Scenario

X-Mart is having different malls in our city, where daily sales take place for various products. Higher management is facing an issue while decision making due to non availability of integrated data they can’t do study on their data as per their requirement. So they asked us to design a system which can help them quickly in decision making and provide Return on Investment (ROI).

Let us start designing of data warehouse, we need to follow a few steps before we start our data warehouse design.

Developing a Data Warehouse

The phases of a data warehouse project listed below are similar to those of most database projects, starting with identifying requirements and ending with executing the T-SQL Script to create data warehouse:

1. Identify and collect requirements
2. Design the dimensional model
3. Execute T-SQL queries to create and populate your dimension and fact tables

Identify and Collect Requirements

We need to interview the key decision makers to know, what factors define the success in the business? How does management want to analyze their data? What are the most important business questions, which need to be satisfied by this new system?

We also need to work with persons in different departments to know the data and their common relations if any, document their entire requirement which need to be satisfied by this system.

Let us first identify the requirement from management about their requirements.

1. Need to see daily, weekly, monthly, quarterly profit of each store.
2. Comparison of sales and profit on various time periods.
3. Comparison of sales in various time bands of the day.
4. Need to know which product has more demand on which location?
5. Need to study trend of sales by time period of the day over the week, month, and year?
6. On what day sales is higher?
7. On every Sunday of this month, what is sales and what is profit?
8. What is trend of sales on weekday and weekend?
9. Need to compare weekly, monthly and yearly sales to know growth and KPI?

Design the Dimensional Model

We need to design Dimensional Model to suit requirements of users which must address business needs and contains information which can be easily accessible. Design of model should be easily extensible according to future needs. This model design must supports OLAP cubes to provide "instantaneous" query results for analysts.

Let us take a quick look at a few new terms and then we will identify/derive it for our requirement.

**Dimension**

The dimension is a master table composed of individual, non-overlapping [data elements](http://en.wikipedia.org/wiki/Data_element). The primary functions of dimensions are to provide filtering, grouping and labeling on your data. Dimension tables contain textual descriptions about the subjects of the business.

Let me give you a glimpse on different types of dimensions available like confirmed dimension, Role Playing dimension, Degenerated dimension, Junk Dimension.

Slowly changing dimension (SCD) specifies the way using which you are storing values of your dimension which is changing over a time and preserver the history. Different methods / types are available to store history of this change **E.g**. SCD1, SCD2, and SCD3 you can use as per your requirement.

Let us identify dimensions related to the above case study.

Product, Customer, Store, Date, Time, Sales person

**Measure**

A measure represents a column that contains quantifiable data, usually numeric, that can be aggregated. A measure is generally mapped to a column in a fact table. For your information, various types of measures are there. **E.g.** Additive, semi additive and Non additive.

Let us define what will be the Measures in our case.

Actual Cost, Total Sales, Quantity, Fact table record count

**Fact Table**

Data in fact table are called measures (or dependent attributes), Fact table provides statistics for sales broken down by customer, salesperson, product, period and store dimensions. Fact table usually contains historical transactional entries of your live system, it is mainly made up of Foreign key column which references to various dimension and numeric measure values on which aggregation will be performed. Fact tables are of different types, **E.g.** Transactional, Cumulative and Snapshot.

Let us identify what attributes should be there in our Fact Sales Table.

1. **Foreign Key Column**

Sales Date key, Sales Time key, Invoice Number, Sales Person ID, Store ID, Customer ID

1. **Measures**

Actual Cost, Total Sales, Quantity, Fact table record count

Design the Relational Database

We have done some basic workout to identify dimensions and measures, now we have to use appropriate schema to relate this dimension and Fact tables.

