Module 1

DWH Fundamentals

What is Data Warehouse?

- A data warehouse is a large collection of business data used to help an organization make decisions.

- It is a system used for **reporting** and **data analysis** and is considered a core component of **business intelligence.**

- DWs are central repositories of integrated data from one or more disparate sources. They store current and historical data in one single place that are used for creating analytical reports for workers throughout the enterprise.

- According to William H. Inmon, a leading architect in the construction of data warehouse systems, "A data warehouse is a **subject-oriented, integrated, time-variant, and nonvolatile** collection of data in support of management's decision making process"

  The four keywords—subject-oriented, integrated, time-variant, and nonvolatile—distinguish data warehouses from other data repository systems, such as relational database systems, transaction processing systems, and file systems

What is Data Warehouse?

- A data warehouse is a large collection of business data used to help an organization make decisions.
- A system used for **reporting** and **data analysis** ---a core component of **business intelligence**.
- DWs are central repositories of integrated data from one or more disparate sources. They store current and historical data in one single place .
- According to William H. Inmon, a leading architect in the construction of data warehouse systems, "A data warehouse is a **subject-oriented, integrated, time-variant, and nonvolatile** collection of data in support of management's decision making process"

Data warehouse characteristics

There are basic features that define the data in the data warehouse that include subject orientation, data integration, time-variant, nonvolatile data, and data granularity.

- **Subject-oriented:** Unlike the operational systems, the data in the data warehouse revolves around the subjects of the enterprise. Subject orientation is not database normalization. Subject orientation can be really useful for decision-making. Gathering the required objects is called subject-oriented.

- **Integrated:** The data found within the data warehouse is integrated. Since it comes from several operational systems, all inconsistencies must be removed. Consistencies include naming conventions, measurement of variables, encoding structures, physical attributes of data, and so forth.

- **Nonvolatile:** The data in the data warehouse is read-only, which means it cannot be updated, created, or deleted (unless there is a regulatory or statutory obligation to do so).

- **Time-variant:** While operational systems reflect current values as they support day-to-day operations, data warehouse data represents a long time horizon (up to 10 years) which means it stores mostly historical data. It is mainly meant for data mining and forecasting. (E.g. if a user is searching for a buying pattern of a specific customer, the user needs to look at data on his current and past purchases.)

- **Subject-oriented:** A data warehouse is organized around major subjects such as customer, supplier, product, and sales.

  Rather than concentrating on the day-to-day operations and transaction processing of an organization, a data warehouse focuses on the modeling and analysis of data for decision makers.

  Hence, data warehouses typically provide a simple and concise view of particular subject issues by excluding data that are not useful in the decision support process.

- I**ntegrated:** A data warehouse is usually constructed by integrating multiple heterogeneous sources, such as relational databases, flat files, and online transaction records.

  Data cleaning and data integration techniques are applied to ensure consistency in naming conventions, encoding structures, attribute measures, and so on.

DWH Characteristics

- **Subject-oriented:** A data warehouse is organized around major subjects .

  A data warehouse focuses on the modeling and analysis of data for decision makers.

  Hence, data warehouses typically provide a simple and concise view of particular subject issues .

- **Integrated:** A data warehouse is usually constructed by integrating multiple heterogeneous sources, such as relational databases, flat files, and online transaction records.

  Data cleaning and data integration techniques are applied to ensure consistency in naming conventions, encoding structures, attribute measures, and so on.

DWH Characteristics

- **Time-variant:** Data are stored to provide information from an historic perspective (e.g., the past 5–10 years). Every key structure in the data warehouse contains, either implicitly or explicitly, a time element.

  –Data in the DWH is mainly meant for data mining and forecasting.

- **Nonvolatile:** The data in the data warehouse is read-only.

Operational DBS Vs. DWH

- **Data Contents:**

Operational DB Systems: Current and detailed data and is subject to modifications.

Data Warehouse: Historical data, course granularity, generally not modified.

- **Users:**

Operational DB Systems: *Customer – Oriented, thus used by customers/clerks/IT professionals.*

Data Warehouse: Market – Oriented, thus used by Managers/Executives/Analysts.

