Star Schema Design

(Additonal Material; Partly Covered in Chapter 8)

Star Schema Overview

- Star Schema: A simple database architecture used extensively in analytical applications, particularly data marts
  - Popularized in the late 1980s by Ralph Kimball, CEO of Redbrick Systems
  - Redbrick was the first commercial star schema DBMS

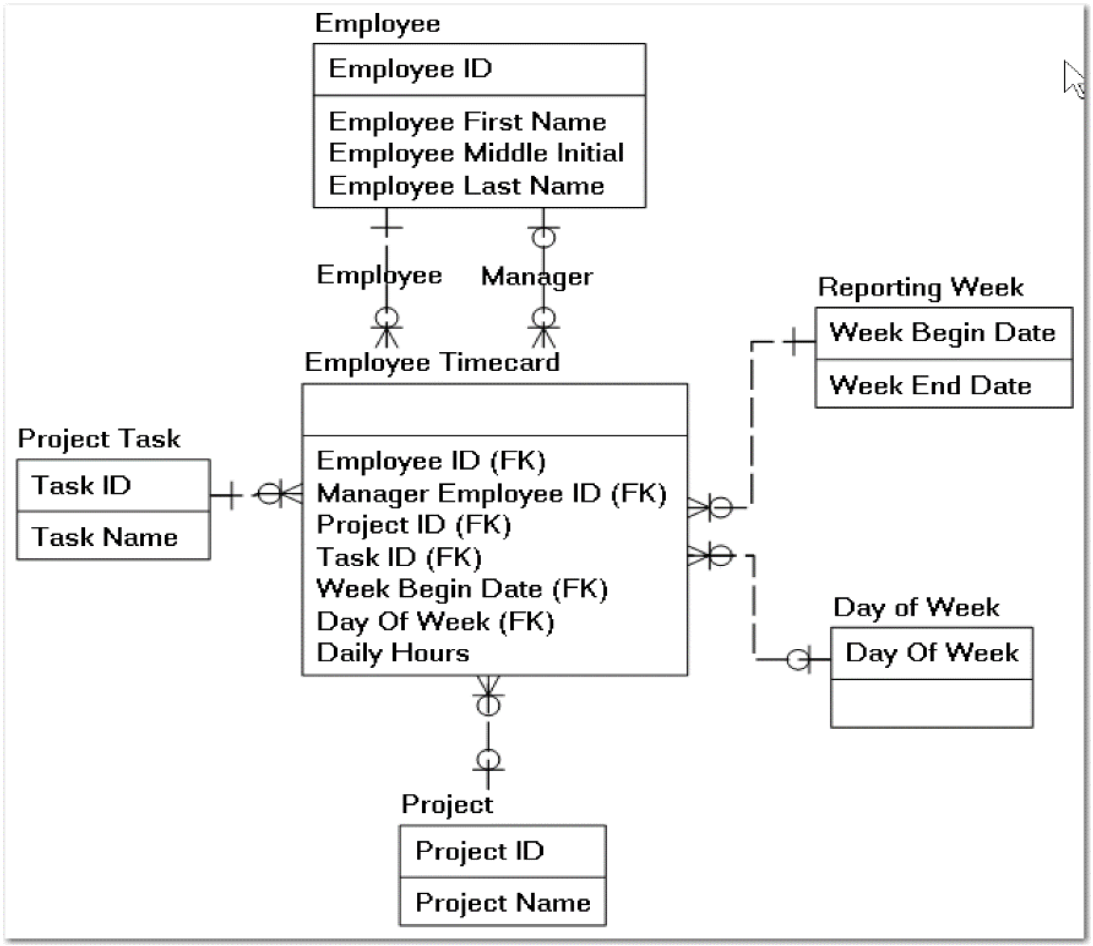
Star Schema Architecture

- Two Types of Tables:
  - ***Fact Tables***: contain ***facts***, which are quantitative measures taken from a business process.
    - Fact examples: Purchase Quantity or Amount Paid
    - Facts almost always numeric and cumulative
    - Some facts are ***key performance indicators*** (***KPI***s)
  - Dimension Tables: contain ***attributes***, which describe or characterize facts.
    - Attribute examples: Purchase Date, Product Code, and Product Description
    - Provide the business context for the facts
    - Used for filtering, sorting, and grouping accumulated facts

