DATA WAREHOUSING
LECTURE 7

ENGR. MADEHA MUSHTAQ

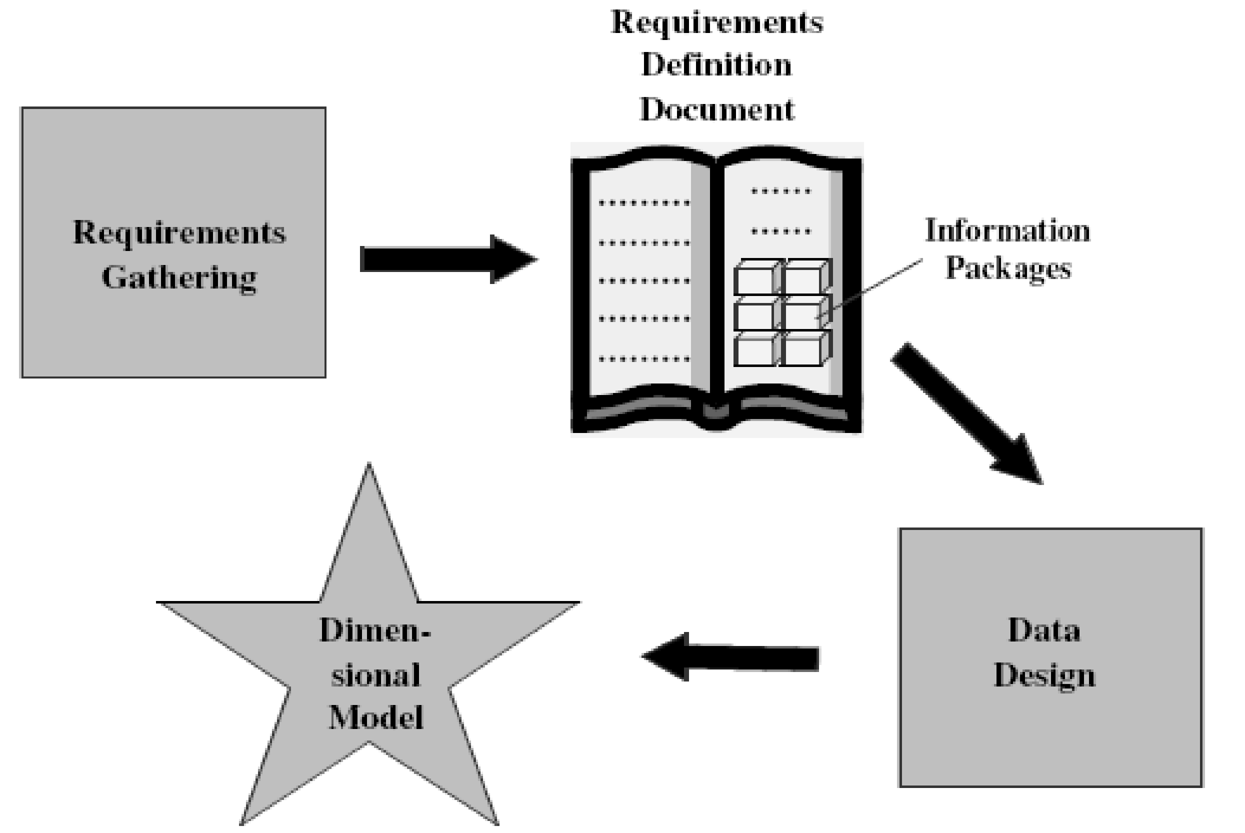
DEPARTMENT OF COMPUTER SCIENCE

IQRA NATIONAL UNIVERSITY

DIMENSIONAL MODELING

- Here are some of the criteria for combining the tables into a dimensional model:
  - The model should provide the best data access.
  - The whole model must be query-centric.
  - It must be optimized for queries and analyses.
  - The model must show that the dimension tables interact with the fact table.
  - It should also be structured in such a way that every dimension can interact equally with the fact table.