- **Database Design:**

Operational DB Systems: *Usually E-R model.*

Data Warehouse: Usually Multidimensional model. (Star, Snowflake…)

- **Nature of Queries:**

Operational DB Systems: *Short, atomic queries desiring high performance (less latency) and accuracy.*

Data Warehouse: Mostly read only queries, operate on HUGE volumes of data, queries are quite complex.

OLTP Vs OLAP

OLTP	OLAP
OLTP is an **online transactional system.**	OLAP is an online analysis and data retrieving process.
It is characterized by large numbers of short online transactions.	It is characterized by a large volume of data.
OLTP is an online database modifying system.	OLAP is an online database query management system.
OLTP uses traditional DBMS.	OLAP uses the data warehouse.
Insert, Update, and Delete information from the database.	Mostly select operations
OLTP and its transactions are the sources of data.	Different OLTP databases become the source of data for OLAP.
OLTP database must maintain data integrity constraints.	OLAP database does not get frequently modified. Hence, data integrity is not an issue.
It's response time is in a millisecond.	Response time in seconds to minutes.
The data in the OLTP database is always detailed and organized.	The data in the OLAP process might not be organized.
Allow read/write operations.	Only read and rarely write.
It is a customer-oriented process.	It is a market oriented process.
Queries in this process are standardized and simple.	Complex queries involving aggregations.
Complete backup of the data combined with incremental backups.	OLAP only need a backup from time to time. Backup is not important compared to OLTP
DB design is an application-oriented example: Database design changes with the industry like retail, airline, banking, etc.	DB design is subject-oriented. Example: Database design changes with subjects like sales, marketing, purchasing, etc.

OLTP	OLAP
It is used by Data critical users like clerk, DBA & Data Base professionals.	It is used by Data knowledge users like workers, managers, and CEO.
It is designed for real time business operations.	It is designed for analysis of business measures by category and attributes.
Transaction throughput is the performance metric	Query throughput is the performance metric.
This kind of Database allows thousands of users.	This kind of Database allows only hundreds of users.
It helps to Increase user's self-service and productivity	Help to Increase the productivity of business analysts.
It provides a fast result for daily used data.	It ensures that response to the query is quicker consistently.
It is easy to create and maintain.	It lets the user create a view with the help of a spreadsheet.

Why to have a separate Warehouse?

3 Main reasons:

1. OLTP systems require high concurrency, reliability, locking which provide good performance for short and simple OLTP queries. An OLAP query is very complex and does not require these properties. Use of OLAP query on OLTP system **degrades its performance**.
2. An OLAP query reads HUGE amount of data and generates the required result. The query is very complex too. Thus **special primitives** have to provided to support this kind of data access.
3. OLAP systems access historical data and not current volatile data while OLTP systems access current up-to-date data and do not need historical data.

Thus,

Solution is to have a separate database system which supports primitives and structures suitable to store, access and process OLAP specific data …

in short…have a data warehouse.

What data is stored in a DWH?

- In simple words: Subject(s) per Dimension

  Example: If our subject/measure is ‘quantity sold’ and if the dimensions are : Item Type, Location and Period then,

  Data warehouse stores the items sold per type, per geographical location during the particular period.

- Data warehousing is the process of construction and using data warehouses.
- The construction of a data warehouse requires data cleaning, data integration, and data consolidation.
- The utilization of a data warehouse often necessitates a collection of decision support technologies.
- This allows “knowledge workers” (e.g., managers, analysts, and executives) to use the warehouse to quickly and conveniently obtain an overview of the data, and to make sound decisions based on information in the warehouse.

“How are organizations using the information from data warehouses?”

Many organizations use this information to support business decision-making activities, including:
(1) increasing customer focus, which includes the **analysis of customer buying patterns** (such as buying preference, buying time, budget cycles, and appetites for spending);

(2) repositioning products and managing product portfolios **by comparing the performance of sales by quarter, by year,** and by geographic regions in order to fine-tune production strategies;

(3) analyzing operations and looking for sources of profit; and

(4) managing customer relationships, making environmental corrections, and managing the cost of corporate assets.

“How are organizations using the information from data warehouses?”

