Lecture 8
Data Warehouse

Dr. Lydia Wahid

Agenda

- Defining Data warehouse
- Defining Data Marts and Data Lakes
- Data warehouse architecture
- Data Modeling for Data Warehouses

Defining Data warehouse

Defining Data warehouse

- A database is defined as a collection of related data. A **data warehouse** is also a **collection of information** as well as a supporting system.

- However, a clear distinction exists: Traditional databases are **transactional** and they support **online transaction processing (OLTP)** which includes insertions, updates, and deletions.

- Data warehouses have the distinguishing characteristic that they are mainly intended for **decision-support applications**. They are optimized for **data retrieval**, not routine transaction processing.

Defining Data warehouse

- A data warehouse also keeps **historical data.**

- To understand this term better, let us consider the database of a University Management System**:**
  - Once a student has left, then his or her database is usually **removed** from OLTP because OLTP is meant to perform day-to-day operations.
  - However, the management may be interested in retaining the data of old students. It can be used for later queries as well as for **analysis purposes**.
  - All such historical records can be moved to a separate data store known as the data warehouse.

Defining Data warehouse

- Historical data is not to be tampered with; no insertion, up-dation and deletion are to be made. Usually, it is used only for retrieval such as verification and data analysis.

- Thus, when data is shifted from OLTP to the data warehouse it is **de-normalized**, because normalization was earlier conducted to remove insert, update and delete anomalies but now only retrieval is important.
  - To **improve the performance of retrieval**, smaller tables are combined together to form larger tables.

- Data warehouses are typically used for **Online Analytical Processing (OLAP)** to support management queries.

Defining Data warehouse

- **OLAP (online analytical processing)** is a term used to describe the analysis of complex data from the data warehouse.

- Data warehouses are designed to support efficient extraction, processing, and presentation for analytic and decision making purposes such as:
  - OLAP
  - Decision-support systems (DSS) or Executive information systems (EIS)
  - Data mining

Defining Data warehouse

- **Data warehouse definition:**
  - According to Inmon, a data warehouse could be defined as: "A **subject-oriented**, **integrated**, **time-variant**, and **non-volatile** collection of data in support of management's decision-making process."
    - **Subject-Oriented:** they are built around the major data entity or subjects of an organization.
    - **Integrated:** integrates (combines) data from multiple systems
    - **Time variant:** data is not always up to date as it contains historical data which is valid or accurate till some point of time
    - **Non-volatile:** the previous data is not erased when new data is added

Defining Data warehouse

Traditional databases	Data warehouse
OLTP applications	OLAP applications
Hold current data	Hold current and historical data
Data is Dynamic	Data is static
Transaction-driven	Analysis-driven
Normalized data	De-normalized data
Retrieve – Insert – Update – Delete	Retrieve only
Application-oriented	Subject-oriented
Pattern of usage is predictable	Pattern of usage is unpredictable

Defining Data Marts and Data Lakes

Defining Data Marts and Data Lakes

- **Data Marts: Analytical** databases similar to data warehouses but with a defined **narrow scope**
  - It is a small localized data warehouse built for a **single purpose**.
  - It is usually built to cater to the needs of a group of users or a department in an organization.
  - A collection of data marts can constitute an enterprise-wide data warehouse.

Defining Data Marts and Data Lakes

- **Data Lakes:**
  - It is a massive storage pool for data in its natural, **raw state** (like a lake).
  - A data lake can handle huge volumes of data without the need to structure it first.
  - On the other hand, a data warehouse stores processed structured data.

Data warehouse architecture

Data warehouse architecture

- Every data warehouse has three fundamental components:
  - Load Manager
  - Warehouse Manager
  - Query manager

Data warehouse architecture

- **Load Manager:**
  - Responsible for **Data collection**
  - Performs data **conversion** into some usable form to be further utilized by the user.
  - Includes all the programs and application interfaces which are required for **extracting data**, it's **preparation** and finally **loading** of data into the data warehouse itself.

