

Data Warehousing(Lab)

Engr. Madeha Mushtaq
Department of Computer Science
Iqra National University

Data Warehouse(DWH)

- Data Warehouse is integrated, non volatile, subject oriented and time variant storage of data.
- Whenever data is distributed across various databases, application or at various places stored in different formats, In order to get useful information, we might need DWH, a single unique storage at a single location.

Why Data Warehouse

- DWH is important to take perfect decision on the basis of your historical data, so you can study data using multiple dimensions and can do better analysis to take accurate decision.
- Now a day's users need to have self service BI (Business Intelligence) capabilities so they can create reports on their own (Ad-Hoc reports) and can do analysis of data without much technical knowledge.
- Data warehousing is a business analyst's dream - all the information about the organization's activities gathered in one place, open to a single set of analytical tools.

Designing a DWH

- 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.
- Problem Statement:
 - Management asked us to design a system which can help them quickly in decision making and provide Return on Investment (ROI).

Designing a DWH

- 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 SQL Script to create data warehouse:
 - Identify and collect requirements
 - Design the dimensional model
 - Execute SQL queries to create and populate your dimension and fact tables.

Identify and Collect Requirements

- Let us first identify the requirement from management about their requirements.
- Need to see daily, weekly, monthly, quarterly profit of each store.
- Comparison of sales and profit on various time periods.
- Comparison of sales in various time bands of the day.
- Need to know which product has more demand on which location?
- On every Sunday of this month, what is sales and what is profit?
- What is trend of sales on weekday and weekend?
- 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.
- **Dimensions:**
 - Let us identify dimensions related to the above case study.
 - Product, Customer, Store, Date, Time, Sales person
- **Measures:**
 - 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.
 - Let us define what will be the Measures in our case.
 - Actual Cost, Total Sales, Quantity, Fact table record count