Many organizations use this information to support business decision-making activities, including:
(1) increasing customer focus, which includes the analysis of customer buying patterns ;

(2) repositioning products and managing product portfolios (by comparing the performance of sales by quarter, by year, and by geographic regions in order to fine-tune production strategies);

(3) analyzing operations and looking for sources of profit; and

(4) managing customer relationships, making environmental corrections, and managing the cost of corporate assets.

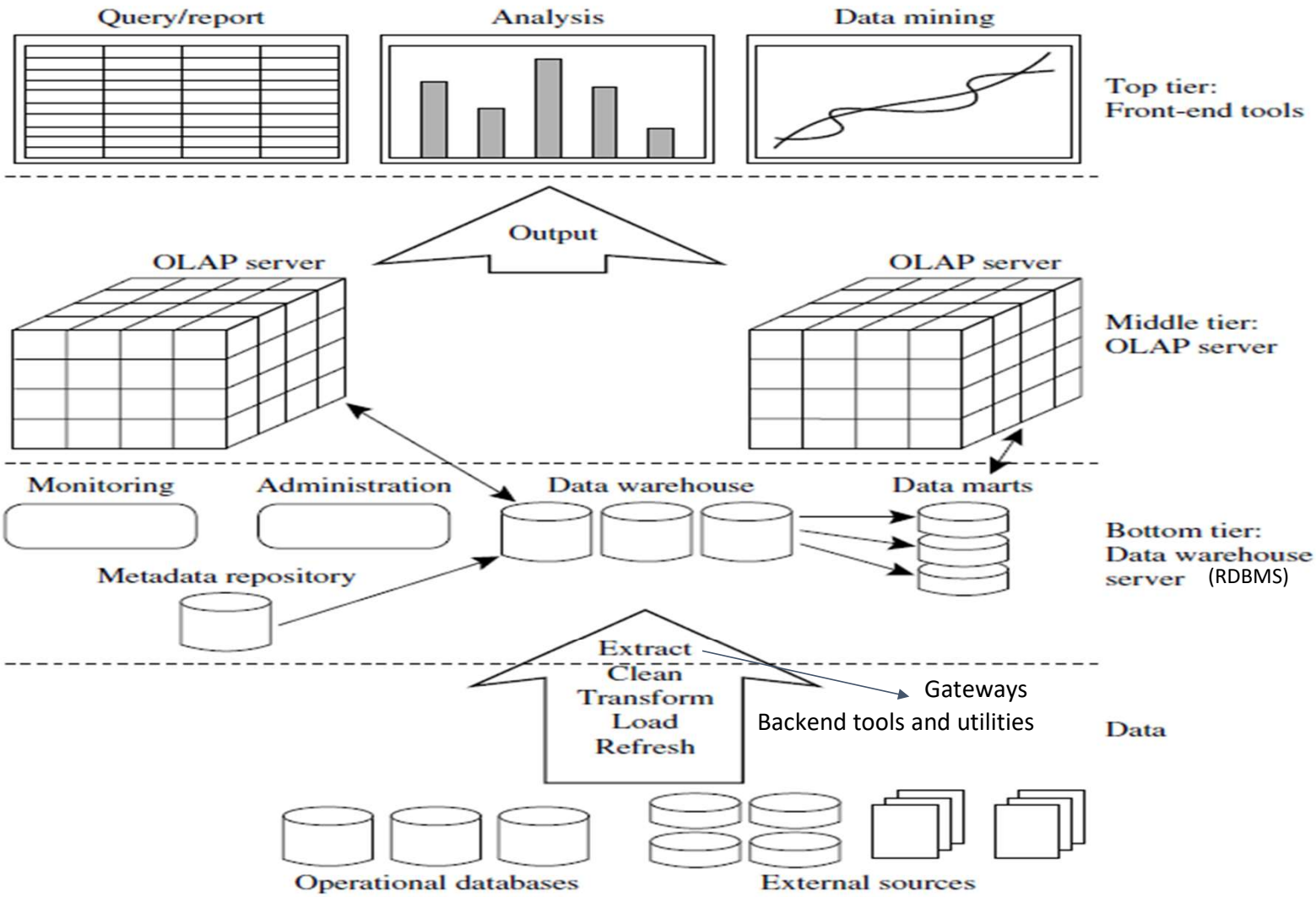


Figure 4.1 A three-tier data warehousing architecture.

