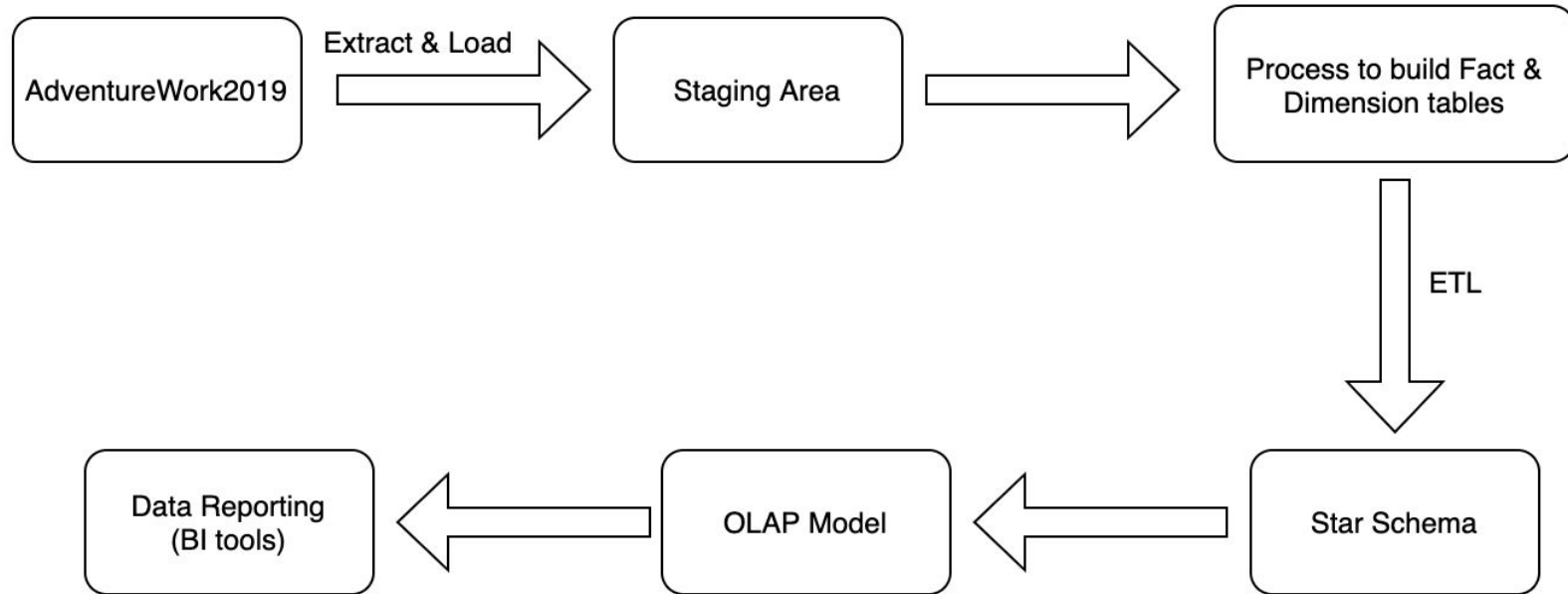


# Introduction

- Create dimensional model for DW that will support:  
Sales analysis  
Inventory analysis
- DI: Load data sources into DW  
SOR: AdventureWorks 2019  
Tool: MS SQL Server (SSIS)
- BI: Develop visualization dashboards & reports  
Tool: R



# Data Flow Process



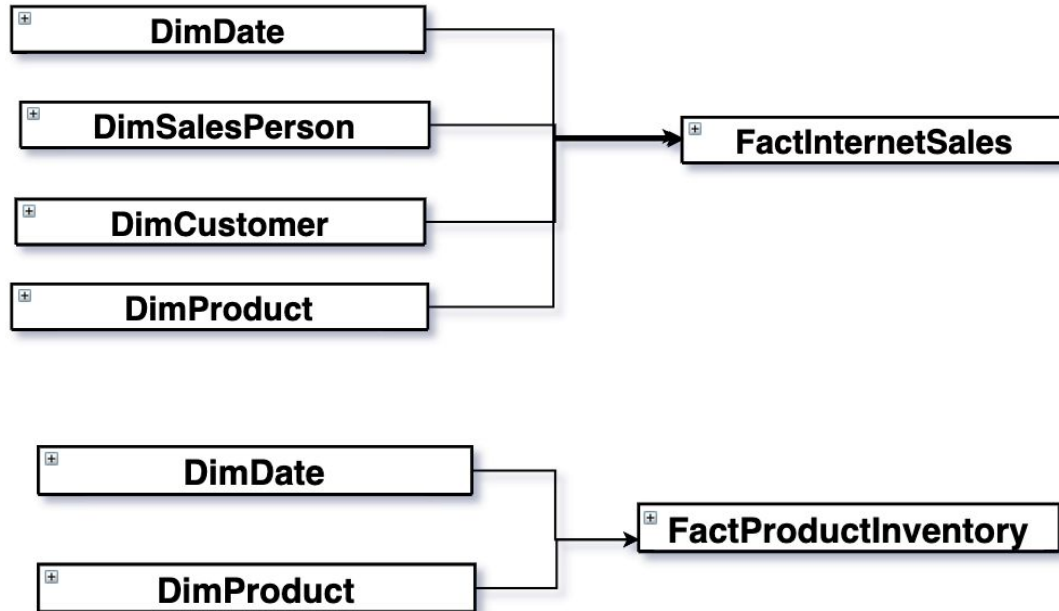
# Data Warehouse Model ---- Facts and Dims

FactInternetSales	
PK	<u>SalesOrderID</u>
PK	<u>SalesOrderDetailID</u>
FK	OrderDateKey
FK	DueDateKey
FK	ShipDateKey
FK	SalesPersonKey
FK	CustomerKey
FK	ProductKey
CarrierTrackingNumber	
OrderQty	
UnitPrice	
UnitPriceDiscount	
LineTotal	
Status	
Freight	
SubTotal	

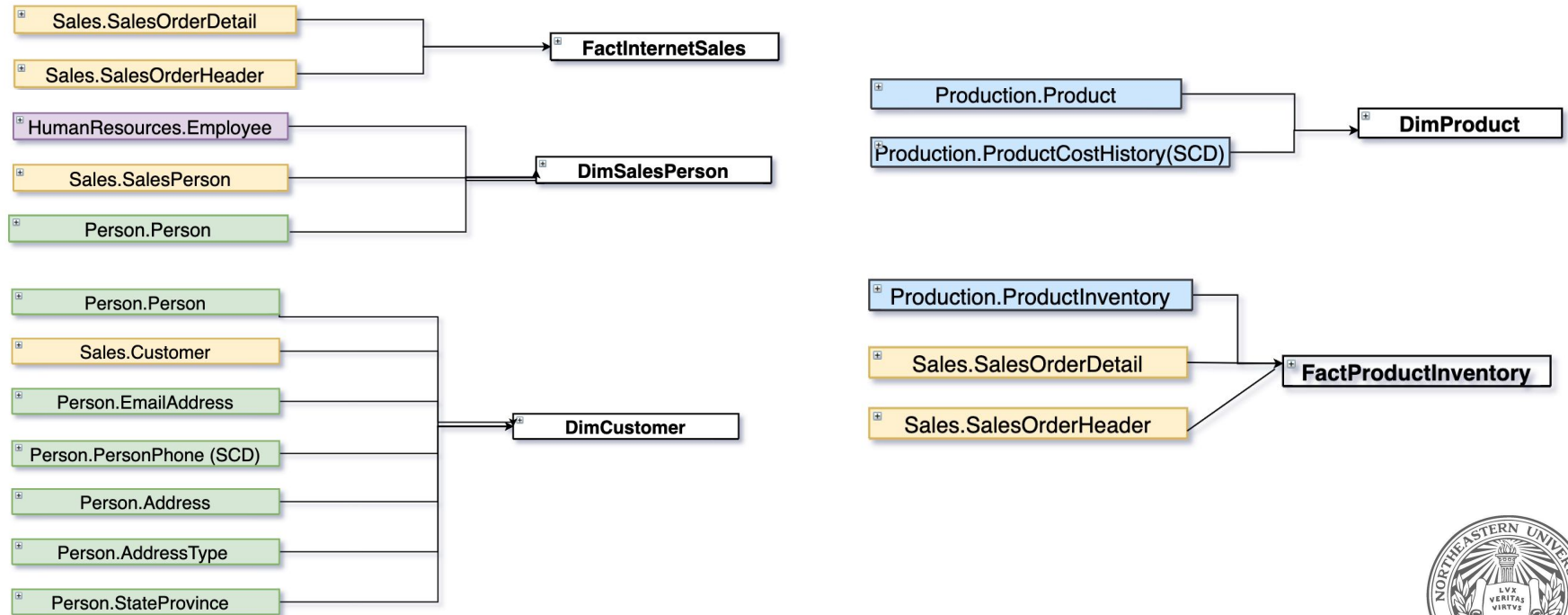
FactProductInventory	
PK FK	<u>ProductKey</u>
PK FK	<u>SellStartDate</u>
UnitCost	
UnitBalance	
UnitIn	
UnitOut	