Star Schema and Normalization

- Normalization rules are generally not applied to star schemas
  - Not used as system of record for transaction data, but rather for analysis of transaction results
  - Typically contains history, sometimes including versions of dimensions over time

Star Schema: Employee Timecard



Fact Types

- ***Additive***: can be summed without losing business meaning (e.g. Hours Worked)
- ***NonAdditive***: cannot be summed without losing business meaning (e.g. Hourly Wage)
  - Can usually be transformed into additive facts.
  - For example, Hourly Wage can be multiplied by Hours Worked to produce additive fact Gross Earnings.
- ***Semi-Additive***: values can only be summed within some known context
  - Financial account monthly balances are semi-additive

Fact Table Design Process

- Analyze all numeric fields to see which are usable as facts
- Determine which dimensions (attributes) affect each fact's value
  - Dimensions determine the ***grain*** (level of detail) of the fact
  - For example, the grain of Daily Hours is Employee by Project Task by Reporting Week by Day of Week
- Facts can be placed in common fact tables when:
  - The grain is the same
  - The facts come from the same business event (e.g. time entry and payroll payment occur at different times)
  - Fact tables normally do not have primary keys

Orders and Shipments

Product	Date	Customer	Ordered Units
A	12/05/15	X	1
B	12/04/15	Y	2
B	12/05/15	Y	1

Product	Date	Customer	Shipped Units
A	12/05/15	X	1
B	12/05/15	Y	1
B	12/06/15	Y	1

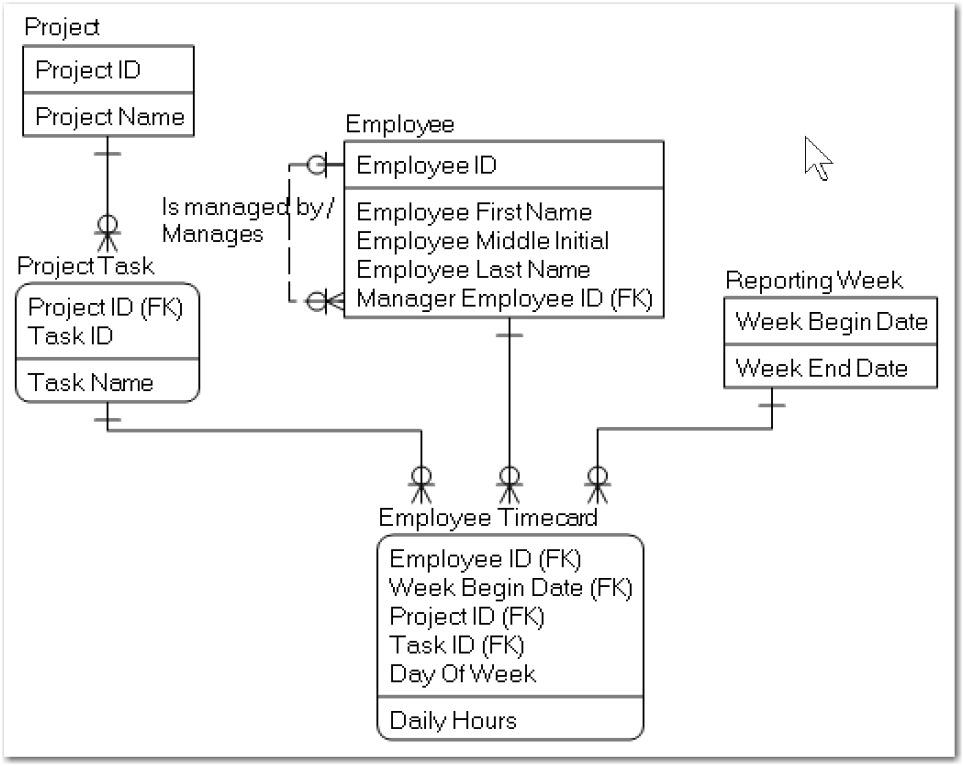
- Combined fact (below) implies Orders and Shipments in the same row are related to each other
- Orders and Shipments occur at different times
- Also, what happens if an Order has multiple shipments?
- Without an Order dimension, we cannot associate them

Product	Date	Customer	Ordered Units	Shipped Units
A	12/05/15	X	1	1
B	12/04/15	Y	2	
B	12/05/15	Y	1	1
B	12/06/15	Y		1