Extraction, Transformation, and Loading

Data warehouse systems use back-end tools and utilities to populate and refresh their Data.These tools and utilities include the following functions:

- Data extraction, which typically gathers data from multiple, heterogeneous, and external sources.
- Data cleaning, which detects errors in the data and rectifies them when possible.
- Data transformation, which converts data from legacy or host format to warehouse Format.
- Load, which sorts, summarizes, consolidates, computes views, checks integrity, and builds indices and partitions.
- Refresh, which propagates the updates from the data sources to the warehouse.

Data Warehouse Models: Enterprise Warehouse, Data Mart, and Virtual Warehouse

Enterprise warehouse:

- Collects information about subjects spanning the entire organization.
- It provides corporate-wide data integration
- It typically contains detailed data as well as summarized data.
- An enterprise data warehouse may be implemented on traditional mainframes, computer superservers, or parallel architecture platforms.
- It requires extensive business modeling and may take years to design and build

Data mart:

- A data mart contains a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to specific selected subjects.

- The data contained in data marts tend to be summarized.

- Data marts are usually implemented on low-cost departmental servers that are Unix/Linux or Windows based.

- The implementation cycle of a data mart is more likely to be measured in weeks rather than months or years. However, it may involve complex integration in the long run if its design and planning were not enterprise-wide.

- Depending on the source of data, data marts can be categorized as independent or dependent.
  - —Independent data marts are sourced from data captured from one or more operational systems or external information providers, or from data generated locally within a particular department or geographic area.
  - —Dependent data marts are sourced directly from enterprise data warehouse

Virtual warehouse:

- A virtual warehouse is a set of views over operational databases.

For efficient query processing, only some of the possible summary views may be materialized.

- A virtual warehouse is easy to build but requires excess capacity on operational database servers.

Types of Data Warehouse: 3 main types

1. **Enterprise Data Warehouse (EDW):**

- A centralized warehouse.
- It provides decision support service across the enterprise.
- It offers a unified approach for organizing and representing data.
- It also provide the ability to classify data according to the subject and give access according to those divisions.

1. **Operational Data Store:**

- A data store required when neither Data warehouse nor OLTP systems support organizations reporting needs.
- In ODS, Data warehouse is refreshed in real time.
- Hence, it is widely preferred for routine activities

3. **Data Mart:**

- A data mart is a subset of the data warehouse.
- It is specially designed for a particular line of business, such as sales, finance.
- In an independent data mart, data can be collected directly from sources.

How Data warehouse works?

- The large amount of data in data warehouses comes from different places such as internal applications like marketing, sales, and finance; customer-facing apps; and external partner systems etc..
- On a technical level, a data warehouse periodically pulls data from those apps and systems; then, the data goes through formatting and import processes to match with the data that is already in the warehouse. The data warehouse stores this processed data so that it is ready for decision makers to access.
- How frequently data pulls occur, or how data is formatted, etc., will vary depending on the needs of the organization.

How Data warehouse works?

- On a technical level, a data warehouse periodically pulls data from apps and systems;
- Then, the data goes through formatting and import processes to match with the data that is already in the warehouse.
- The data warehouse stores this processed data so that it is ready for decision makers to access.


Decision Support System
Data Warehouse
Executive Information System
Other Names for Datawarehouse
Analytic Application
Management Information System
Business Intelligence Solution

Who needs Data warehouse?

DWH (Data warehouse) is needed for all types of users like:

- Decision makers who rely on mass amount of data
- Users who use customized, complex processes to obtain information from multiple data sources.
- It is also used by the people who want simple technology to access the data
- It is also essential for those people who want a systematic approach for making decisions.
- If the user wants fast performance on a huge amount of data which is a necessity for reports, grids or charts, then Data warehouse proves useful.
- Data warehouse is a first step If you want to discover ‘hidden patterns’ of data-flows and groupings.