Data warehouse architecture

- Warehouse insertions are handled by the warehouse's **ETL (extract, transform, load)** process, which does a large amount of preprocessing.
  - **Extract:**
    - Raw data is copied or exported from source locations
    - Data management teams can extract data from a variety of data sources, which can be structured or unstructured
  - **Transform:**
    - A set of rules are applied over the data, in order to transform it from source format to target format
    - This phase can involve the following tasks: Filtering, cleansing, validating, authenticating the data, performing calculations, translations, or summarizations based on the raw data
  - **Load:**
    - Transformed data is moved to the target data warehouse

Data warehouse architecture

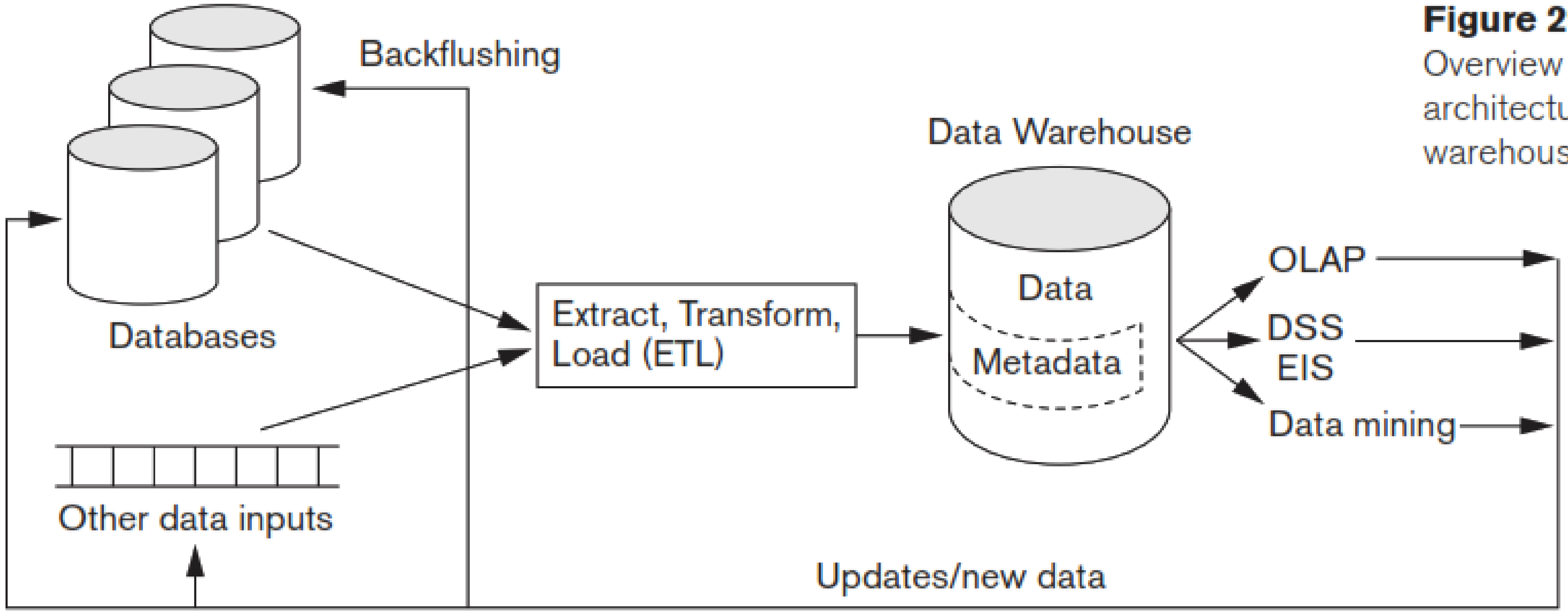


Figure 29.1
Overview of the general architecture of a data warehouse.

Data warehouse architecture

- **Warehouse Manager:**
  - It is the main part of Data Warehousing system as it **holds the massive amount of information**.
  - It **organizes data** to analyze it or find the required information.
  - It maintains <u>three levels of information</u>, i.e, detailed, lightly summarized and highly summarized.
  - It also maintains **metadata**, i.e., data about data.
- **Query Manager:**
  - Query manager is that interface which connects the end users with the information stored in data warehouse through the usage of specialized end-user tools.