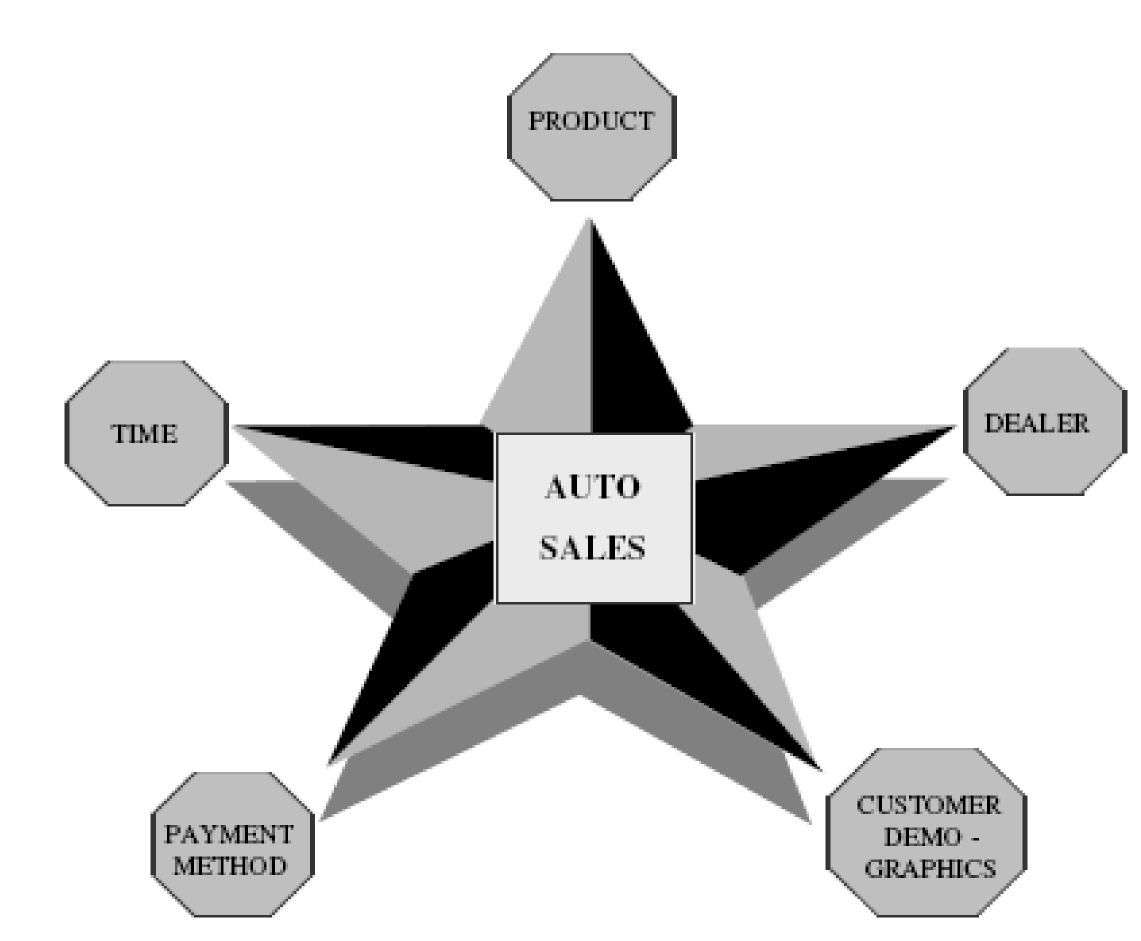
DIMENSIONAL MODELING



STAR SCHEMA

- For a dimensional model, we should have the fact table in the middle and the dimension tables arranged around the fact table.
- In this arrangement, each of the dimension tables has a direct relationship with the fact table in the middle.
- Such an arrangement in the dimensional model looks like a star formation, with the fact table at the core of the star and the dimension tables along the spikes of the star.
- This dimensional model is therefore called a STAR schema.