Advantages of Data Warehouse (DWH):

- Data warehouse allows users to access critical data from the number of sources in a single place. Therefore, it saves user's time of **retrieving data from multiple sources.**
- Data warehouse provides consistent information on various cross-functional activities. It also supports ad-hoc reporting and query.
- Data Warehouse helps to **integrate many sources of data to reduce stress on the production system.**
- Data warehouse helps to reduce total turnaround time for **analysis and reporting.**
- Restructuring and Integration make it easier for the user to use for reporting and analysis.
- Data warehouse stores a large amount of historical data. This helps users to analyze different **time periods and trends to make future predictions.**

Parameter	Database	Data Warehouse
Purpose	Is designed to record	Is designed to analyze
Processing Method	The database uses the Online Transactional Processing (OLTP)	Data warehouse uses Online Analytical Processing (OLAP).
Usage	The database helps to perform fundamental operations for your business	Data warehouse allows you to analyze your business.
Tables and Joins	Tables and joins of a database are complex as they are normalized.	Table and joins are simple in a data warehouse because they are denormalized.
Orientation	Is an application-oriented collection of data	It is a subject-oriented collection of data
Storage limit	Generally limited to a single application	Stores data from any number of applications
Availability	Data is available real-time	Data is refreshed from source systems as and when needed
Usage	ER modeling techniques are used for designing.	Data modeling techniques are used for designing.
Technique	Capture data	Analyze data
Data Type	Data stored in the Database is up to date.	Current and Historical Data is stored in Data Warehouse. May not be up to date.
Storage of data	Flat Relational Approach method is used for data storage.	Data Warehouse uses dimensional and normalized approach for the data structure. Example: Star and snowflake schema.
Query Type	Simple transaction queries are used.	Complex queries are used for analysis purpose.
Data Summary	Detailed Data is stored in a database.	It stores highly summarized data.

Applications of Database

Sector	Usage
Banking	Used in the banking sector for customer information, account-related activities, payments, deposits, loans, credit cards, etc.
Airlines	Used for reservations and schedule information.
Universities	To store student information, course registrations, colleges, and results.
Telecommunication	It helps to store call records, monthly bills, balance maintenance, etc.
Finance	Helps you to store information related stock, sales, and purchases of stocks and bonds.
Sales & Production	Use for storing customer, product and sales details.
Manufacturing	It is used for the data management of the supply chain and for tracking production of items, inventories status.
HR Management	Detail about employee's salaries, deduction, generation of paychecks, etc.

Applications of Data Warehousing

Sector	Usage
Airline	It is used for airline system management operations like crew assignment, analysis of route, frequent flyer program discount schemes for passenger, etc.
Banking	It is used in the banking sector to manage the resources available on the desk effectively.
Healthcare sector	Data warehouse used to strategize and predict outcomes, create patient's treatment reports, etc. Advanced machine learning, big data enable data warehouse systems to predict illness.
Insurance sector	Data warehouses are widely used to analyze data patterns, customer trends, and to track market movements quickly.
Retain chain	It helps you to track items, identify the buying pattern of the customer, promotions and also used for determining pricing policy.
Telecommunication	In this sector, data warehouse is used for product promotions, sales decisions and to make distribution decisions.

Disadvantages of Database

- Cost of Hardware and Software of an implementing Database system is high which can increase the budget of your organization.
- Many DBMS systems are often complex systems, so the training for users to use the DBMS is required.
- DBMS can’t perform sophisticated calculations
- Issues regarding compatibility with systems which is already in place
- Data owners may lose control over their data raising security, ownership, and privacy issues.

Disadvantages of Data Warehouse

- Creation and Implementation of Data Warehouse is surely time confusing affair.
- Data Warehouse **can be outdated** relatively quickly
- **Adding new data sources takes time, and it is associated with high cost.**
- Sometimes problems associated with the data warehouse may be undetected for many years.
- Data warehouses are high maintenance systems. **Extracting, loading, and cleaning data could be time-consuming.**
- The data warehouse may look simple, but actually, it is too complicated for the average users. You need to provide training to end-users, who end up not using the data mining and warehouse.
- Despite best efforts at project management, the scope of data warehousing will always increase.

Metadata Repository

When used in a data warehouse, metadata are the data that define warehouse objects.