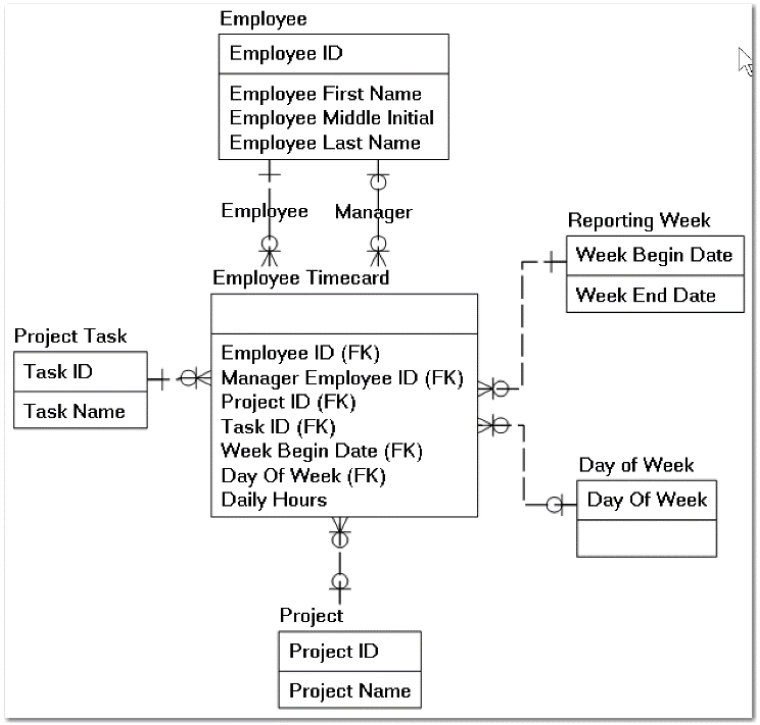
Dimension Table Design

- Once you identify the facts, everything left over is an attribute.
  - Organize into tables based on unique keys
- Dimensions are normally related only to fact tables (not to other dimensions)
- Dimension hierarchies need to be either:
  - Split: related to facts instead of each other
  - Collapsed: hierarchies flattened into lowest level dimension
  - (or a combination of the two)

Normalized Timecard Schema

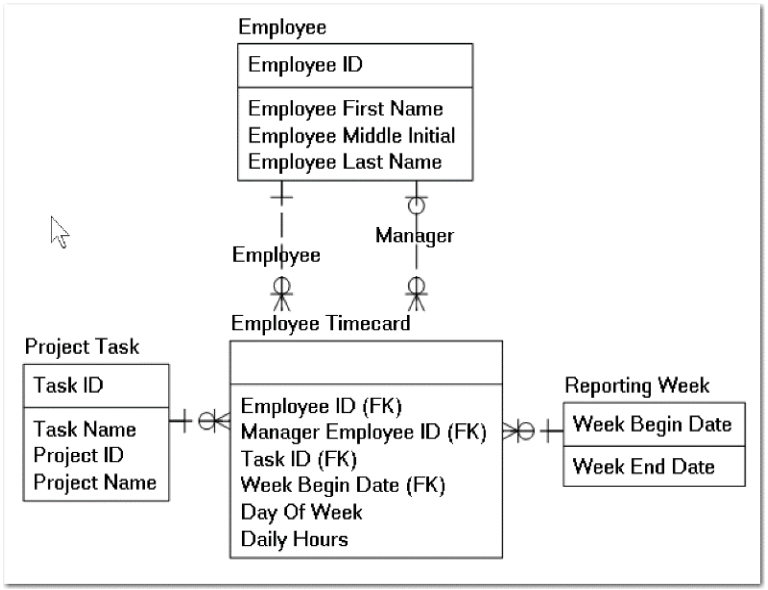


Timecard Star Schema (Split)