Few popular schemas used to develop dimensional model are as follows:

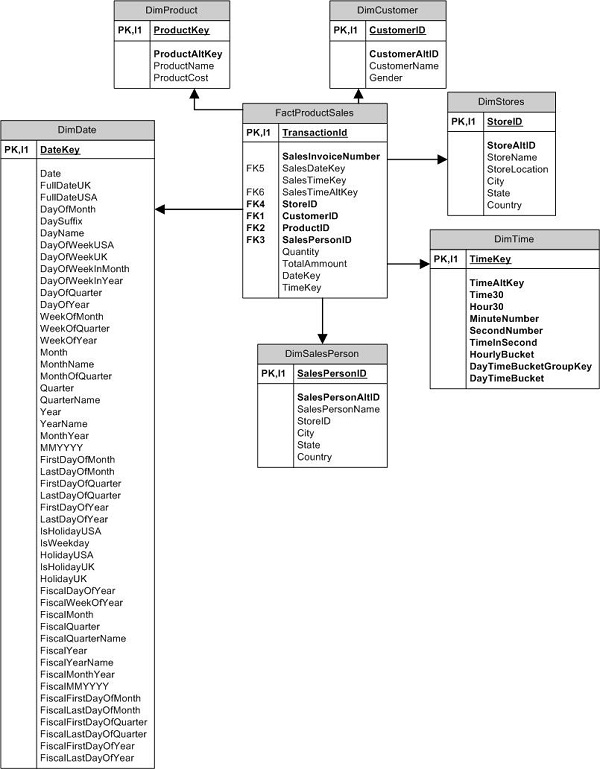
**E.g.** Star Schema, Snow Flake Schema, Star Flake Schema, Distributed Star Schema, etc.

In a different article, we will discuss all these schemas, dimension types, measure types, etc., in detail.

Personally, I will first try to use Star schema due to hierarchical attribute model it provides for analysis and speedy performance in querying the data.

Star schema the diagram resembles a star, with points radiating from a center. The center of the star consists of fact table and the points of the star are the dimension tables.

**Let us create Our First Star Schema, please refer to the below figure:**



Using the Code

Let us execute our T-SQL Script step by step to create table and populate them with appropriate test values.

Follow the given steps to run the query in SSMS (SQL Server Management Studio).

1. Open SQL Server Management Studio
2. Connect Database Engine
3. Open New Query editor
4. Copy paste Scripts given below in various steps in new query editor window one by one
5. To run the given SQL Script, press F5

**Step 1**

Create database for your Data Warehouse in SQL Server:

Createdatabase Sales\_DW

Go

Use Sales\_DW

Go

**Step 2**

Create **Customer dimension** table in Data Warehouse which will hold customer personal details.

Create table DimCustomer

(

CustomerID int primary key identity,

CustomerAltID varchar(10) not null,

CustomerName varchar(50),

Gender varchar(20)

)

go

Fill the **Customer dimension** with sample Values

Insert into DimCustomer(CustomerAltID,CustomerName,Gender)values

('IMI-001','Henry Ford','M'),

('IMI-002','Bill Gates','M'),

('IMI-003','Muskan Shaikh','F'),

('IMI-004','Richard Thrubin','M'),

('IMI-005','Emma Wattson','F');

Go

**Step 3**

Create basic level of **Product Dimension** table without considering any Category or Subcategory

Create table DimProduct

(

ProductKey int primary key identity,

ProductAltKey varchar(10)not null,

ProductName varchar(100),

ProductActualCost money,

ProductSalesCost money

)

Go

Fill the **Product dimension** with sample Values

Insert into DimProduct(ProductAltKey,ProductName, ProductActualCost, ProductSalesCost)values

('ITM-001','Wheat Floor 1kg',5.50,6.50),

('ITM-002','Rice Grains 1kg',22.50,24),

('ITM-003','SunFlower Oil 1 ltr',42,43.5),

('ITM-004','Nirma Soap',18,20),

('ITM-005','Arial Washing Powder 1kg',135,139);

GO

**Step 4**

Create **Store Dimension** table which will hold details related stores available across various places.

Create table DimStores

(

StoreID int primary key identity,

StoreAltID varchar(10)not null,

StoreName varchar(100),

StoreLocation varchar(100),

City varchar(100),

State varchar(100),

Country varchar(100)

)

Go

Fill the **Store Dimension** with sample Values

Insert into DimStores(StoreAltID,StoreName,StoreLocation,City,State,Country )values

('LOC-A1','X-Mart','S.P. RingRoad','Ahmedabad','Guj','India'),

('LOC-A2','X-Mart','Maninagar','Ahmedabad','Guj','India'),

('LOC-A3','X-Mart','Sivranjani','Ahmedabad','Guj','India');

Go

**Step 5**

Create **Dimension Sales Person** table which will hold details related stores available across various places.