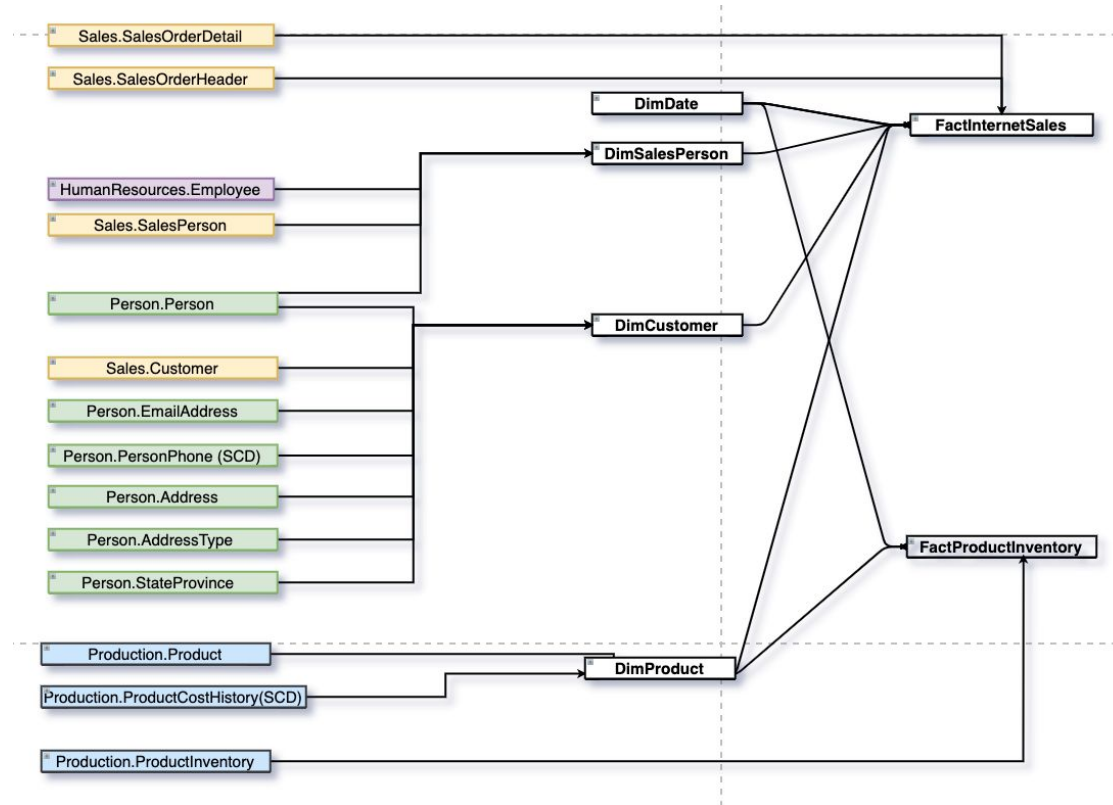
# Data Warehouse Model ---- Facts and Dims



## Data Warehouse Model ---- SOR



# Data Warehouse Model



## Data Integration ---- SCD

- DimCustomer: Customer Phone Number (Overwrite)

CustomerKey	BusinessEntityID	FirstName	LastName	Phone	UpdateDate
11000	11000	Jon	Yang	1(11) 500 555-0162	2011-08-30 11:34:29:403

Replace the original phone number

CustomerKey	BusinessEntityID	FirstName	LastName	Phone	UpdateDate
11000	11000	Jon	Yang	1(11) 500 555-0932	2011-11-04 10:01:24:522

- DimProduct: Product Unit Cost (Full History)

ProductKey	HistoryCostID	ProductUnitCost	CostStartDate	CostEndDate
301	1	33.25	2015-09-15	2016-03-20
301	2	34.75	2016-03-20	2016-05-25
301	3	40.25	2016-05-25	NULL



# Data Integration

SQL Server Integration Services  
(SSIS)





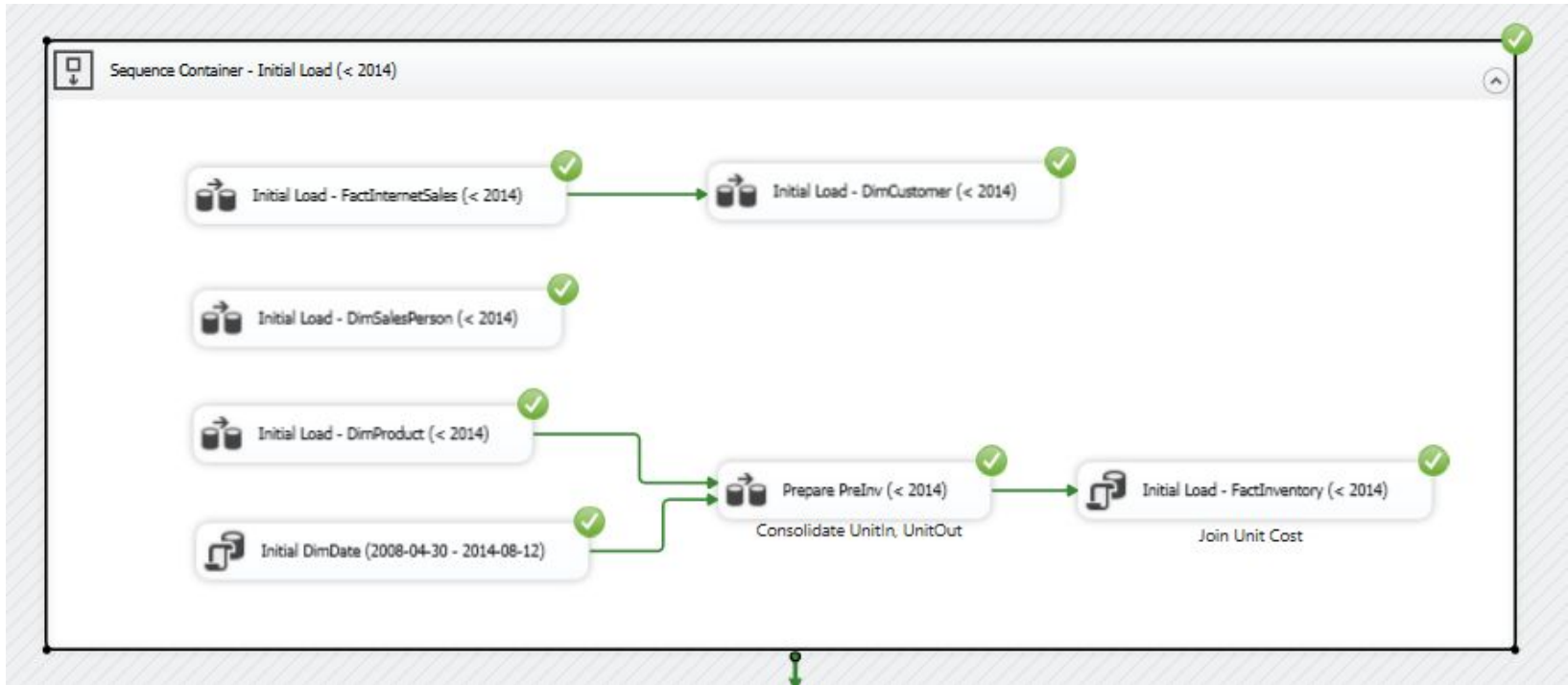
# SSIS

- **Initial Load:** load data before 2014
- **Incremental Loads:** 8 Batches (2014-01-01 to 2014-08-31)
- **SCD:** 2 test cases for customer Phone and product UnitCost respectively



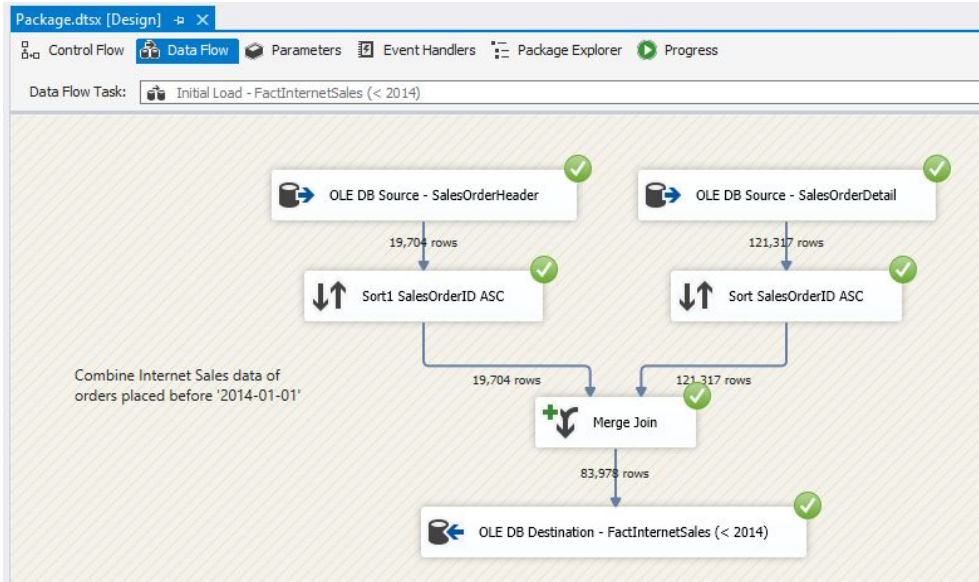
# Initial Load

- Load data prior to 2014

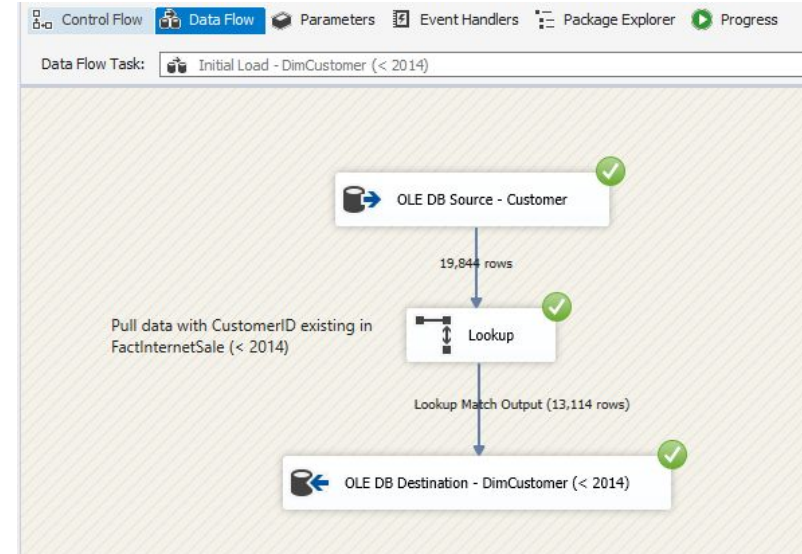


## Initial Load

- Initial Load - FactInternetSales (< 2014)