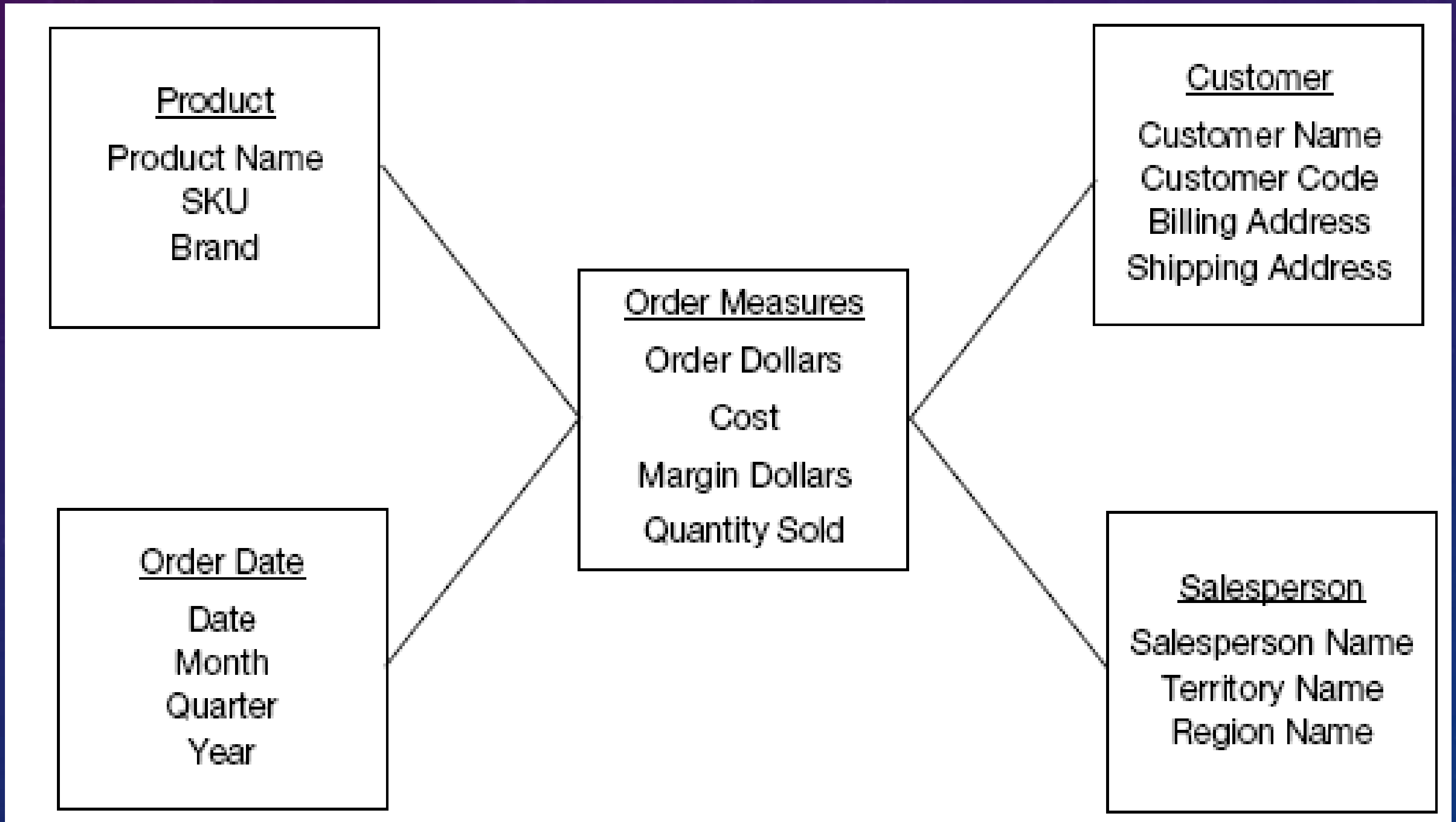
STAR SCHEMA


PRODUCT
TIME
AUTO SALES
DEALER
PAYMENT METHOD
CUSTOMER DEMO - GRAPHICS

STAR SCHEMA

- Example:
- Creating the STAR schema is the fundamental data design technique for the data warehouse.
- We will take a simple STAR schema designed for order analysis.
- Figure on next slide shows this simple STAR schema.
- It consists of the orders fact table shown in the middle of schema diagram.
- Surrounding the fact table are the four dimension tables of customer, salesperson, order date, and product.

