT NULL,

[CalendarYear] [smallint] NOT NULL,

[CalendarSemester] [tinyint] NOT NULL,

[FiscalQuarter] [tinyint] NOT NULL,

[FiscalYear] [smallint] NOT NULL,

[FiscalSemester] [tinyint] NOT NULL

)

GO

-- Create relationships to DimDate

ALTER TABLE [dbo].FactInternetSales

ADD

OrderDateKey int NULL REFERENCES [dbo].DimDate(DateKey),

ShipDateKey int NULL REFERENCES [dbo].DimDate(DateKey)

GO

-- Create indexes on date key fields

CREATE NONCLUSTERED INDEX [IX\_FactInternetSales\_OrderDateKey] ON [dbo].FactInternetSales

(

[OrderDateKey] ASC

)

GO

CREATE NONCLUSTERED INDEX [IX\_FactInternetSales\_ShipDateKey] ON [dbo].FactInternetSales

(

[ShipDateKey] ASC

)

GO

-- Populate DimDate dimension table with values from 1/1/2000 to 1/1/2003

-- declare variables to hold the start and end date

DECLARE @StartDate datetime

DECLARE @EndDate datetime

--- assign values to the start date and end date we

-- want our reports to cover (this should also take

-- into account any future reporting needs)

SET @StartDate = '01/01/2000'

SET @EndDate = getdate()

-- using a while loop increment from the start date

-- to the end date

DECLARE @LoopDate datetime

SET @LoopDate = @StartDate

WHILE @LoopDate <= @EndDate

BEGIN

-- add a record into the date dimension table for this date

INSERT INTO dbo.DimDate VALUES

(

CAST(CONVERT(VARCHAR(8), @LoopDate, 112) AS int) , -- date key

@LoopDate, -- date alt key

datepart(dw, @LoopDate), -- day number of week

datename(dw, @LoopDate), -- day name of week

Day(@LoopDate), -- day number of month

datepart(dy, @LoopDate), -- day of year

datepart(wk, @LoopDate), -- week of year

datename(mm, @LoopDate), -- month name

Month(@LoopDate), -- month number of year

datepart(qq, @LoopDate), -- calendar quarter

Year(@LoopDate), -- calendar year

CASE

WHEN Month(@LoopDate) < 7 THEN 1

ELSE 2

END, -- calendar semester

CASE

WHEN Month(@LoopDate) IN (1, 2, 3) THEN 3

WHEN Month(@LoopDate) IN (4, 5, 6) THEN 4

WHEN Month(@LoopDate) IN (7, 8, 9) THEN 1

WHEN Month(@LoopDate) IN (10, 11, 12) THEN 2

END, -- fiscal quarter (assuming fiscal year runs from Jul to June)

CASE

WHEN Month(@LoopDate) < 7 THEN Year(@LoopDate)

ELSE Year(@Loopdate) + 1

END, -- Fiscal year

CASE

WHEN Month(@LoopDate) > 6 THEN 1

ELSE 2

END -- fiscal semester

)

-- increment the date by 1 day and do next loop

SET @LoopDate = DateAdd(dd, 1, @LoopDate)

END

GO