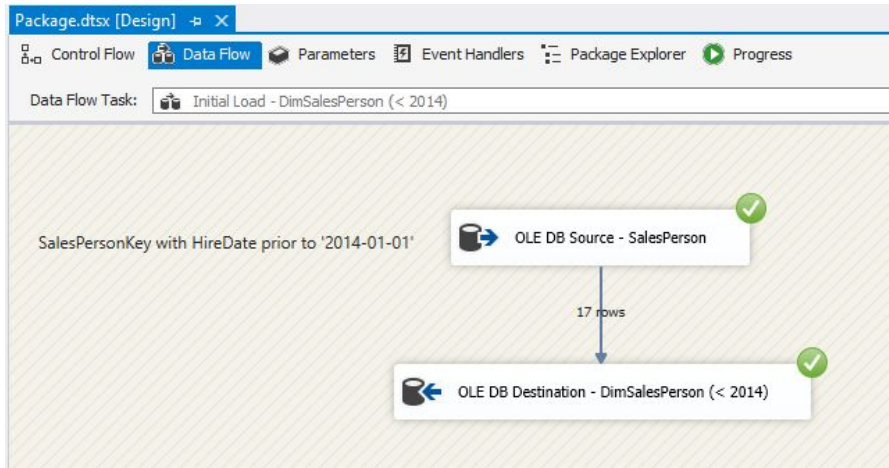


- Initial load - DimCustomer (< 2014)

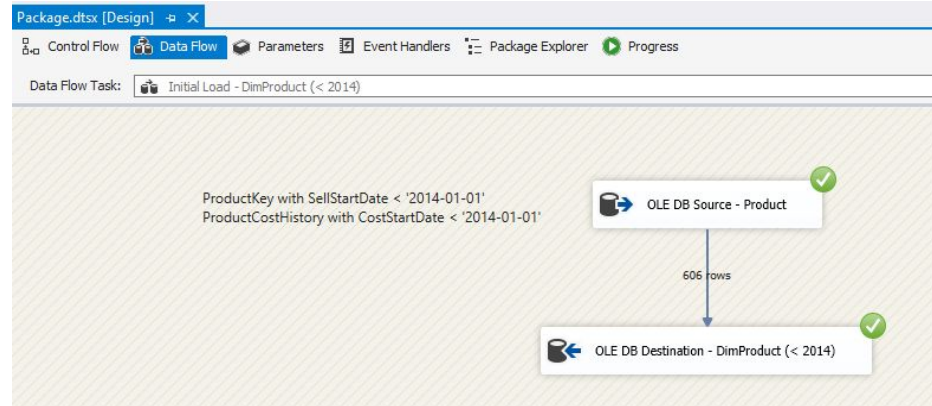


## Initial Load

- Initial Load - DimSalesPerson (< 2014)

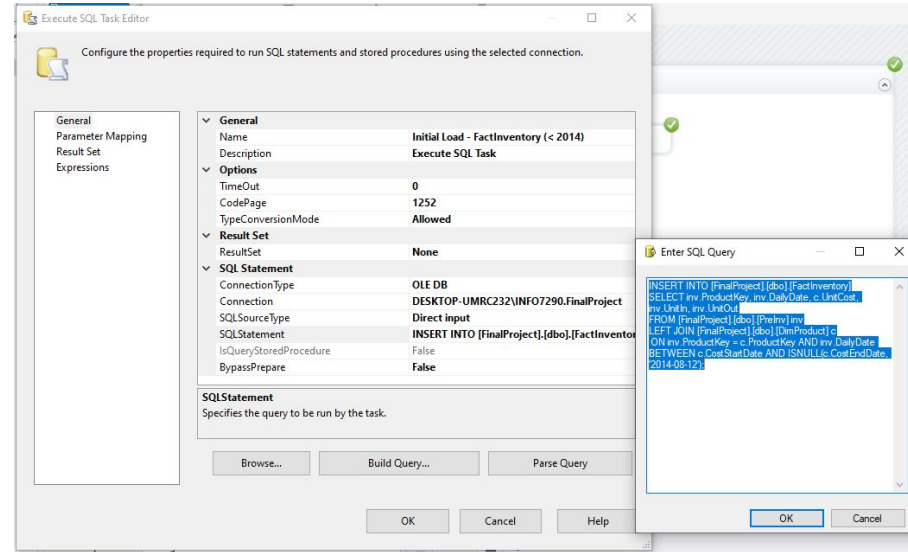
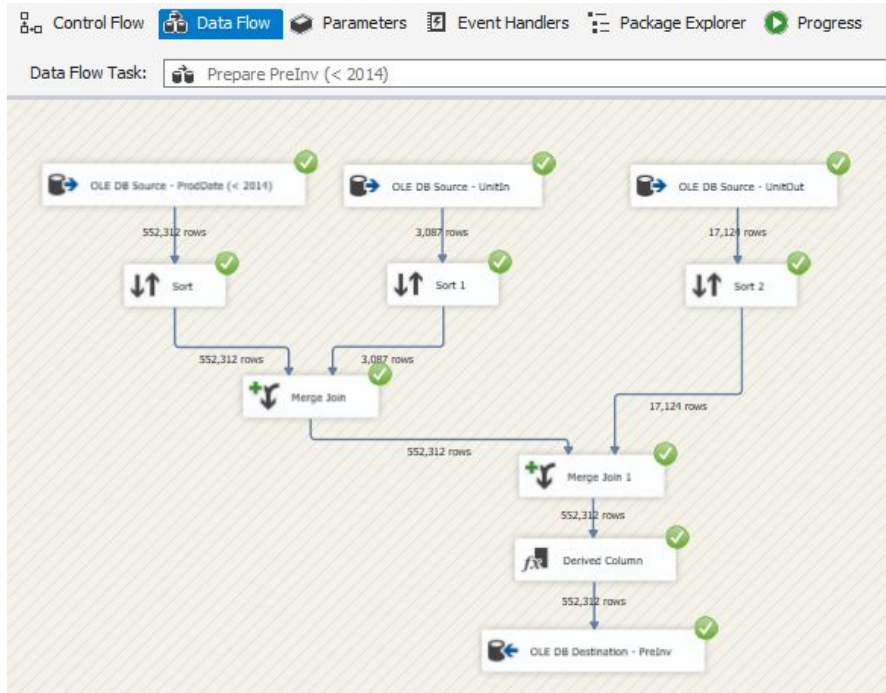


- Initial load - DimProduct (< 2014)

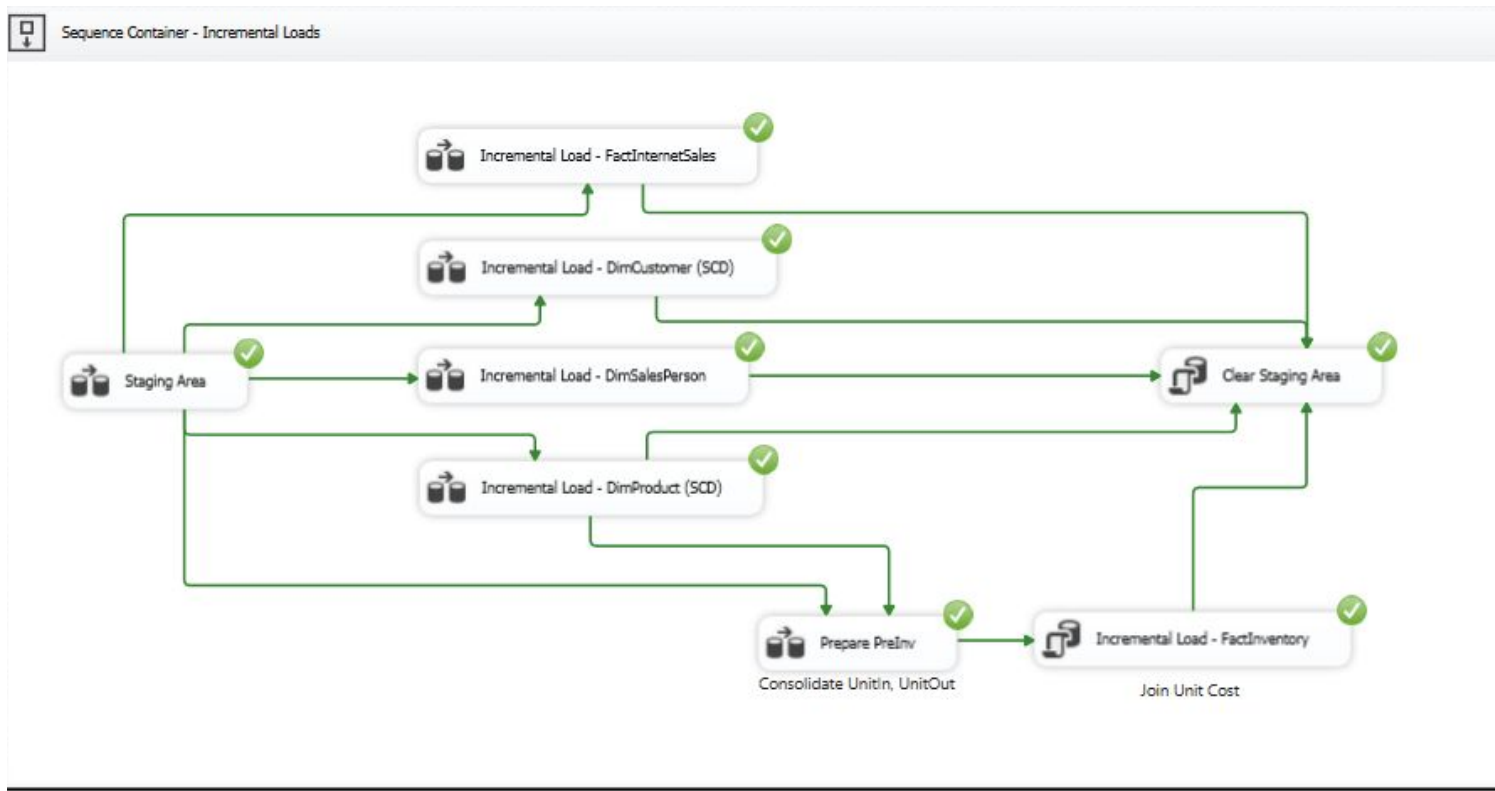


## Initial Load

- Initial Load - FactInventory (< 2014)