Data Modeling for Data Warehouses

Data Modeling for Data Warehouses

- Multidimensional models populate data in multidimensional matrices called **data cubes**. (These may be called **hypercubes** if they have more than three dimensions)

- Query performance in multidimensional matrices can be much better than in the relational data model.

- More than three dimensions **cannot be easily visualized**; however, the data can be **queried directly** in any combination of dimensions, thus bypassing complex database queries.

Data Modeling for Data Warehouses

- The figure shows three-dimensional data cube that organizes product sales data by fiscal quarters and sales regions.
- Each cell contains data for a specific product, specific fiscal quarter, and specific region.

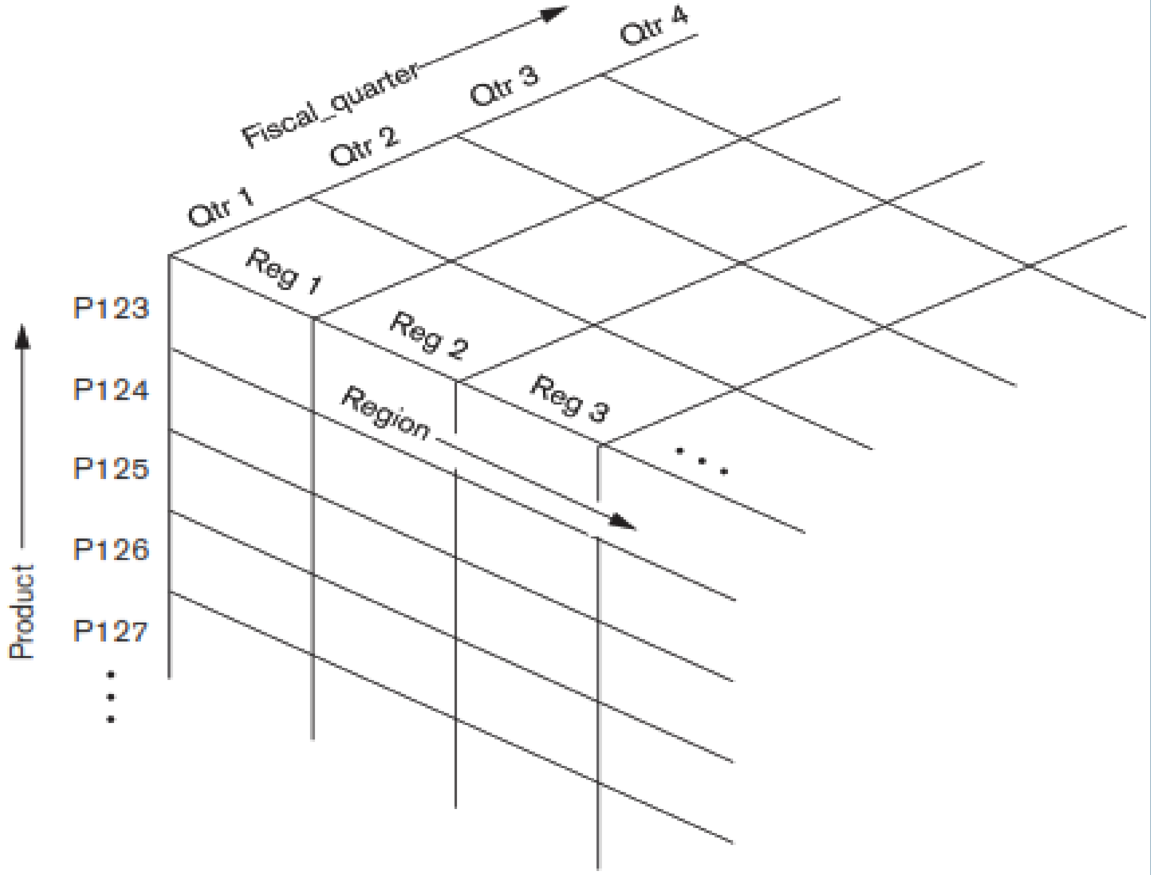


Figure 29.3
A three-dimensional data cube model.