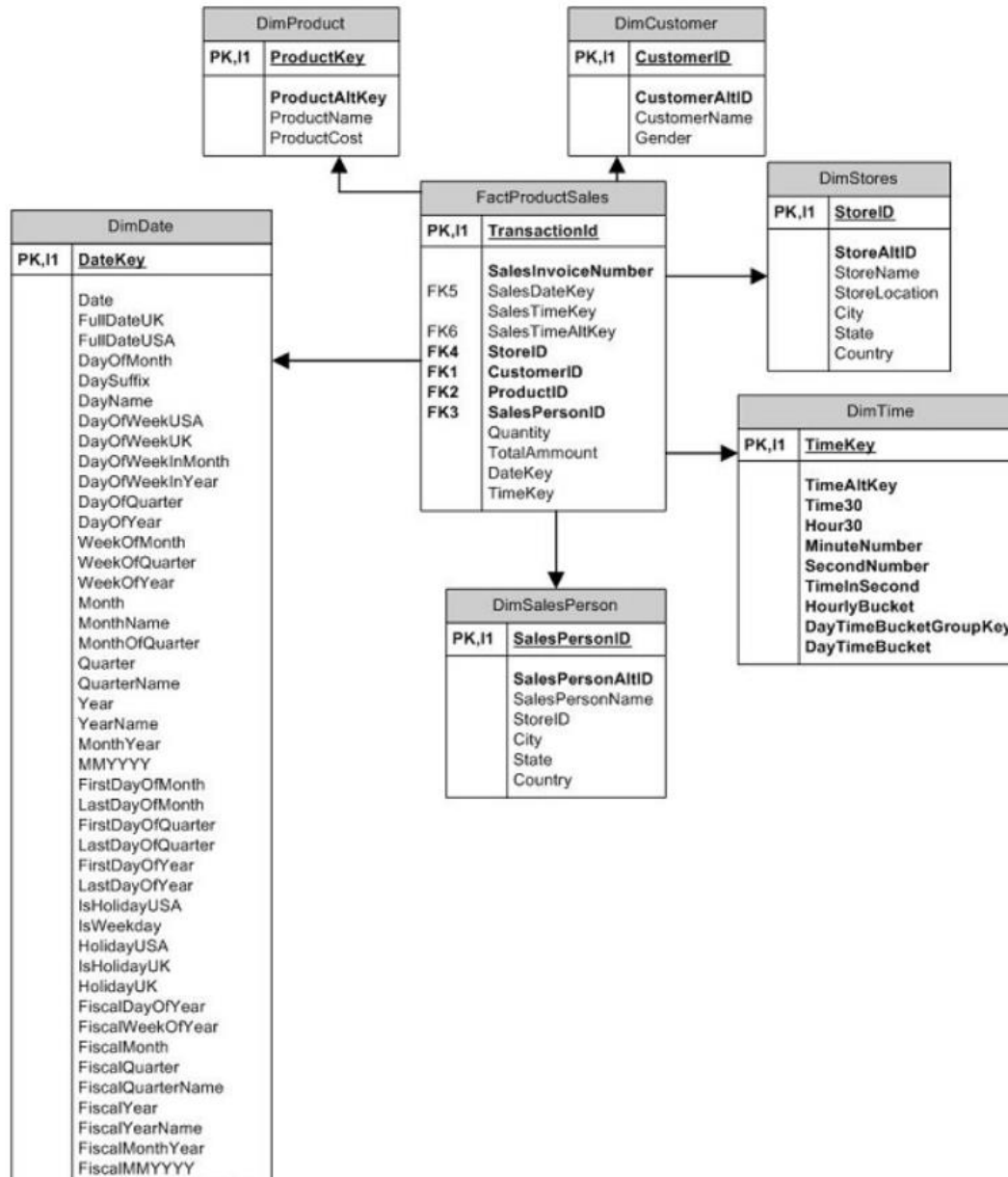
Design the Dimensional Model

- **Fact Table:**
 - Data in fact table are called measures (or dependent attributes).
 - Let us identify what attributes should be there in our Fact Sales Table.
 - **Foreign Key Column:** Sales Date key, Sales Time key, Invoice Number, Sales Person ID, Store ID, Customer ID
 - **Measures:** Actual Cost, Total Sales, Quantity, Fact table record count

Design the Relational Database

- We have identified dimensions and measures, now we have to use appropriate schema to relate this dimension and Fact tables.
- We will use Star schema due to hierarchical attribute model it provides for analysis and speedy performance in querying the data.
- In 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.

Design the Relational Database



Using the Code

- Let us execute our SQL Script step by step to create table and populate them with appropriate test values.
- Follow the given steps to run the query in My SQL.
- **Step 1**
- Create database for your Data Warehouse in My SQL:
- Create database Sales_DW;
- **Step 2**
- Create **Customer dimension** table in Data Warehouse which will hold customer personal details.

Using the Code

- Create table DimCustomer
(CustomerID int primary key identity,
CustomerAltID varchar(10) not null,
CustomerName varchar(50),
Gender varchar(20))
- 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');

Using the Code

- **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)
- 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);

Using the Code

- **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))

Using the Code

- 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');

Using the Code

- **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))

Using the Code

- 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

Using the Code

- **Step 6**
- Create **Date Dimension** table which will create and populate date data divided on various levels.
- **Step 7**
- Create **Time Dimension** table which will create and populate Time data for the entire day with various time buckets.

Using the Code

- **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.
- Example:
- Customer Henry Ford has 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.

Using the Code

- 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

Using the Code

- 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)

Using the Code

- Add Relation between Fact table and dimension tables:
- 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);
- ALTER TABLE FactProductSales ADD CONSTRAINT _FK_SalesDateKey FOREIGN KEY (SalesDateKey)REFERENCES DimDate(DateKey);
- ALTER TABLE FactProductSales ADD CONSTRAINT FK_SalesTimeKey FOREIGN KEY (SalesTimeKey)REFERENCES DimDate(TimeKey);

Using the Code

- 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),

Using the Code

- *--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),

Using the Code

- *--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)

Using the Code

- After executing the above SQL script, your sample data warehouse for sales will be ready, now you can use it for analysis and queries.

End of Slides