STAR SCHEMA



Product
Product Name
SKU
Brand
Order Date
Date
Month
Quarter
Year
Order Measures
Order Dollars
Cost
Margin Dollars
Quantity Sold
Customer
Customer Name
Customer Code
Billing Address
Shipping Address
Salesperson
Salesperson Name
Territory Name
Region Name

STAR SCHEMA

- From the STAR schema, the users can easily visualize the answers to these questions:
- For a given amount of dollars, what was the product sold?
- Who was the customer?
- Which salesperson brought the order?
- When was the order placed?

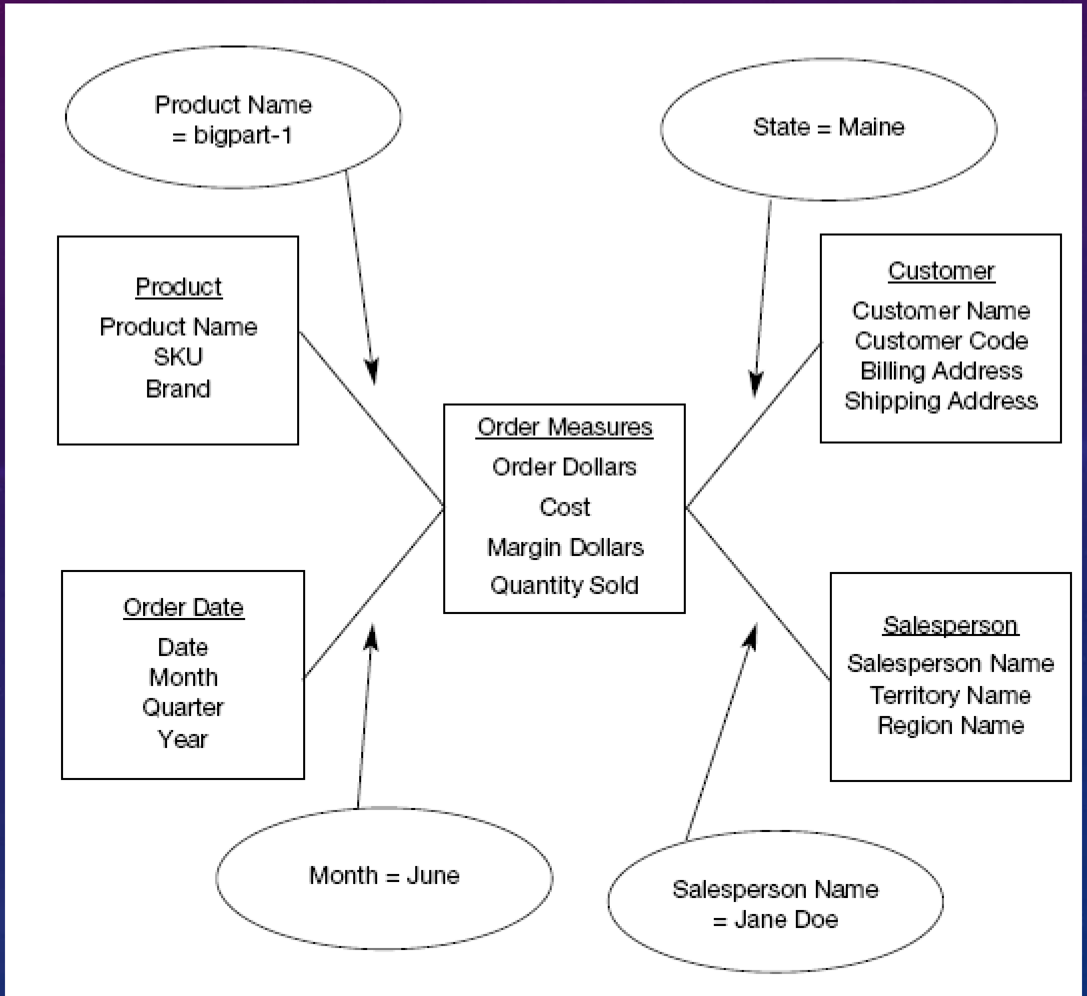
STAR SCHEMA

- When a query is made against the data warehouse, the results of the query are produced by combining or joining one or more dimension tables with the fact table.
- The joins are between the fact table and individual dimension tables.
- These individual relationships are clearly shown as the spikes of the STAR schema.