Data Modeling for Data Warehouses

- The term **slice** is used to refer to a two-dimensional view of a three- or higher-dimensional cube.
- The term **"slice and dice"** implies a systematic reduction of a body of data into smaller chunks or views so that the information is made visible from multiple angles or viewpoints.
- Multidimensional models lend themselves readily to hierarchical views in what is known as roll-up display and drill-down display.
- A **roll-up display** moves up the hierarchy, grouping into larger units along a dimension (for example, summing weekly data by quarter)
- A **drill-down display** provides the opposite capability, for example, disaggregating country sales by region and then regional sales by subregion.
  - Typically, in a warehouse, the drill-down capability is limited to the lowest level of aggregated data stored in the warehouse.

Data Modeling for Data Warehouses

- The multidimensional model involves two types of tables: **dimension tables** and **fact tables**.
- A **dimension table** consists of the attributes of the dimension.
- A **fact table** can be thought of as having tuples, one per a recorded fact.
- The fact table contains the data, and the dimensions identify each tuple in that data.

Data Modeling for Data Warehouses

- **Logical descriptions** of database are known as Schema. It is the blueprint of the entire database.
- It defines **how the data are organized** and how the **relations among them** are associated.
- A database uses relational models whereas a data warehouse uses different types of schema, namely, **Star**, **Snowflake**, and **Fact Constellation.**

Data Modeling for Data Warehouses

- The **star schema** consists of a fact table with a single table for each dimension. The fact table is at the center and the dimension tables at the nodes of the star.

- Generally, fact tables are in third normal form (3NF) in the case of star schema while dimensional tables are in **de-normalized** form.