A metadata repository should contain the following:

- A description of the data warehouse structure, which includes the warehouse schema, view, dimensions, hierarchies, and derived data definitions, as well as data mart locations and contents

- Operational metadata, which include data lineage (history of migrated data and the sequence of transformations applied to it), currency of data (active, archived, or purged), and monitoring information (warehouse usage statistics, error reports, and audit trails).

- The algorithms used for summarization, which include measure and dimension definition algorithms, data on granularity, partitions, subject areas, aggregation, summarization, and predefined queries and reports.

Metadata Repository

- Mapping from the operational environment to the data warehouse, which includes source databases and their contents, gateway descriptions, data partitions, data extraction, cleaning, transformation rules and defaults, data refresh and purging rules, and security (user authorization and access control).
- Data related to system performance, which include indices and profiles that improve data access and retrieval performance, in addition to rules for the timing and scheduling of refresh, update, and replication cycles.
- Business metadata, which include business terms and definitions, data ownership information, and charging policies.

Data Warehouse Modeling: Data Cube and OLAP

- Data warehouses and OLAP tools are based on a multidimensional data model.
- This model views data in the form of a data cube.
- A data cube allows data to be modeled and viewed in multiple dimensions. It is defined by dimensions and facts/measures.
- Dimensions are the perspectives or entities with respect to which an organization wants to keep records.
- Each dimension may have a table associated with it, called a dimension table, which further describes the dimension.
- Dimension tables can be specified by users or experts, or automatically generated and adjusted based on data distributions

Data Warehouse Modeling: Data Cube and OLAP

- A multidimensional data model is typically organized around a central theme, such as sales. This theme is represented by a fact table.

- Facts are numeric measures.(quantities by which we want to analyze relationships between dimensions).

- The fact table contains the names of the facts, or measures, as well as keys to each of the related dimension tables.

- In data warehousing the data cube is n-dimensional.

Table 4.2 2-D View of Sales Data for *AllElectronics* According to *time* and *item*

	location = "Vancouver"			
	item (type)			
time (quarter)	home entertainment	computer	phone	security
Q1	605	825	14	400
Q2	680	952	31	512
Q3	812	1023	30	501
Q4	927	1038	38	580

Note: The sales are from branches located in the city of Vancouver. The measure displayed is *dollars_sold* (in thousands).

- Now, suppose that we would like to view the sales data according to time and item, as well as location, for the cities Chicago, New York, Toronto, and Vancouver.

Table 4.3 3-D View of Sales Data for *AllElectronics* According to *time*, *item*, and *location*

	location = "Chicago"				***location*** = "New York"				***location*** = "Toronto"				***location*** = "Vancouver"			
	item				***item***				***item***				***item***			
	home				*home*				*home*				*home*			
time	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	*sec.*	*ent.*	*comp.*	*phone*	*sec.*
Q1	854	882	89	623	1087	968	38	872	818	746	43	591	605	825	14	400
Q2	943	890	64	698	1130	1024	41	925	894	769	52	682	680	952	31	512
Q3	1032	924	59	789	1034	1048	45	1002	940	795	58	728	812	1023	30	501
Q4	1129	992	63	870	1142	1091	54	984	978	864	59	784	927	1038	38	580

Note: The measure displayed is *dollars_sold* (in thousands).

The 3-D data in the table are represented as a series of 2-D tables.