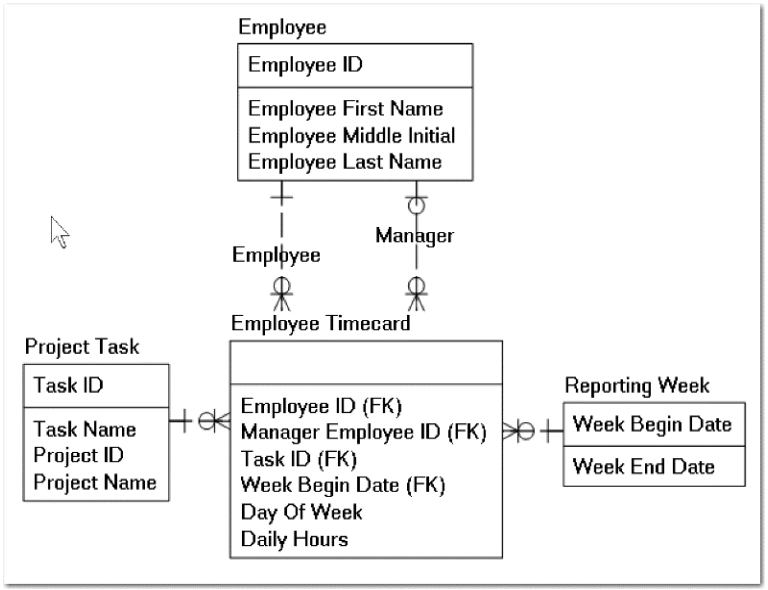
- Project and Task were split
- Recursive relationship (Manager) collapsed

Timecard Star Schema (Collapsed)



- Project collapsed into Project Task
- Splitting recommended when some facts need the coarser grain (e.g. Fact at Project grain instead of Task)

Degenerated Dimension

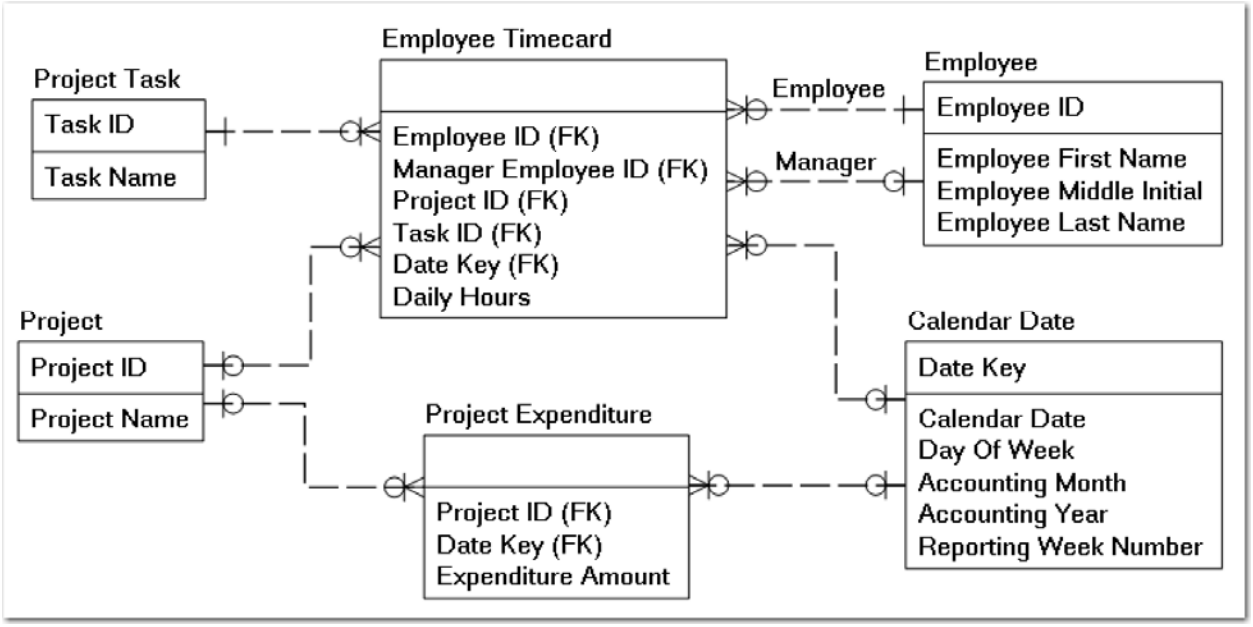


- Dimensions used by only one fact can be ***degenerated*** into the fact table
- Day of Week is a degenerated dimension attribute

Conformed Dimensions

- The only way to combine fact table data is by using one or more dimensions (accumulating across all others)
- Two dimensions are ***conformed*** when either:
  - The dimensions are exactly the same, including attribute definitions, primary keys, and all contents
  - One of the dimensions is a perfect subset of the other, meaning one dimension is a roll-up of the other.
- Shared dimensions are (and must be) conformed