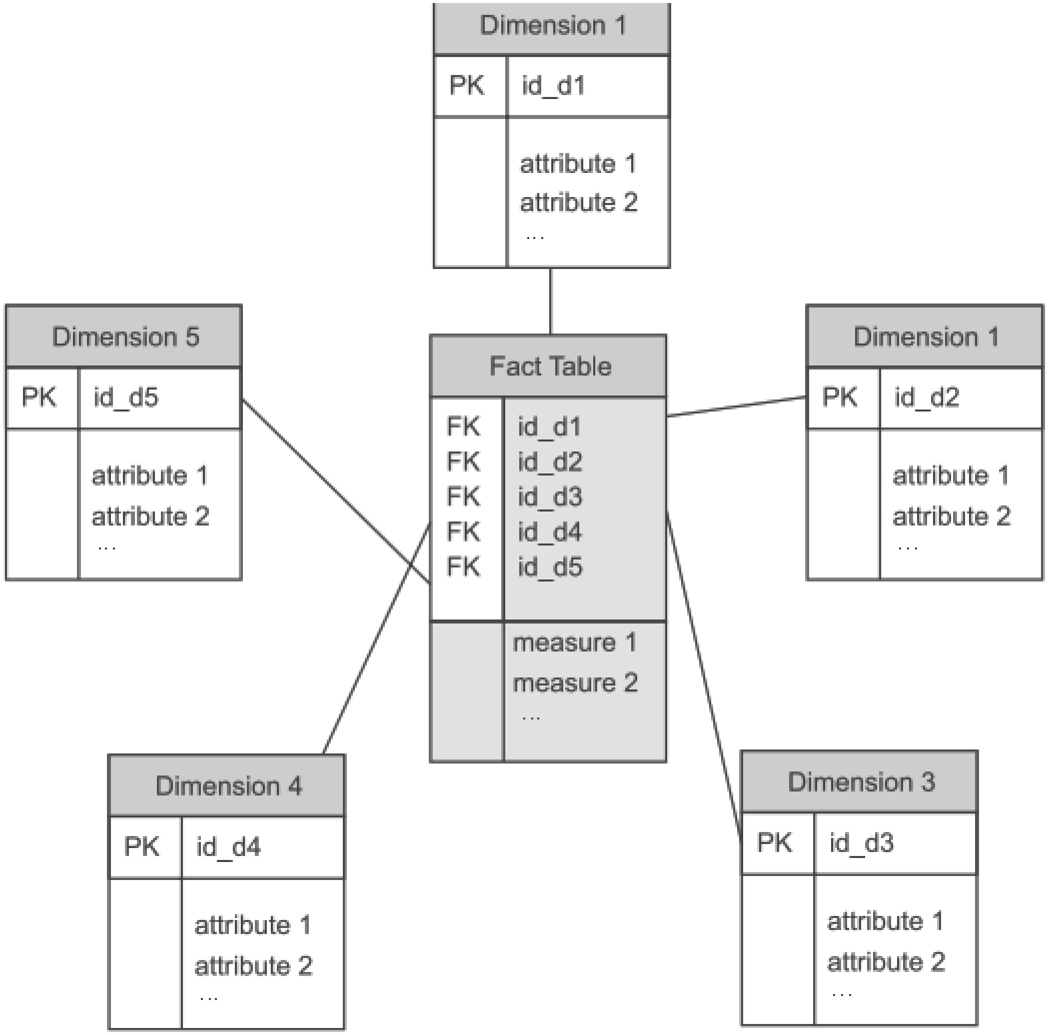


Figure 13.5 Graphical representation of Star schema

Data Modeling for Data Warehouses

➢ An example of **star schema:**