STAR SCHEMA

- Take a simple query against the STAR schema.
- Let us say that the marketing department wants
- The quantity sold and order dollars for:
  - Product bigpart-1,
  - relating to customers
  - in the state of Maine,
  - obtained by salesperson Jane Doe,
  - during the month of June.

STAR SCHEMA



Understanding a query from the STAR schema

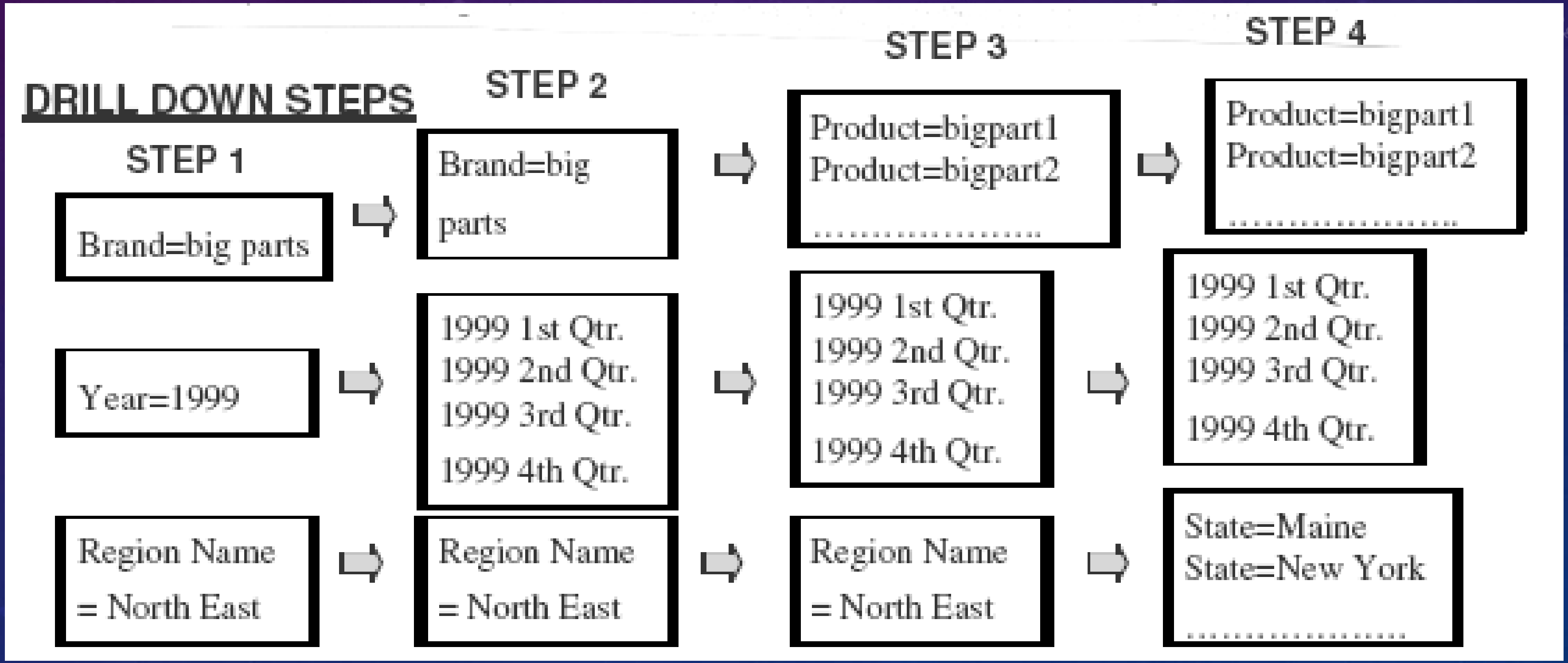
DRILL DOWN ANALYSIS FROM STAR SCHEMA

- A common type of analysis is the drilling down of summary numbers to get at the details at the lower levels.
- Suppose that the marketing department has initiated a specific analysis by placing the following query:
- Show me the total quantity sold of product brand big parts to customers in the Northeast Region for year 1999.
- In the next step of the analysis, the marketing department now wants to drill down to the level of quarters in 1999 for the Northeast Region for the same product brand, big parts.

DRILL DOWN ANALYSIS FROM STAR SCHEMA

- Next, the analysis goes down to the level of individual products in that brand.
- Finally, the analysis goes to the level of details by individual states in the Northeast Region.
- The users can easily discern all of this drill-down analysis by reviewing the STAR schema.
- Figure next shows how this drill-down is derived from the STAR schema.

DRILL DOWN ANALYSIS FROM STAR SCHEMA



DRILL DOWN STEPS

STEP 1		STEP 2		STEP 3		STEP 4
Brand=big parts	⇨	Brand=big parts	⇨	Product=bigpart1 Product=bigpart2	⇨	Product=bigpart1 Product=bigpart2 ....
Year=1999	⇨	1999 1st Qtr. 1999 2nd Qtr. 1999 3rd Qtr. 1999 4th Qtr.	⇨	1999 1st Qtr. 1999 2nd Qtr. 1999 3rd Qtr. 1999 4th Qtr.	⇨	1999 1st Qtr. 1999 2nd Qtr. 1999 3rd Qtr. 1999 4th Qtr.
Region Name = North East	⇨	Region Name = North East	⇨	Region Name = North East	⇨	State=Maine State=New York ....

INSIDE A DIMENSION TABLE

- We have seen that a key component of the STAR schema is the set of dimension tables.
- These dimension tables represent the business dimensions along which the metrics are analyzed.