Conformed Dimensions

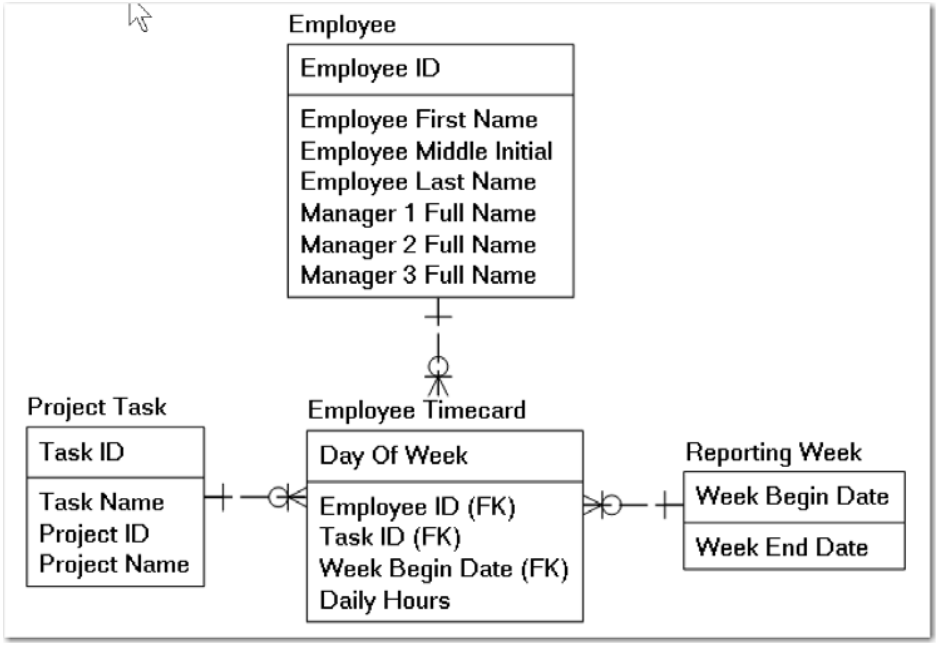


- Reporting week changed to Calendar Date for conformity
- Project is also shared (and therefore conformed)

Many-to-Many Relationships

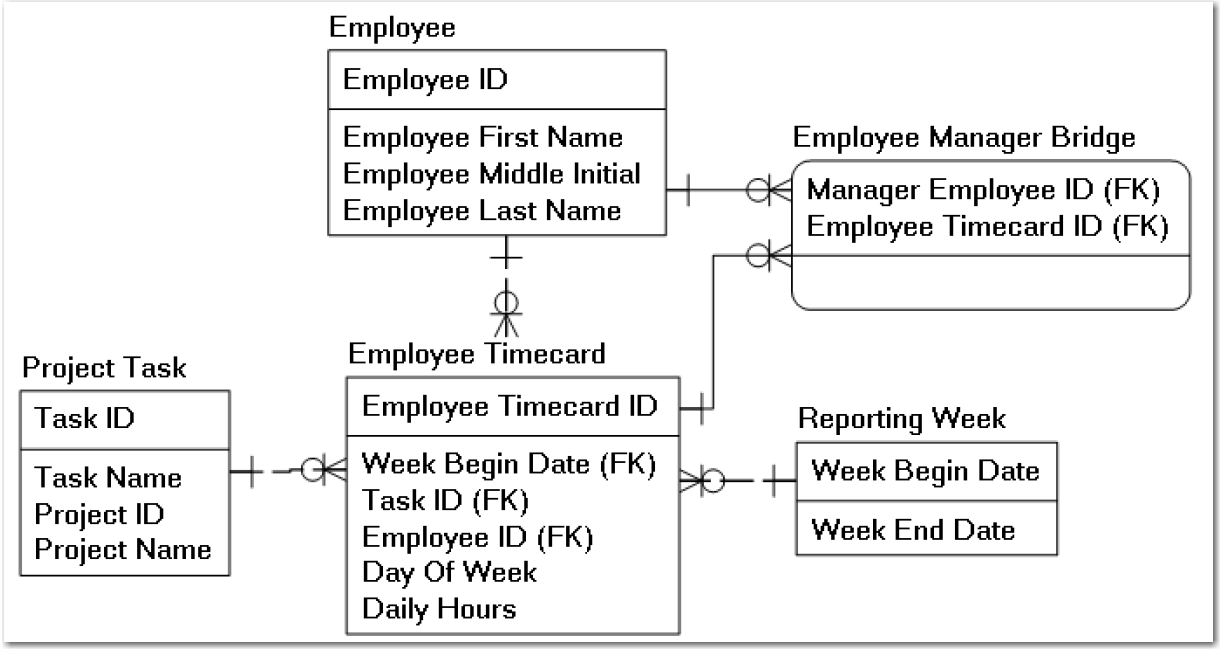
- As with normalization, handling many-to-many relationships can be complicated
- In the Time Card star schema, assume a new requirement where an Employee can have several managers.
  - This makes the relationship between Employee and Manager many-to-many
  - Or we can say that the relationship between the Employee and the Time Card is many-to-many (one employee for the subordinate, and several more for the managers).