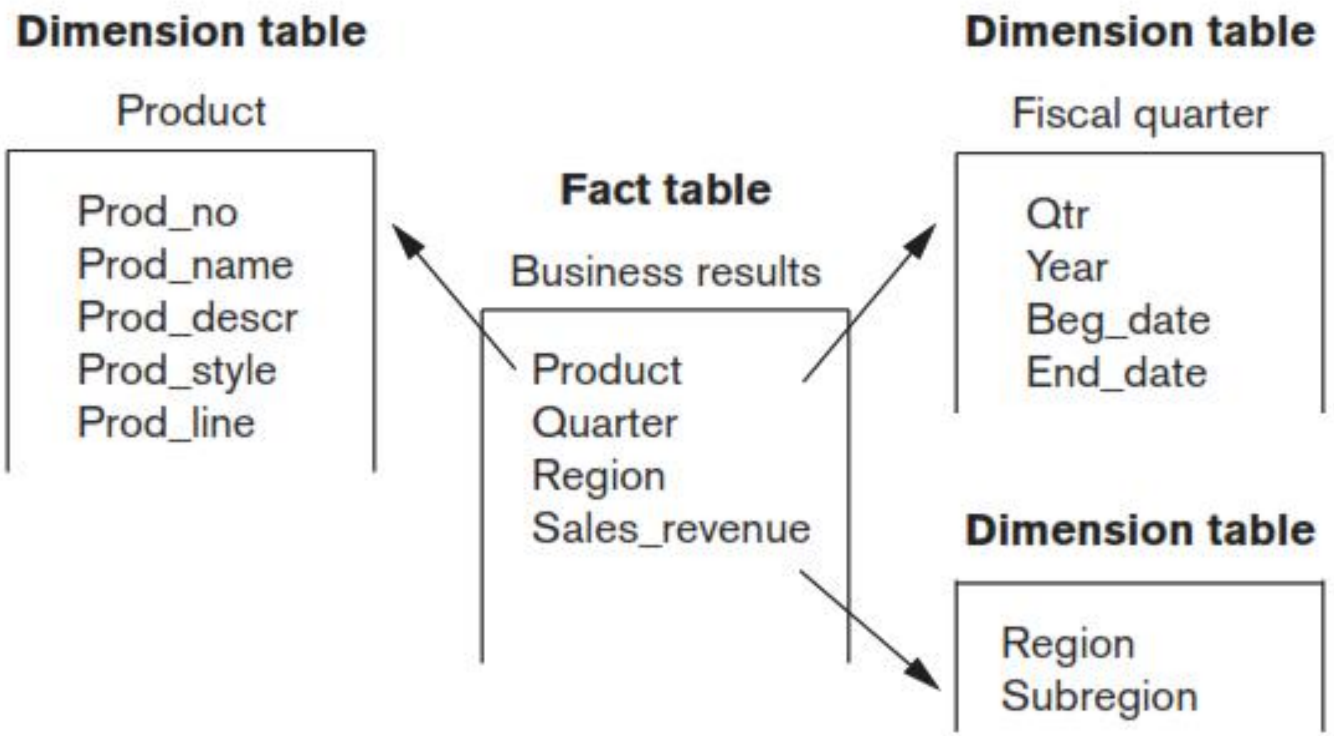
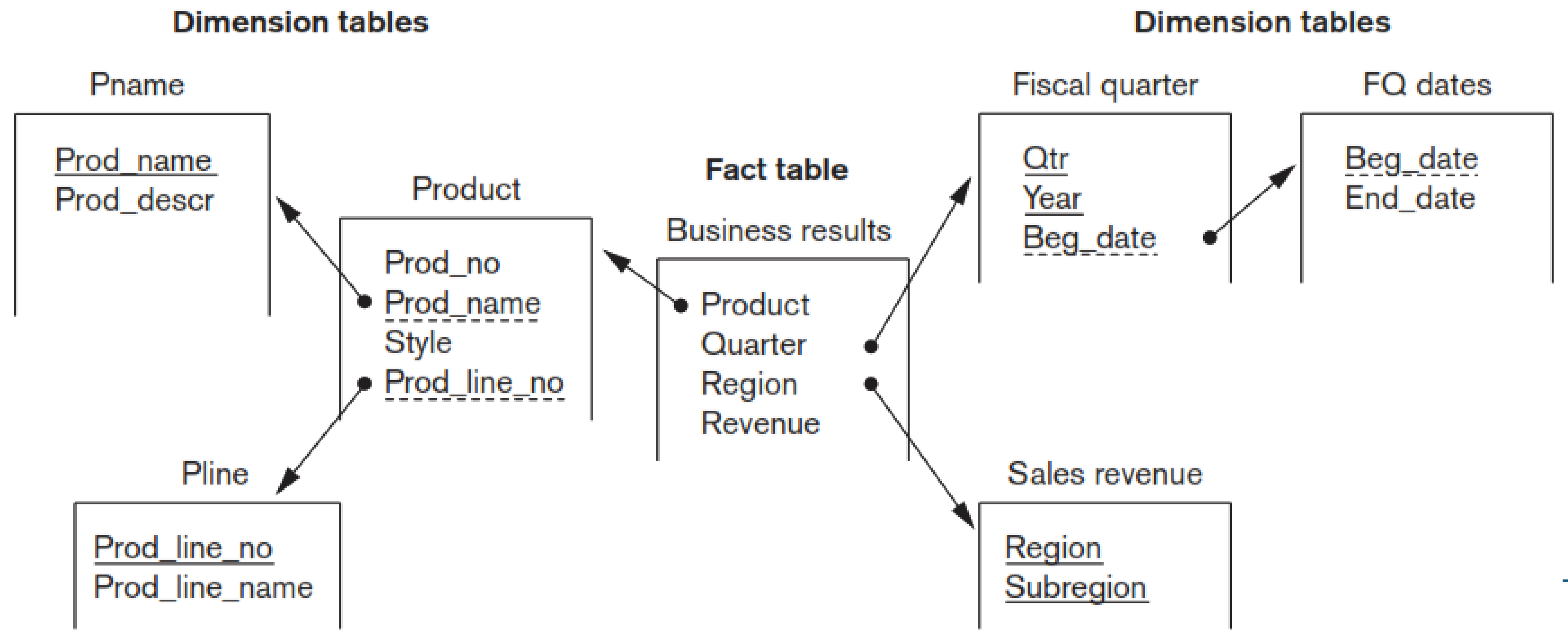


Figure 29.7
A star schema with fact and dimensional tables.