# Incremental Loads



# Incremental Loads

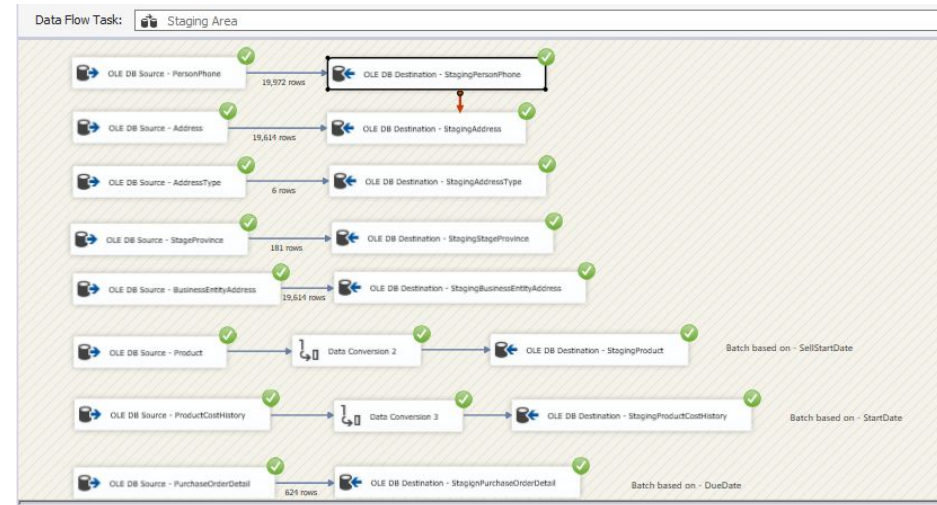
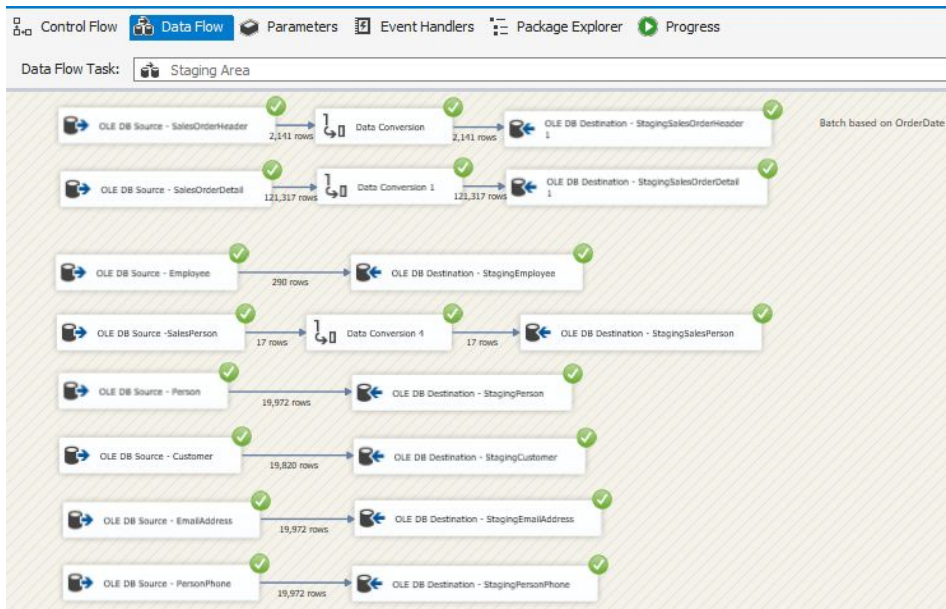
- Based on **month**
- Variables @BatchStartDate and @BatchEndDate to separate batches

Table \ Batch	Initial load (Prior to 2014)	Batch 1 (20140101- 20140131)	Batch 2 (20140201- 20140228)	Batch 3 (20140301- 20140331)	Batch 4 (20140401- 20140430)	Batch 5 (20140501- 20140531)	Batch 6 (20140601- 20140630)	Batch 7 (20140701- 20140731)	Batch 8 (20140801- 20140831)
FactInternetSales	83,978	91,023	95,304	105,259	110,561	119,187	121,317	121,317	121,317
FactInventory	552,312	564,898	576,266	588,852	601,032	613,618	625,798	638,384	638,384
DimCustomer	13,114	19,844	19,844	19,844	19,844	19,844	19,844	19,844	19,844
DimSalesPerson	17	17	17	17	17	17	17	17	17
DimProduct	606	606	606	606	606	606	606	606	606
DimDate	2,296	2,296	2,296	2,296	2,296	2,296	2,296	2,296	2,296





## Incremental Loads - Staging Area

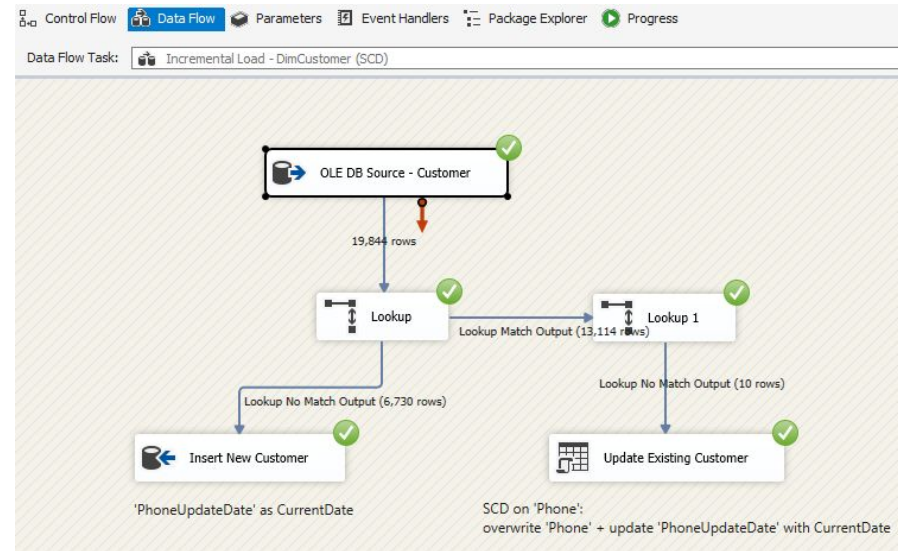
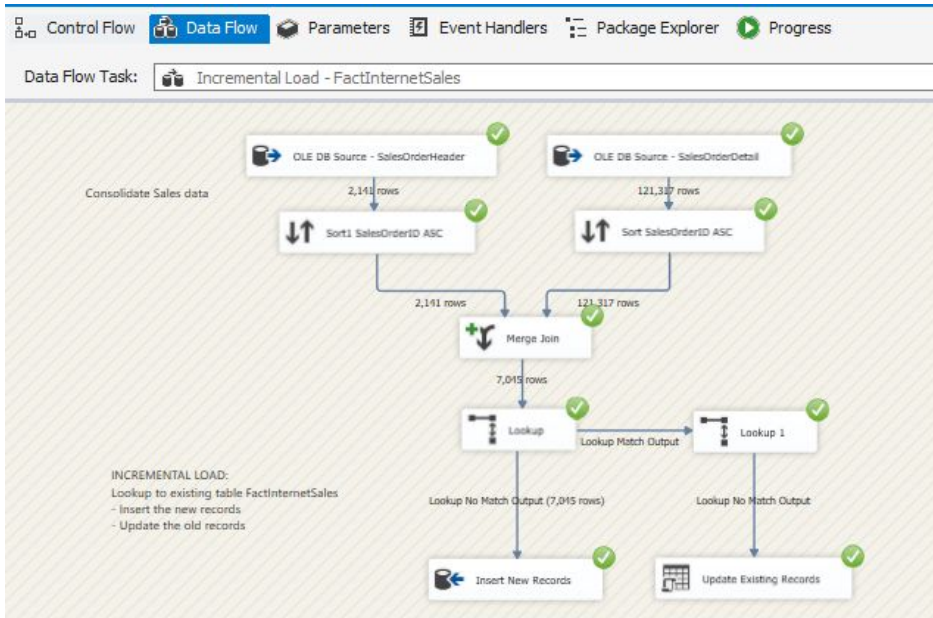