M:N Bridge Attributes Method



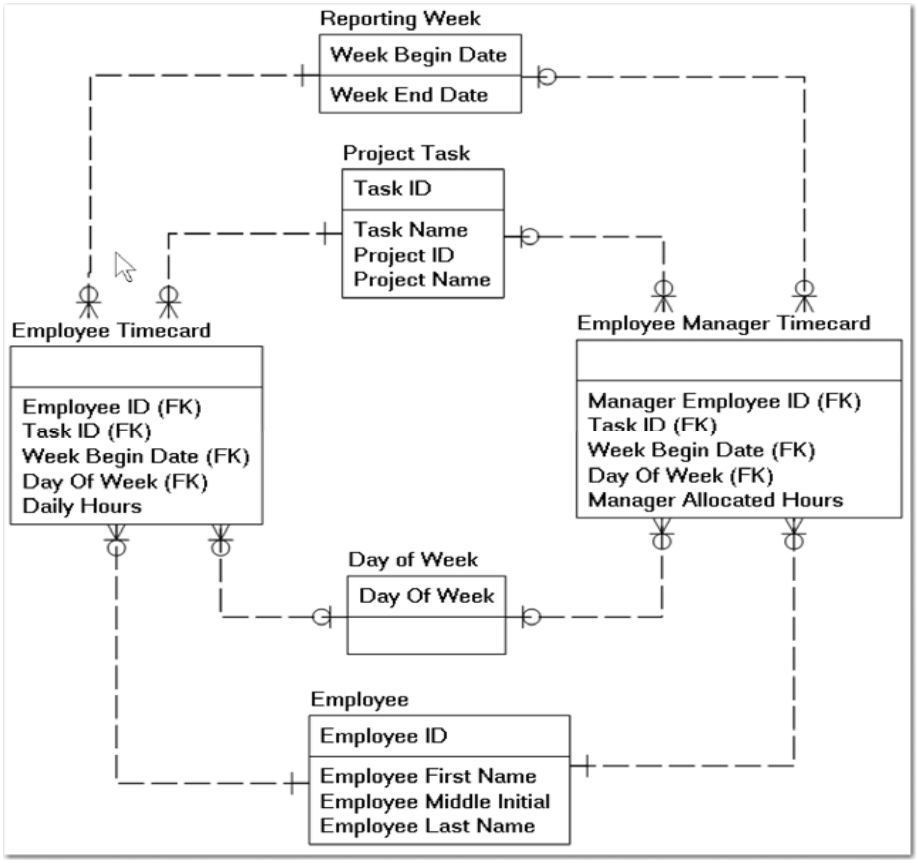
- Quick solution
- Clumsy for filtering by Manager
- Must settle on a finite number of repetitions

M:N Bridge Table Method



- Fact table usually needs a key (to be used as a foreign key in the bridge table)
- Not quite star schema, but generally accepted alternative

M:N Fact Table Method



- Fact table at M:N grain instead of bridge table
- Alternatively, can be factless fact table (if fact cannot be allocated)

Star Schema Advantages

- Star schemas are easily understood by business users.
- Star schemas are very commonly used as a user interface for analytical applications. In fact, if you have ever used pivot tables in Microsoft Excel, you have a used an implementation of the star schema.
- Queries are easier to write compared with 3NF schemas because all the dimensions are just one layer (one join) away from the fact tables.
- Compared with 3NF schemas, star schemas are easier to change and expand as business requirements change.

Star Schema Disadvantages

- Data integrity is not enforced. The source systems that provide the data for the analytical database must be responsible for data integrity.
- Star schemas do not handle many-to-many relationships as elegantly as 3NF schemas.