Data Modeling for Data Warehouses

- The **snowflake schema** is a variation on the star schema in which the dimensional tables are **normalized**.

Figure 29.8
A snowflake schema.



Data Modeling for Data Warehouses

- A **fact constellation schema** consists of multiple fact tables. It is a set of fact tables that share some dimension tables.
- Fact constellations limit the possible queries for the warehouse.

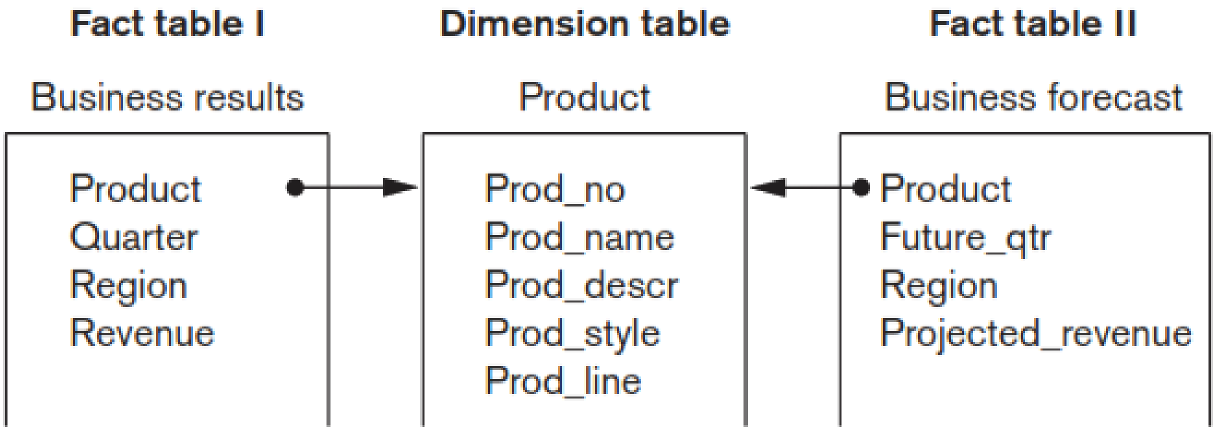


Figure 29.9
A fact constellation.

Data Modeling for Data Warehouses

- The fact table contains the specific measurable (or quantifiable) primary data to be analyzed, such as sales records, logged performance data or financial data.

- It may be **transactional** -- in that rows are added as events happen -- or it may be a snapshot of **historical data** up to a point in time.

- The fact table stores two types of information: **numeric values** and **dimension attribute values** (the foreign key value for a row in a related dimensional table)

Star schema

Warehouse (dimension table)

WAREHOUSE_ID	OFFICE NAME	CITY	MANAGER NAME
1	Texas	Houston	John
2	Florida	Orlando	Phil

Items (dimension table)

ITEM_ID	ITEM NAME	ITEM COLOR	...
1	Sedan	Blue	•••
2	Truck	Brown	•••

Date (dimension table)

DATE_ID	MONTH	YEAR	QUARTER
202005	May	2020	Q2

Orders (fact table)

VALUE DATA			DIMENSION ATTRIBUTE DATA				
ORDER ID	ORDER PROFIT	ORDER QUANTITY	ITEM_ID	WAREHOUSE_ID	EMPLOYEE_ID	DATE_ID	CUSTOMER_ID
101	$100	1	1	1	2	202005	1
102	$200	2	1	2	1	202005	2
103	$200	2	1	1	3	202005	2
104	$400	2	2	1	2	202005	1
105	$800	4	2	2	3	202005	2

Employee (dimension table)

EMPLOYEE_ID	NAME	GENDER	OFFICE	PHONE
1	Joe	M	Texas	•••
2	Jane	F	Utah	•••
3	Jill	F	Texas	•••

Customer (dimension table)

CUSTOMER_ID	NAME	ADDRESS	PHONE
1	Bill	123 Place	•••
2	Ben	456 Street	•••

Snowflake schema

Color (dimension table)