## Incremental Loads - ETL Processing

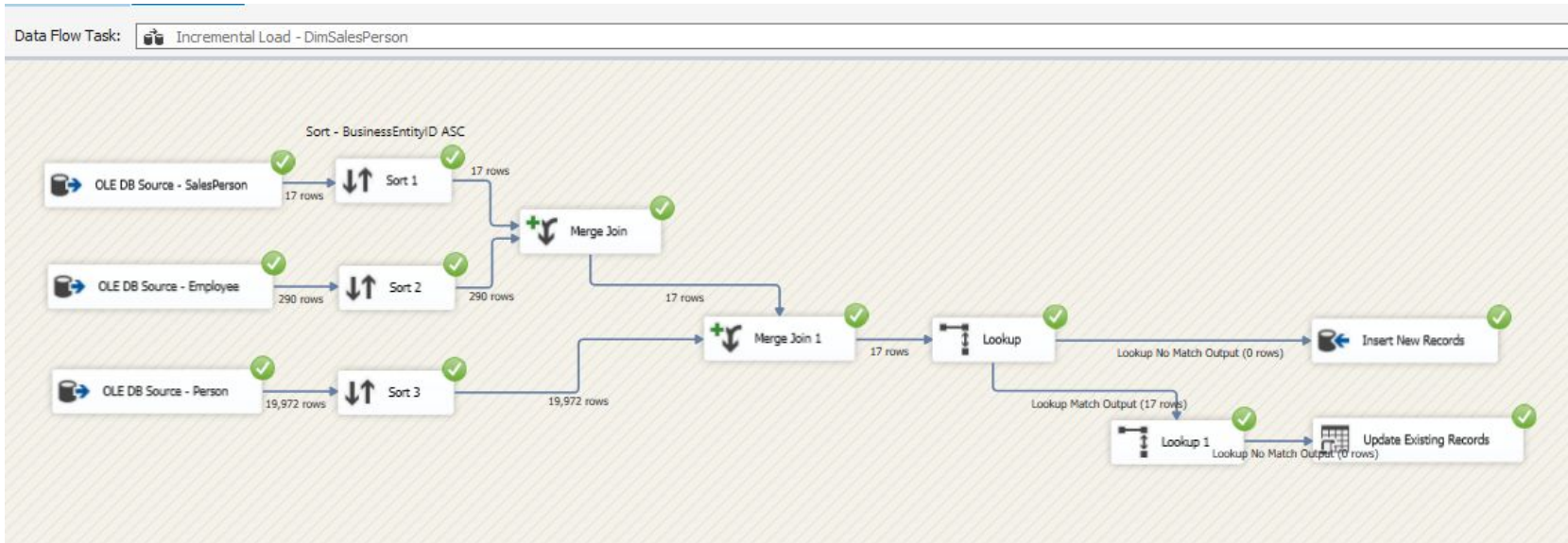
- FactInternetSales

- DimCustomer



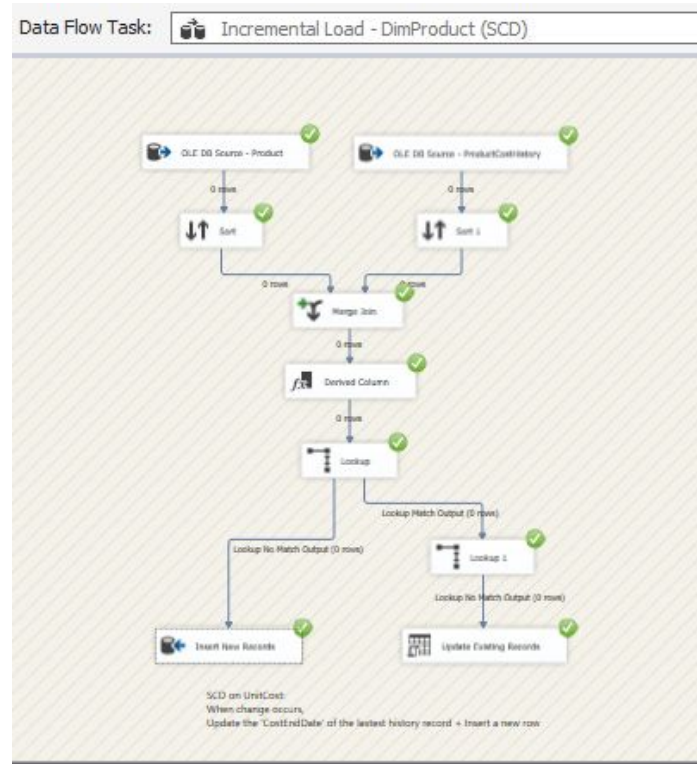
# Incremental Loads - ETL Processing

- DimSalesPerson



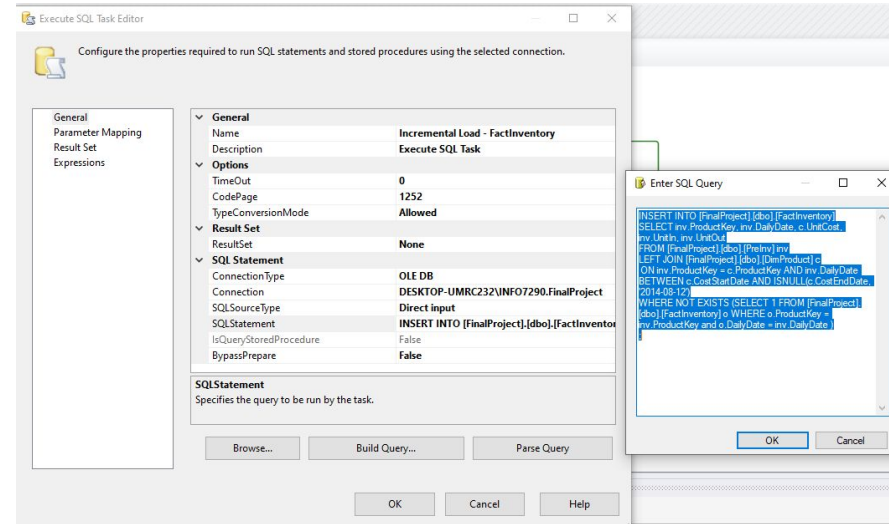
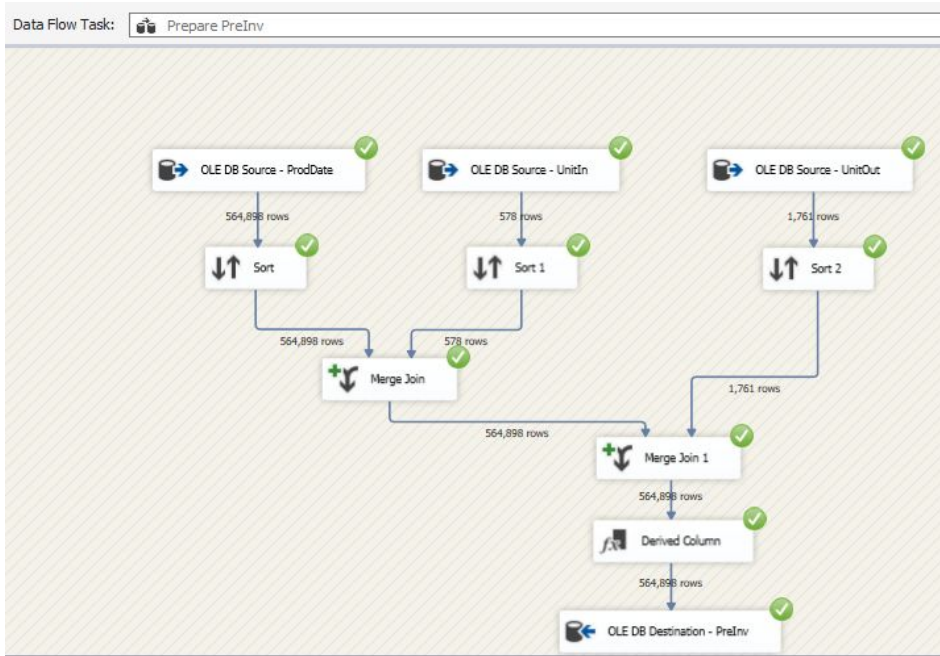
# Incremental Loads - ETL Processing

- DimProduct



## Incremental Loads - ETL Processing

- FactProductInventory



# Incremental Loads - Clear Staging Tables

Execute SQL Task Editor

Configure the properties required to run SQL statements and stored procedures using the selected connection.

- General
- Parameter Mapping
- Result Set
- Expressions

**General**

Name: Clear Staging Area

Description: Execute SQL Task

**Options**

TimeOut: 0

CodePage: 1252

TypeConversionMode: Allowed

**Result Set**

ResultSet: None

**SQL Statement**

ConnectionType: OLE DB

Connection: DESKTOP-UMRC232\INFO7290.FinalProject

SQLSourceType: Direct input

SQLStatement: Truncate Table [FinalProject].[dbo].[Staging\_Sal

IsQueryStoredProcedure: False

BypassPrepare: True

**SQLStatement**

Specifies the query to be run by the task.

Browse... Build Query... Parse Query

Enter SQL Query

```
Truncate Table [FinalProject].[dbo].[Staging_SalesOrderHeader]
Truncate Table [FinalProject].[dbo].[Staging_SalesOrderDetail]
Truncate Table [FinalProject].[dbo].[Staging_Employee]
Truncate Table [FinalProject].[dbo].[Staging_Person]
Truncate Table [FinalProject].[dbo].[Staging_SalesPerson]
Truncate Table [FinalProject].[dbo].[Staging_Customer]
Truncate Table [FinalProject].[dbo].[Staging_EmailAddress]
Truncate Table [FinalProject].[dbo].[Staging_PersonPhone]
Truncate Table [FinalProject].[dbo].[Staging_Address]
Truncate Table [FinalProject].[dbo].[Staging_AddressType]
Truncate Table [FinalProject].[dbo].[Staging_StateProvince]
Truncate Table [FinalProject].[dbo].[Staging_BusinessEntityAddress]
Truncate Table [FinalProject].[dbo].[Staging_Product]
Truncate Table [FinalProject].[dbo].[Staging_ProductCostHistory]
Truncate Table [FinalProject].[dbo].[Staging_PurchaseOrderDetail]
TRUNCATE table [FinalProject].[dbo].[Pre]iv
```

OK Cancel





# SCD1 - DimCustomer: Customer Phone Number (Overwrite)

*Staging\_PersonPhone***Before**

	BusinessEntityID	PhoneNumber	ModifiedDate
1	16722	855-555-0159	2013-11-01 00:00:00.000
2	16723	1 (11) 500 555-0161	2014-01-25 00:00:00.000

```
UPDATE [FinalProject].[dbo].[Staging_PersonPhone]
SET PhoneNumber = '99999', ModifiedDate = GETDATE()
WHERE BusinessEntityID = 16722 OR BusinessEntityID = 16723
```

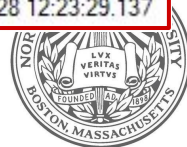
**After**

	BusinessEntityID	PhoneNumber	ModifiedDate
1	16722	99999	2021-04-28 12:23:29.137
2	16723	99999	2021-04-28 12:23:29.137

*DimCustomer*

	CustomerKey	BusinessEntityID	Phone	PhoneUpdateDate
1	28360	16722	855-555-0159	2013-11-01 00:00:00.000
2	17835	16723	1 (11) 500 555-0161	2014-01-25 00:00:00.000

	CustomerKey	BusinessEntityID	Phone	PhoneUpdateDate
1	28360	16722	99999	2021-04-28 12:23:29.137
2	17835	16723	99999	2021-04-28 12:23:29.137



## SCD1 - DimProduct: Product Unit Cost (Full History)

Before

*Staging\_ProductCostHistory*

	ProductID	StartDate	EndDate	StandardCost
1	707	2011-05-31	2012-05-29	12.0278
2	707	2012-05-30	2013-05-29	13.8782
3	707	2013-05-30	NULL	13.0863

*DimProduct*

	ProductKey	HistoryCostID	ProductName	UnitCost	CostStartDate	CostEndDate
1	707	1	Sport-100 Helmet, Red	12.0278	2011-05-31	2012-05-29
2	707	2	Sport-100 Helmet, Red	13.8782	2012-05-30	2013-05-29
3	707	3	Sport-100 Helmet, Red	13.0863	2013-05-30	NULL

```
UPDATE [FinalProject].[dbo].[Staging_ProductCostHistory]
SET EndDate = '9999-01-01'
WHERE ProductID = 707 AND StartDate = '2013-05-30';

INSERT INTO [FinalProject].[dbo].[Staging_ProductCostHistory]
VALUES ('707', '9999-01-01', Null, 99999)
```

After

	ProductID	StartDate	EndDate	StandardCost
1	707	2011-05-31	2012-05-29	12.0278
2	707	2012-05-30	2013-05-29	13.8782
3	707	2013-05-30	9999-01-01	13.0863
4	707	9999-01-01	NULL	99999.0000