- Following are some characteristics of a dimension table.

INSIDE A DIMENSION TABLE

- **Dimension table key:**
  - Primary key of the dimension table uniquely identifies each row in the table.
- **Table is wide:**
  - Typically, a dimension table has many columns or attributes.
  - It is not uncommon for some dimension tables to have more than fifty attributes.
  - Therefore, we say that the dimension table is wide.

INSIDE A DIMENSION TABLE

- **Textual attributes:**
  - In the dimension table we will seldom find any numerical values used for calculations.
  - The attributes in a dimension table are of textual format.
- **Attributes not directly related:**
  - Frequently you will find that some of the attributes in a dimension table are not directly related to the other attributes in the table.
  - For example, package size is not directly related to product brand.

INSIDE A DIMENSION TABLE

- **Not Normalized:**
  - For efficient query performance, it is best if the query picks up an attribute from the dimension table and goes directly to the fact table and not through other intermediary tables.
  - If you normalize the dimension table, you will be creating such intermediary tables and that will not be efficient.
- **Drilling down, rolling up.:**
  - The attributes in a dimension table provide the ability to get to the details from higher levels of aggregation to lower levels of details.
  - For example, the three attributes zip, city, and state form a hierarchy.

INSIDE A DIMENSION TABLE

- **Multiple hierarchies:**
  - In the example of the customer dimension, there is a single hierarchy going up from individual customer to zip, city, and state.
  - But dimension tables often provide for multiple hierarchies, so that drilling down may be performed along any of the multiple hierarchies.
- **Fewer number of records:**
  - A dimension table typically has fewer number of records or rows than the fact table.
  - A product dimension table for an automaker may have just 500 rows.
  - On the other hand, the fact table may contain millions of rows.