3-D data cube representation for the previous table data

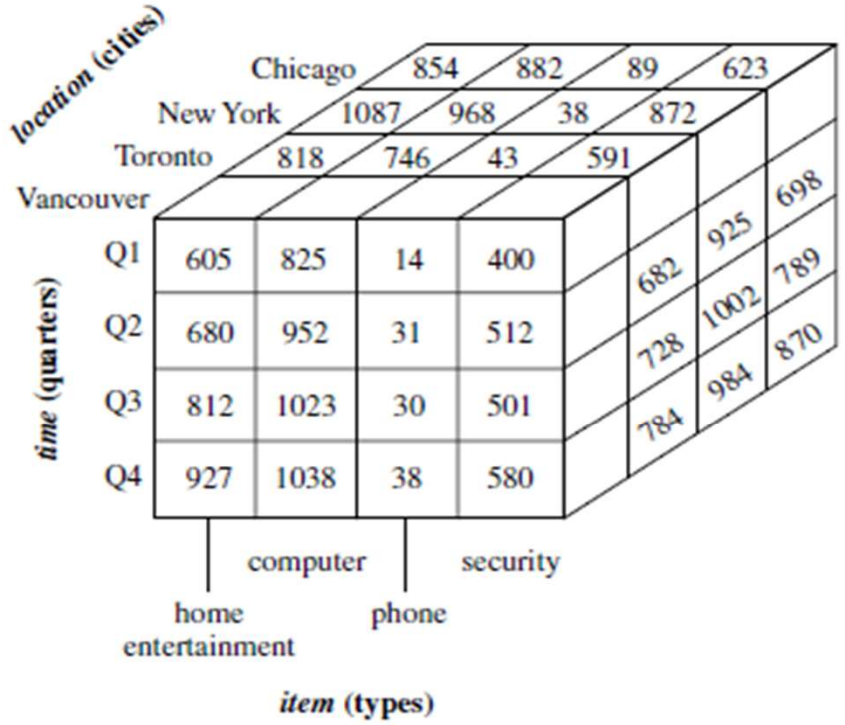


Figure 4.3 A 3-D data cube representation of the data in Table 4.3, according to *time*, *item*, and *location*. The measure displayed is *dollars_sold* (in thousands).

Table 4.3 3-D View of Sales Data for *AllElectronics* According to *time*, *item*, and *location*

	location = "Chicago"				location = "New York"				location = "Toronto"				location = "Vancouver"			
	item				item				item				item			
	home				home				home				home			
time	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.
Q1	854	882	89	623	1087	968	38	872	818	746	43	591	605	825	14	400
Q2	943	890	64	698	1130	1024	41	925	894	769	52	682	680	952	31	512
Q3	1032	924	59	789	1034	1048	45	1002	940	795	58	728	812	1023	30	501
Q4	1129	992	63	870	1142	1091	54	984	978	864	59	784	927	1038	38	580

Note: The measure displayed is *dollars_sold* (in thousands).

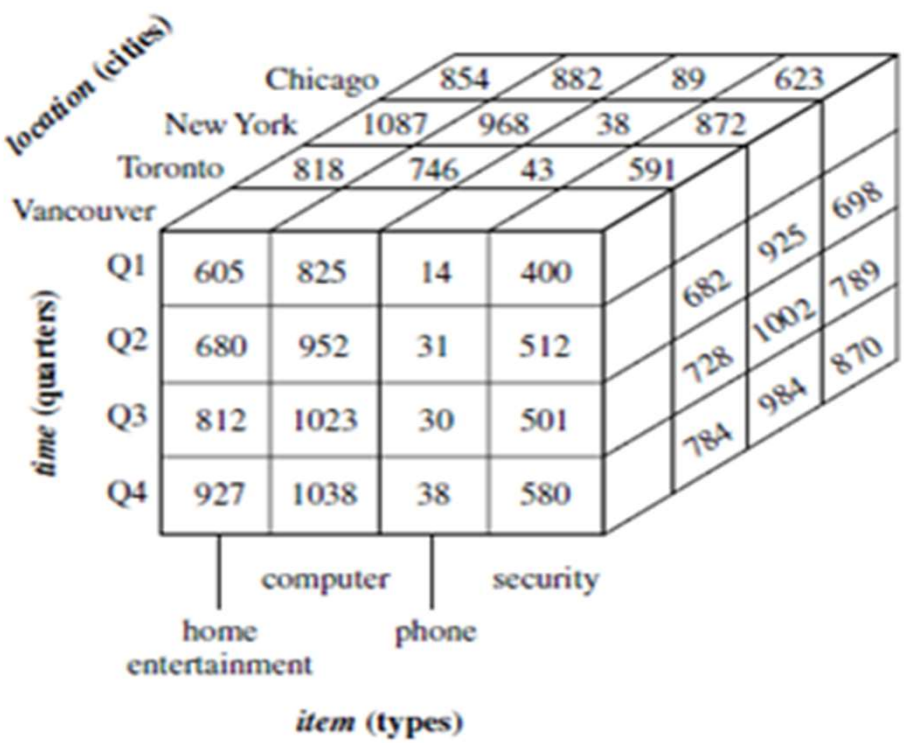


Figure 4.3 A 3-D data cube representation of the data in Table 4.3, according to *time*, *item*, and *location*. The measure displayed is *dollars_sold* (in thousands).