	ProductKey	HistoryCostID	ProductName	UnitCost	CostStartDate	CostEndDate
1	707	1	Sport-100 Helmet, Red	12.0278	2011-05-31	2012-05-29
2	707	2	Sport-100 Helmet, Red	13.8782	2012-05-30	2013-05-29
3	707	3	Sport-100 Helmet, Red	13.0863	2013-05-30	9999-01-01
4	707	4	Sport-100 Helmet, Red	99999.0000	9999-01-01	NULL

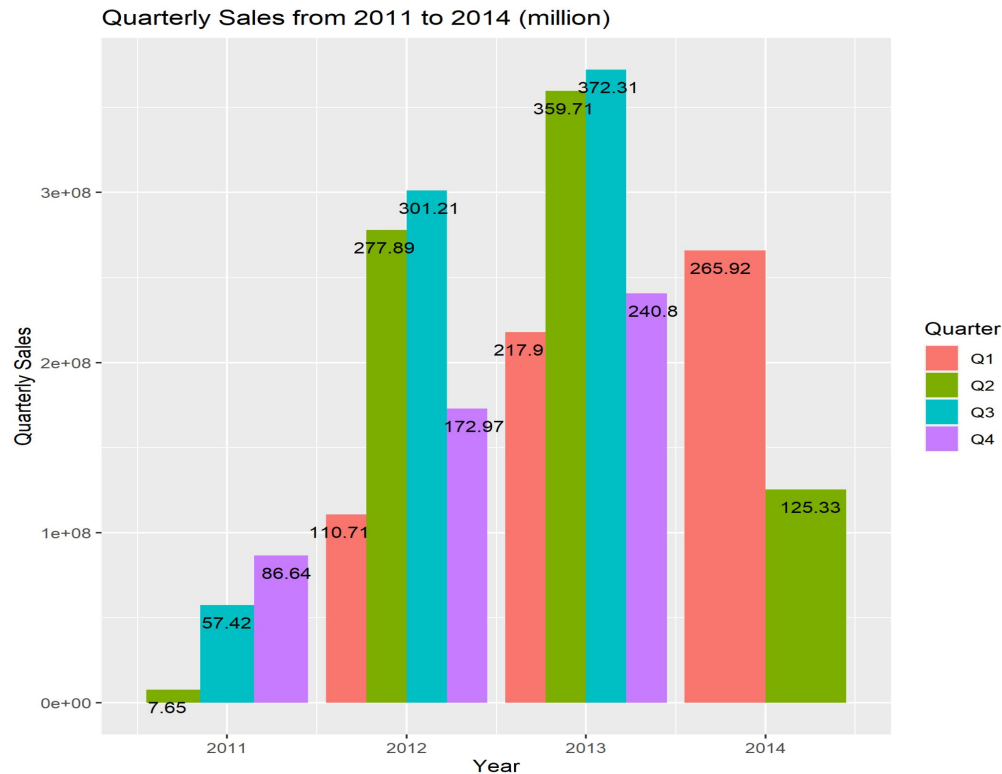
# Data Visualization

## Online Sales Analysis

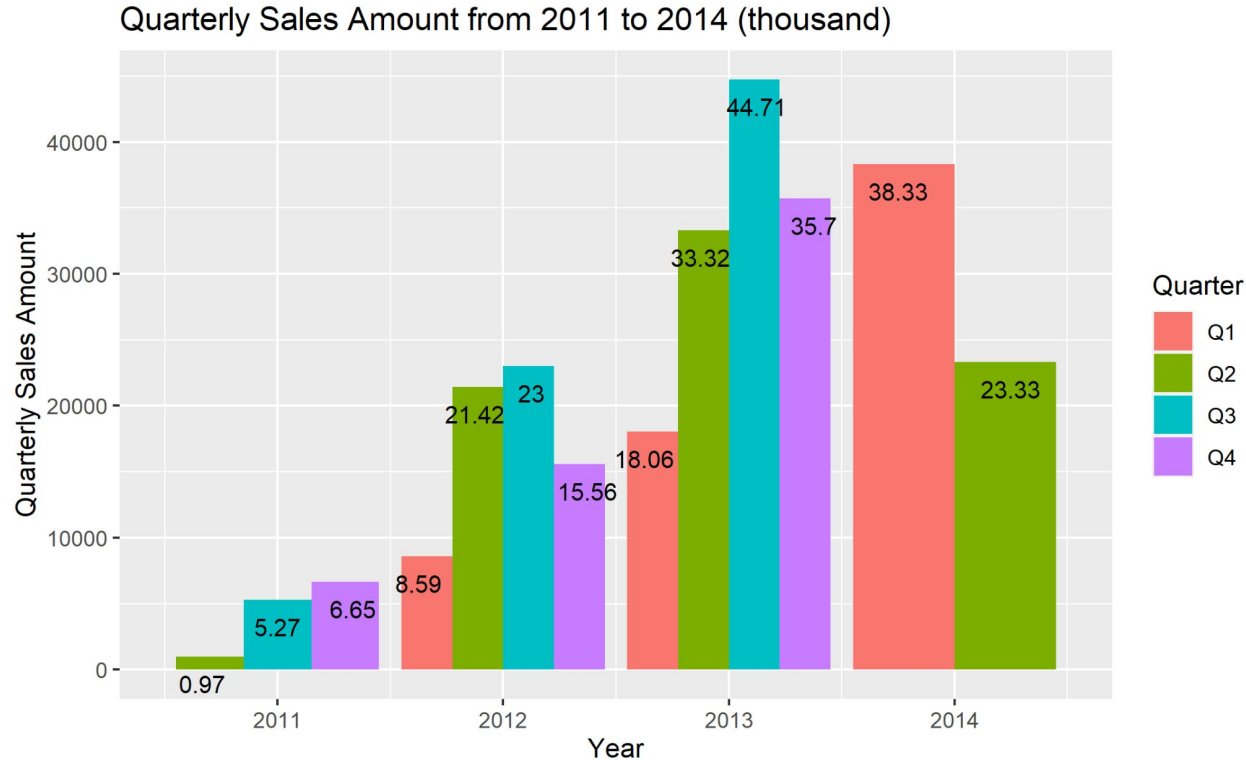




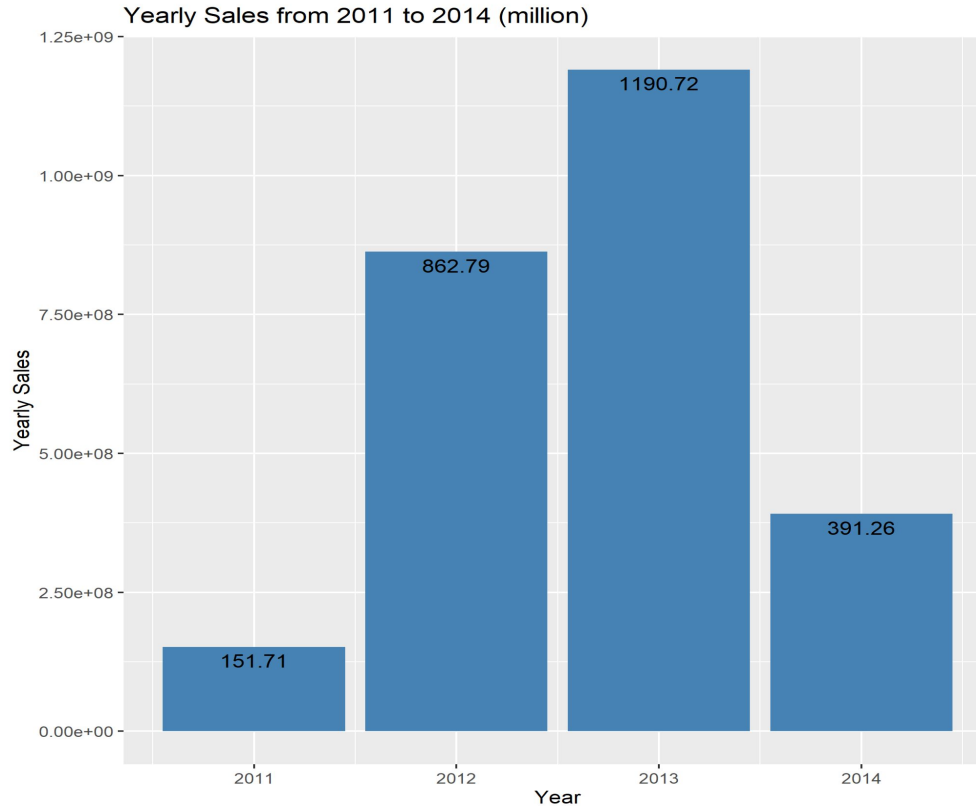
# Data Visualization --- Quarterly Sales



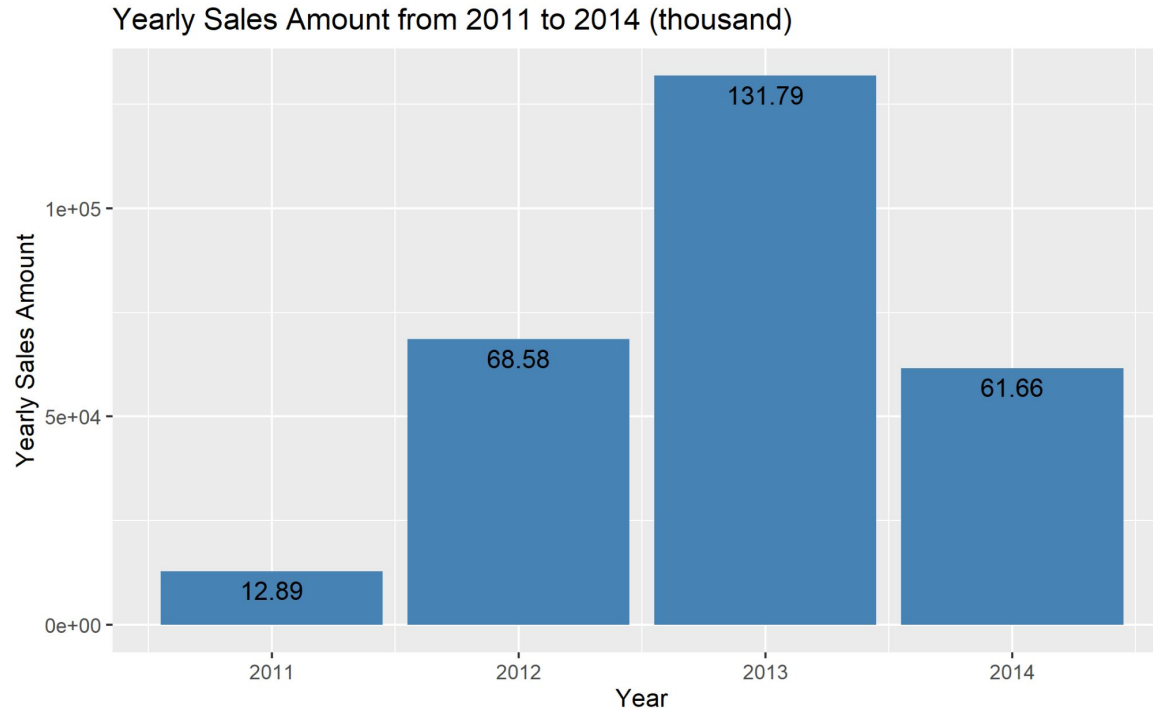
# Data Visualization --- Quarterly Sales Amount