Create table DimSalesPerson

(

SalesPersonID int primary key identity,

SalesPersonAltID varchar(10)not null,

SalesPersonName varchar(100),

StoreID int,

City varchar(100),

State varchar(100),

Country varchar(100)

)

Go

Fill the Dimension **Sales Person** with sample values:

Insert into DimSalesPerson(SalesPersonAltID,SalesPersonName,StoreID,City,State,Country )values

('SP-DMSPR1','Ashish',1,'Ahmedabad','Guj','India'),

('SP-DMSPR2','Ketan',1,'Ahmedabad','Guj','India'),

('SP-DMNGR1','Srinivas',2,'Ahmedabad','Guj','India'),

('SP-DMNGR2','Saad',2,'Ahmedabad','Guj','India'),

('SP-DMSVR1','Jasmin',3,'Ahmedabad','Guj','India'),

('SP-DMSVR2','Jacob',3,'Ahmedabad','Guj','India');

Go

**Step 6**

Create **Date Dimension** table which will create and populate date data divided on various levels.

For this, you have to refer my article on CodeProject [Create and Populate Date Dimension](http://www.codeproject.com/Articles/647950/Create-and-Populate-Date-Dimension-for-Data-Wareho).

Download the script and run it in this database for creating and filling of date dimension with values.

**Step 7**

Create **Time Dimension** table which will create and populate Time data for the entire day with various time buckets.

For this, you have to refer to my article on Code Project, [Create & Populate Time Dimension with 24 Hour+ Values](http://www.codeproject.com/Tips/642912/Create-Populate-Time-Dimension-with-24-Hourplus-Va)

Download the script and run it in this database for creating and filling of time dimension with values.

**Step 8**

Create **Fact table** to hold all your transactional entries of previous day sales with appropriate foreign key columns which refer to primary key column of your dimensions; you have to take care while populating your fact table to refer to primary key values of appropriate dimensions.

e.g.

Customer Henry Ford has purchase purchased 2 items (sunflower oil 1 kg, and 2 Nirma soap) in a single invoice on date 1-jan-2013 from D-mart at Sivranjani and sales person was Jacob , billing time recorded is 13:00, so let us define how will we refer to the primary key values from each dimension.

Before filling fact table, you have to identify and do look up for primary key column values in dimensions as per given example and fill in foreign key columns of fact table with appropriate key values.

| **Attribute Name** | **Dimension Table** | **Primary Key Column/Value** |
| --- | --- | --- |
| Date (1-jan-2013), Sales Date Key (20130101) | Dim Date | Date Key: 20130101 |
| Time (13:00:00) Sales Time Alt Key (130000) | Dim Time | Time Key: 46800 |
| Composite key (Sales Person Alt ID+ Name ) for ('SP-DMSVR1'+’Jacob’) | Dim Sales Person | Sales Person ID: 6 |
| Product Alt Key of (Sunflower Oil 1kg)'ITM-003' | Dim Product | Product ID: 3 |
| Product Alt Key (Nirma Soap) 'ITM-004' | Dim Product | Product ID: 4 |
| Store Alt ID of (Sivranjani store) 'LOC-A3' | Dim Store | Store ID: 3 |
| Customer Alt ID of (Henry Ford) is 'IMI-001' | Dim Customer | Customer ID: 1 |

Create Table FactProductSales

(

TransactionId bigint primary key identity,

SalesInvoiceNumber int not null,

SalesDateKey int,

SalesTimeKey int,

SalesTimeAltKey int,

StoreID int not null,

CustomerID int not null,

ProductID int not null,

SalesPersonID int not null,

Quantity float,

SalesTotalCost money,

ProductActualCost money,

Deviation float

)

Go

Add Relation between Fact table and dimension tables:

*-- Add relation between fact table foreign keys to Primary keys of Dimensions*

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_StoreID FOREIGN KEY (StoreID)REFERENCES DimStores(StoreID);

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_CustomerID FOREIGN KEY (CustomerID)REFERENCES Dimcustomer(CustomerID);

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_ProductKey FOREIGN KEY (ProductID)REFERENCES Dimproduct(ProductKey);

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_SalesPersonID FOREIGN KEY (SalesPersonID)REFERENCES Dimsalesperson(SalesPersonID);

Go

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_SalesDateKey FOREIGN KEY (SalesDateKey)REFERENCES DimDate(DateKey);

Go

AlTER TABLE FactProductSales ADD CONSTRAINT \_

FK\_SalesTimeKey FOREIGN KEY (SalesTimeKey)REFERENCES DimDate(TimeKey);

Go

Populate your **Fact table** with historical transaction values of sales for previous day, with proper values of dimension key values.

Insert into FactProductSales(SalesInvoiceNumber,SalesDateKey,\_

SalesTimeKey,SalesTimeAltKey,StoreID,CustomerID,ProductID ,\_

SalesPersonID,Quantity,ProductActualCost,SalesTotalCost,Deviation)values

*--1-jan-2013*

*--SalesInvoiceNumber,SalesDateKey,SalesTimeKey,SalesTimeAltKey,\_*

StoreID,CustomerID,ProductID ,SalesPersonID,Quantity,\_

ProductActualCost,SalesTotalCost,Deviation)

(1,20130101,44347,121907,1,1,1,1,2,11,13,2),

(1,20130101,44347,121907,1,1,2,1,1,22.50,24,1.5),

(1,20130101,44347,121907,1,1,3,1,1,42,43.5,1.5),

(2,20130101,44519,122159,1,2,3,1,1,42,43.5,1.5),

(2,20130101,44519,122159,1,2,4,1,3,54,60,6),

(3,20130101,52415,143335,1,3,2,2,2,11,13,2),

(3,20130101,52415,143335,1,3,3,2,1,42,43.5,1.5),

(3,20130101,52415,143335,1,3,4,2,3,54,60,6),

(3,20130101,52415,143335,1,3,5,2,1,135,139,4),

*--2-jan-2013*

*--SalesInvoiceNumber,SalesDateKey,SalesTimeKey,SalesTimeAltKey,\_*

StoreID,CustomerID,ProductID ,SalesPersonID,Quantity,ProductActualCost,SalesTotalCost,Deviation)

(4,20130102,44347,121907,1,1,1,1,2,11,13,2),

(4,20130102,44347,121907,1,1,2,1,1,22.50,24,1.5),

(5,20130102,44519,122159,1,2,3,1,1,42,43.5,1.5),

(5,20130102,44519,122159,1,2,4,1,3,54,60,6),

(6,20130102,52415,143335,1,3,2,2,2,11,13,2),

(6,20130102,52415,143335,1,3,5,2,1,135,139,4),

(7,20130102,44347,121907,2,1,4,3,3,54,60,6),

(7,20130102,44347,121907,2,1,5,3,1,135,139,4),

*--3-jan-2013*

*--SalesInvoiceNumber,SalesDateKey,SalesTimeKey,SalesTimeAltKey,StoreID,\_*

CustomerID,ProductID ,SalesPersonID,Quantity,ProductActualCost,SalesTotalCost,Deviation)

(8,20130103,59326,162846,1,1,3,1,2,84,87,3),

(8,20130103,59326,162846,1,1,4,1,3,54,60,3),

(9,20130103,59349,162909,1,2,1,1,1,5.5,6.5,1),

(9,20130103,59349,162909,1,2,2,1,1,22.50,24,1.5),

(10,20130103,67390,184310,1,3,1,2,2,11,13,2),

(10,20130103,67390,184310,1,3,4,2,3,54,60,6),

(11,20130103,74877,204757,2,1,2,3,1,5.5,6.5,1),

(11,20130103,74877,204757,2,1,3,3,1,42,43.5,1.5)

Go