COLOR_ID	NAME	METALIC
1	Blue	Yes
2	Brown	No

Warehouse (dimension table)

WAREHOUSE_ID	OFFICE NAME	CITY	MANAGER NAME
1	Texas	Houston	John
2	Florida	Orlando	Phil

Items (dimension table)

ITEM_ID	ITEM NAME	COLOR_ID
1	Sedan	Blue
2	Truck	Brown

Date (dimension table)

DATE_ID	MONTH	YEAR	QUARTER
202005	May	2020	Q2

Orders (fact table)

VALUE DATA			DIMENSION ATTRIBUTE DATA				
ORDER ID	ORDER PROFIT	ORDER QUANTITY	ITEM_ID	WAREHOUSE_ID	EMPLOYEE_ID	DATE_ID	CUSTOMER_ID
101	$100	1	1	1	2	202005	1
102	$200	2	1	2	1	202005	2
103	$200	2	1	1	3	202005	2
104	$400	2	2	1	2	202005	1
105	$800	4	2	2	3	202005	2

Employee (dimension table)

EMPLOYEE_ID	NAME	GENDER	OFFICE	PHONE
1	Joe	M	Texas	•••
2	Jane	F	Utah	•••
3	Jill	F	Texas	•••

Customer (dimension table)

CUSTOMER_ID	NAME	ADDRESS	PHONE
1	Bill	123 Place	•••
2	Ben	456 Street	•••

Gender (dimension table)

GENDER_ID	NAME	SALUTATION
1	Male	Mr.
2	Female	Ms.

Office (dimension table)

OFFICE_ID	NAME	ADDRESS
1	Texas	5 Road
2	Utah	6 Lane

Data Modeling for Data Warehouses

Table 13.1 Comparison among Star, Snowflake and Fact Constellation Schema

Parameter	*Star*	*Snowflake*	*Fact constellation*
Query Joins	Require simple joins	Requires complicated joins	Requires complicated joins
Data structure	De-normalized data structure	Normalized data structure.	Normalized data structure.
Number of Fact Tables	Single fact table	Single fact table	Multiple fact tables
Query Performance	It gives faster query results due to fewer join operations.	It is slow in query processing due to larger join operations.	It is slow in query processing due to larger join operations.
Dimension	Dimension table does not split into pieces.	Dimension tables are split into many pieces.	Dimension tables are split into many pieces.
Data Redundancy	Data is redundant due to de-normalization.	Data is not redundant as dimensions are normalized.	Data is not redundant as dimensions are normalized.
Data Integrity	Tough to enforce data integrity due to redundancy of data.	Easy to enforce data integrity due to no redundancy of data.	Easy to enforce data integrity due to no redundancy of data.

Data Modeling for Data Warehouses

- Data warehouse storage also utilizes indexing techniques to support high performance access.
- A technique called **bitmap indexing** constructs a bit vector for each value in a domain (column) being indexed. It works very well for domains of low cardinality.
- Bitmap indexing can provide considerable input/output and storage space advantages in low-cardinality domains.
- With bit vectors, a bitmap index can provide dramatic improvements in comparison, aggregation, and join performance.

Data Modeling for Data Warehouses

- There is a 1 bit placed in the jth position in the vector if the jth row contains the value being indexed.

- For example, imagine an inventory of 100,000 cars with a bitmap index on car size. If there are four car sizes—economy, compact, mid-size, and full-size—there will be four bit vectors, each containing 100,000 bits (12.5kbytes) for a total index size of 50K.

References

- Bhatia, P. (2019). Chapter 12 Data Warehouse. In *Data Mining and Data Warehousing: Principles and Practical Techniques*. Cambridge: Cambridge University Press.

- Bhatia, P. (2019). Chapter 13 Data Warehouse Schema. In *Data Mining and Data Warehousing: Principles and Practical Techniques*. Cambridge: Cambridge University Press.

- Elmasri, R., & Navathe, S. (2016). Chapter 29 Overview of Data Warehousing and OLAP. In *Fundamentals of database systems 7th ed.*, Pearson Education.

Thank You