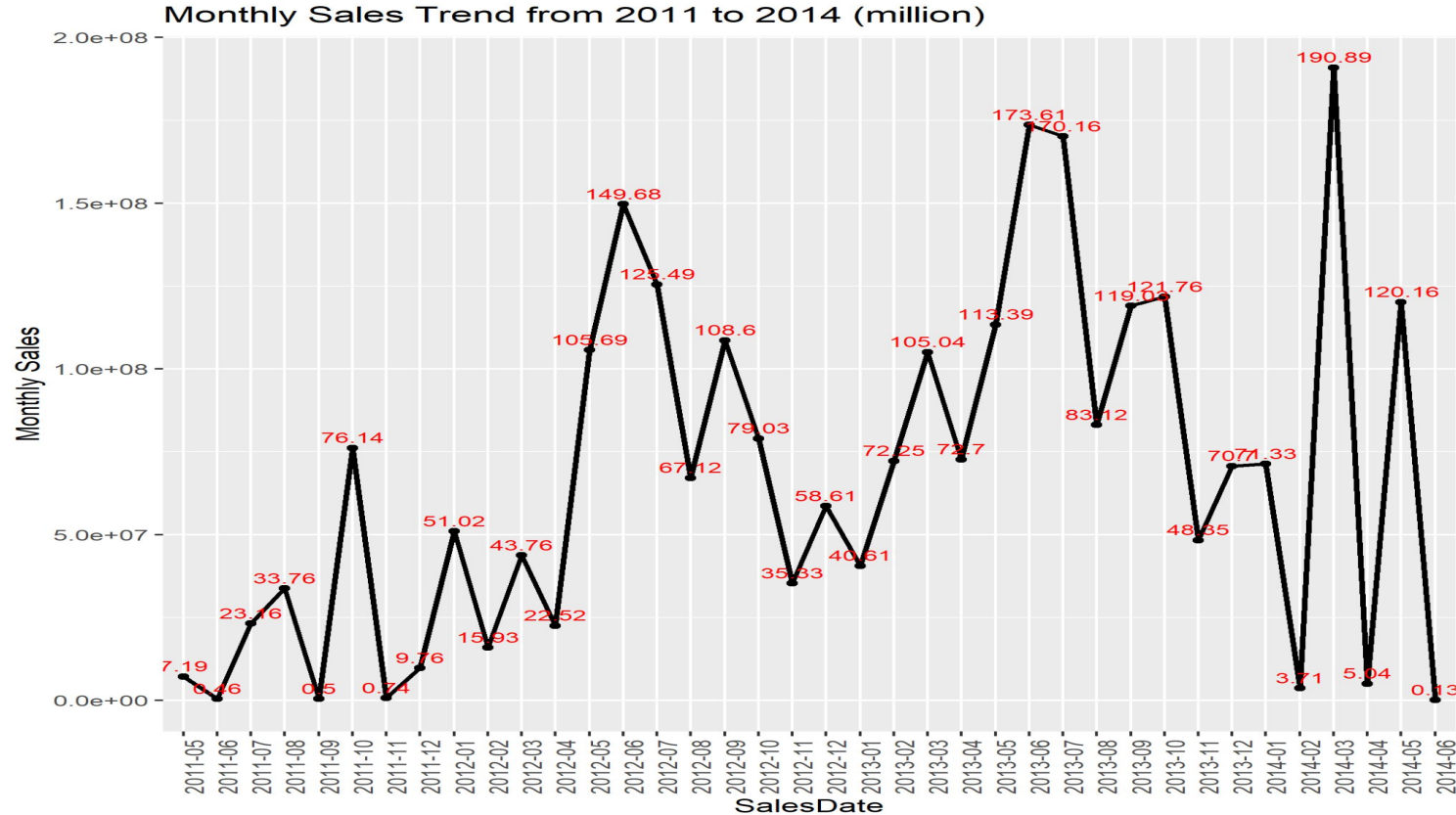
# Data Visualization --- Yearly Sales



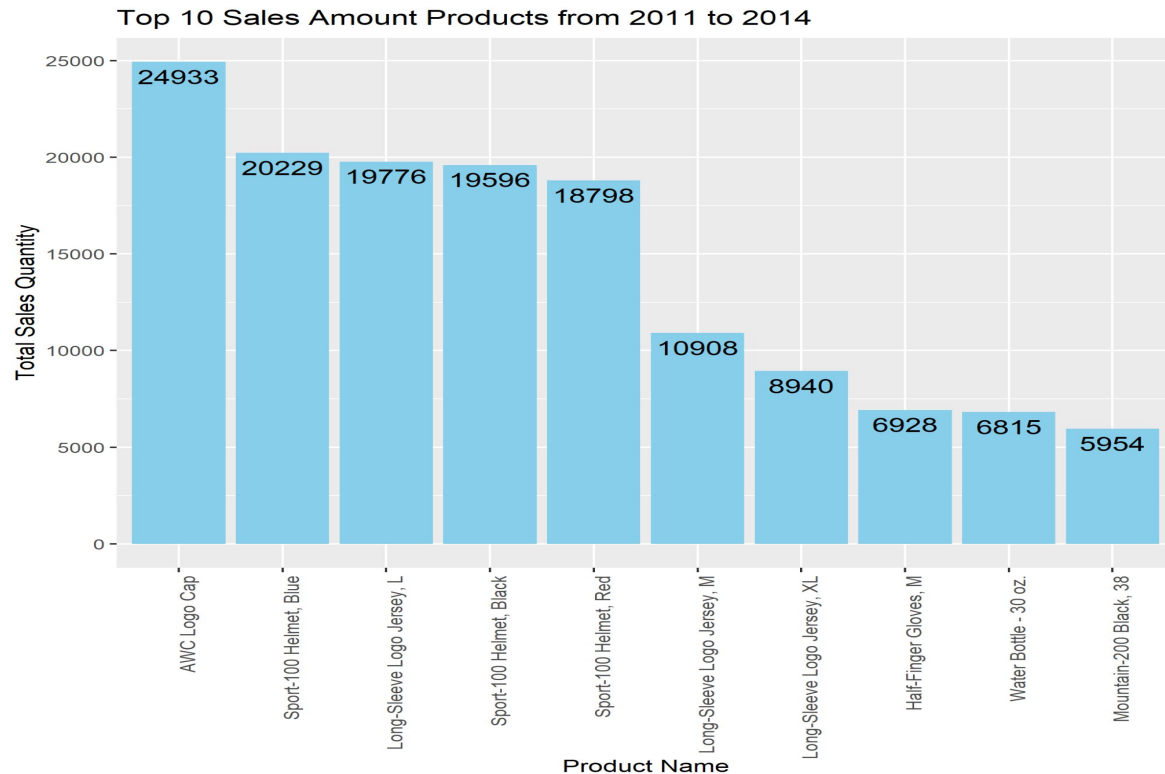
# Data Visualization --- Yearly Sales Amount