INSIDE THE FACT TABLE

- Fact table is where we keep the measurements.
- Following are the characteristics of the Fact Table:
- **Concatenated Key:**
    - A row in the fact table relates to a combination of rows from all the dimension tables.
    - Thus the primary key of the fact table must be the concatenation of the primary keys of all the dimension tables.
- **Data Grain:**
    - This is an important characteristic of the fact table.

INSIDE THE FACT TABLE

- **Fully Additive Measures:**
  - Aggregation of fully additive measures is done by simple addition.
  - When we run queries to aggregate measures in the fact table, we will have to make sure that these measures are fully additive.
- **Semi-additive Measures:**
  - Derived attributes such as *margin_percentage* are not additive.
  - They are known as semiadditive measures.
  - We have to distinguish between semiadditive measures from fully additive measures when we perform aggregations in queries.

INSIDE THE FACT TABLE

- **Table Deep, Not Wide:**
  - Typically a fact table contains fewer attributes than a dimension table.
  - Usually, there are about 10 attributes or less, but the number of records in a fact table is very large in comparison.
  - If we lay the fact table out as a two-dimensional table, we will note that the fact table is narrow with a small number of columns, but very deep with a large number of rows.

INSIDE THE FACT TABLE

- **Sparse Data:**
  - We have said that a single row in the fact table relates to a particular product, a specific calendar date, a specific customer, and an individual sales representative.
  - What happens when the date represents a closed holiday and no orders are received and processed? The fact table rows for such dates will not have values for the measures.
  - Do we need to keep such rows with null measures in the fact table?
  - Therefore, it is important to realize this type of sparse data and understand that the fact table could have gaps.

INSIDE THE FACT TABLE

- **Degenerate Dimensions:**
  - Look closely at the example of the fact table.
  - When you pick up attributes for the dimension tables and the fact tables from operational systems, you will be left with some data elements in the operational systems that are neither facts nor strictly dimension attributes.
  - Examples of such attributes are order numbers, invoice numbers, order line etc.
  - These attributes are called degenerate dimensions and are kept as attributes of the fact table.

STAR SCHEMA KEYS

- **Primary Keys:**
- Each row in a dimension table is identified by a unique value of an attribute designated as the primary key of the dimension.
- For the fact table, we use a concatenated primary key that is the concatenation of all the primary keys of the dimension tables.
- **Surrogate Keys:**
- We should use surrogate keys as primary key for dimension tables.
- The surrogate keys are simply system-generated sequence numbers.
- They do not have any built-in meanings.

STAR SCHEMA KEYS

- **Foreign Keys:**
- Each dimension table is in a one-to-many relationship with the central fact table.
- So the primary key of each dimension table must be a foreign key in the fact table.
- If there are four dimension tables of product, date, customer, and sales representative, then the primary key of each of these four tables must be present in the orders fact table as foreign keys.

ADVANTAGES OF THE STAR SCHEMA

- When we look at the STAR schema, we find that it is simply a relational model with a one-to-many relationship between each dimension table and the fact table.
- What is so special about the arrangement of the STAR schema?
- Although the STAR schema is a relational model, it is not a normalized model.
- The dimension tables are purposely de-normalized.
- This is a basic difference between the STAR schema and relational schemas for OLTP systems.

ADVANTAGES OF THE STAR SCHEMA

- Following are some advantages of the STAR Schema:
- Easy for Users to Understand:
  - The STAR schema reflects exactly how the users think and need data for querying and analysis.
- Optimizes Navigation:
  - A major advantage of the STAR schema is that it optimizes the navigation through the database.
  - Even when you are looking for a query result that is seemingly complex, the navigation is still simple and straightforward.

ADVANTAGES OF THE STAR SCHEMA

- **Most Suitable for Query Processing:**
  - We have already mentioned a few times that the STAR schema is a query-centric structure.
  - This means that the STAR schema is most suitable for query processing.
- **STARjoin and STARindex:**
  - STARjoin is a high-speed, single-pass, parallelizable, multitable join. It can join more than two tables in a single operation. This special scheme boosts query performance.
  - STARindex is a specialized index to accelerate join performance. These are indexes created on one or more foreign keys of the fact table.

END OF SLIDES