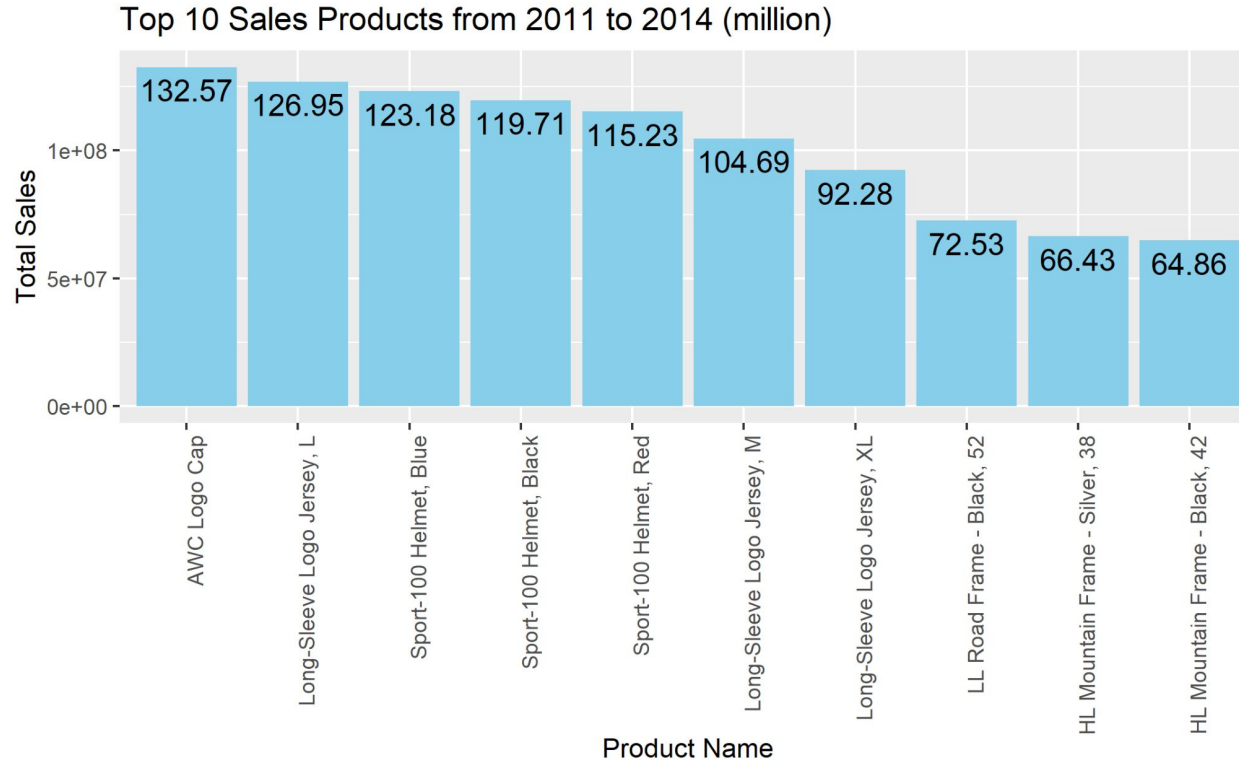
## Data Visualization --- Sales Trends



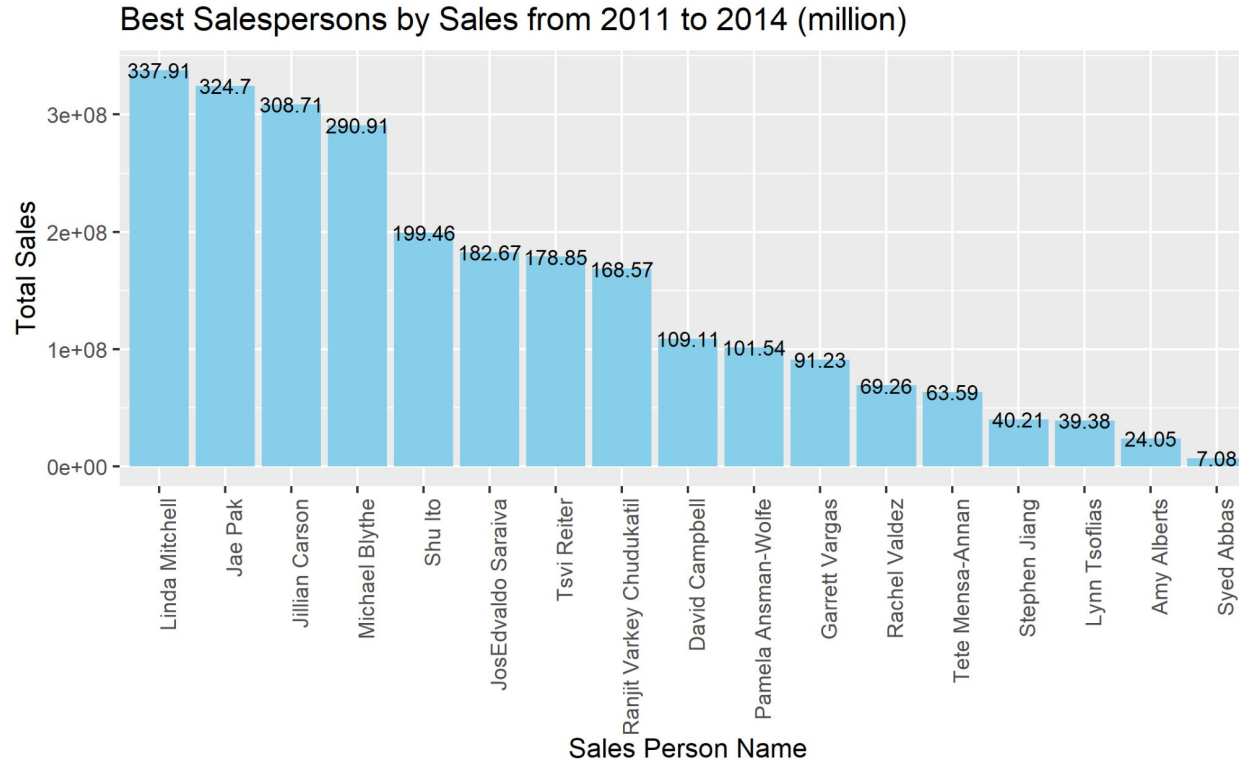
# Data Visualization --- Most Popular Products



## Data Visualization --- Most Popular Products

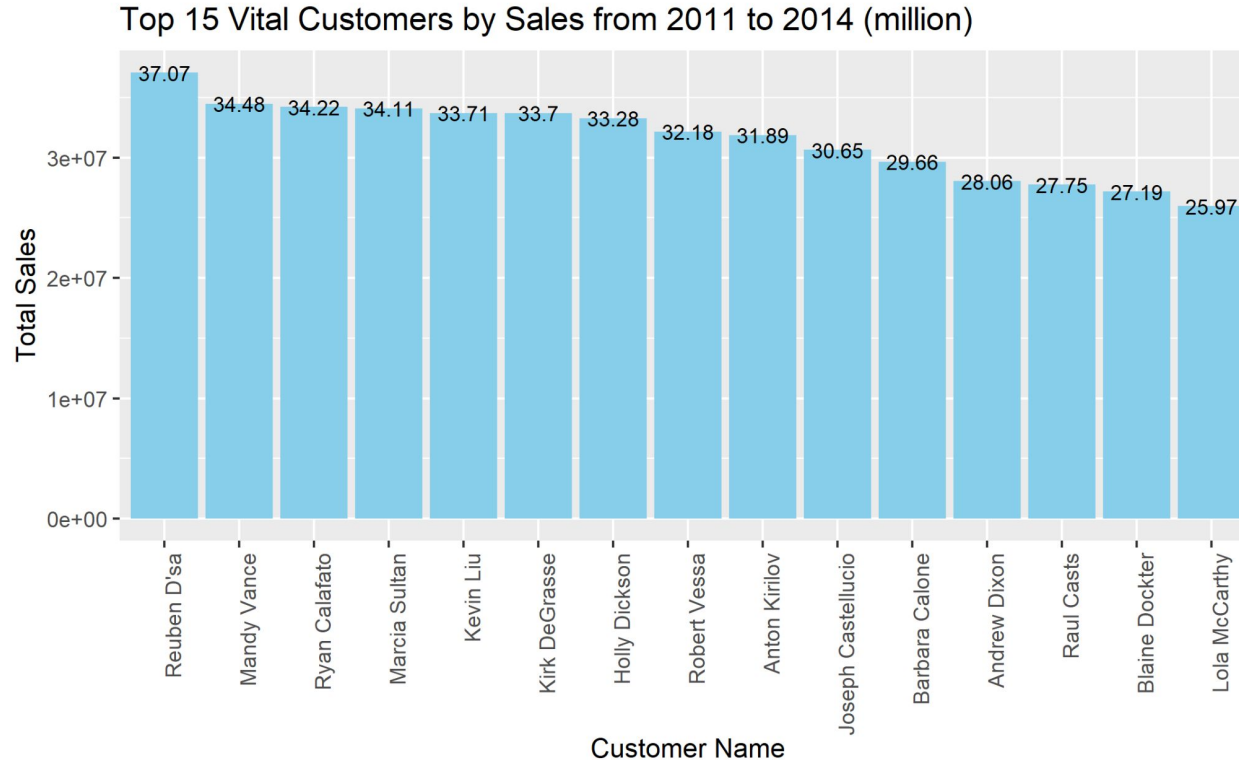


# Data Visualization --- Best Salespersons

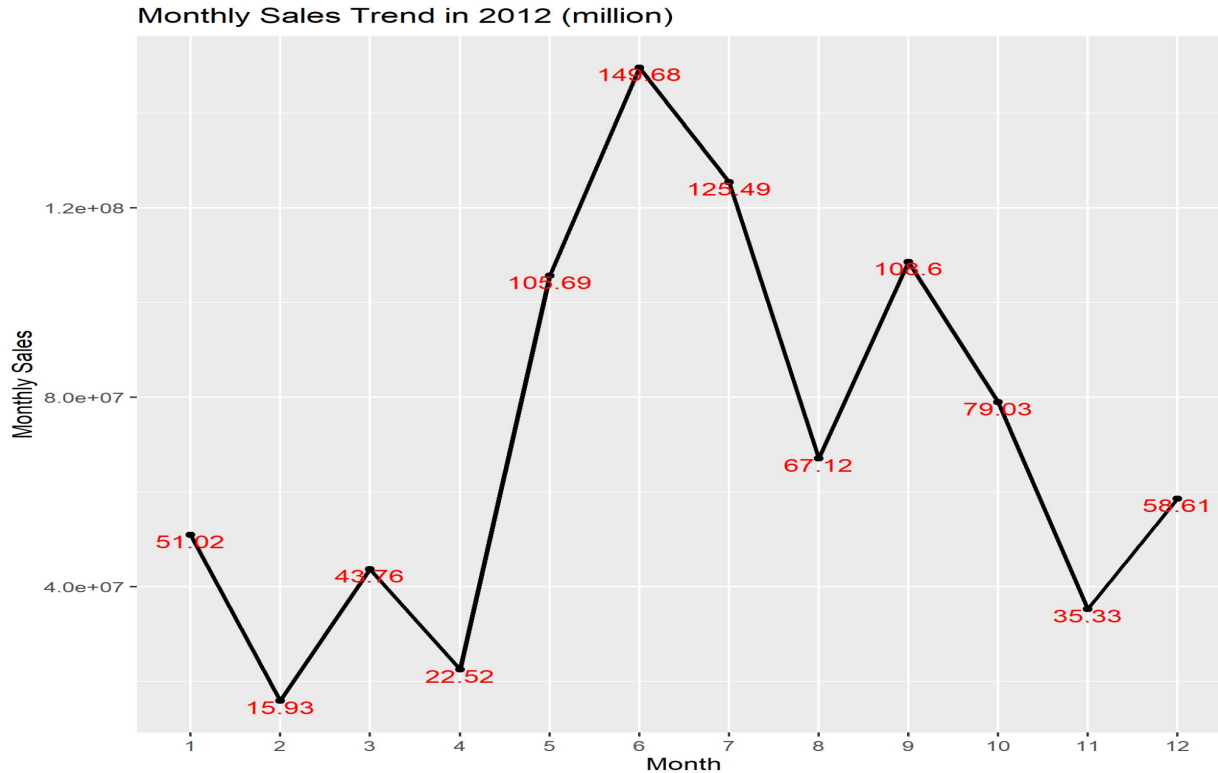




# Data Visualization --- Vital Customers



# Data Visualization --- Sales Trend in 2012



# Data Visualization --- Sales Trend in 2013