- Now I want to view same data with additional 4th dimension as supplier
- Here ,we can think of a 4-D cube as being a series of 3-D cubes as below:

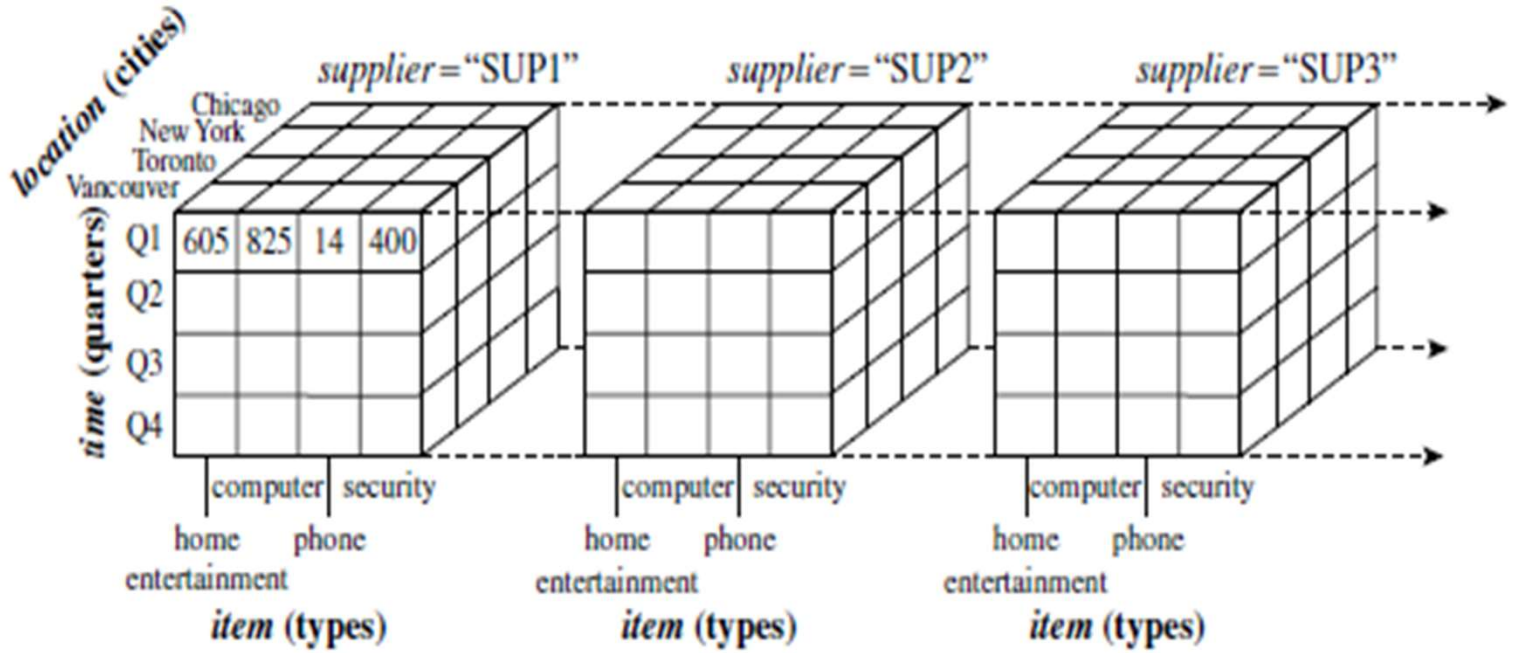


Figure 4.4 A 4-D data cube representation of sales data, according to *time*, *item*, *location*, and *supplier*. The measure displayed is *dollars_sold* (in thousands). For improved readability, only some of the cube values are shown.

If we continue in this way, we may display any n-dimensional data as a series of (n-1)-dimensional "cubes."

- The data cube is a metaphor for multidimensional data storage. The actual physical storage of such data may differ from its logical representation.

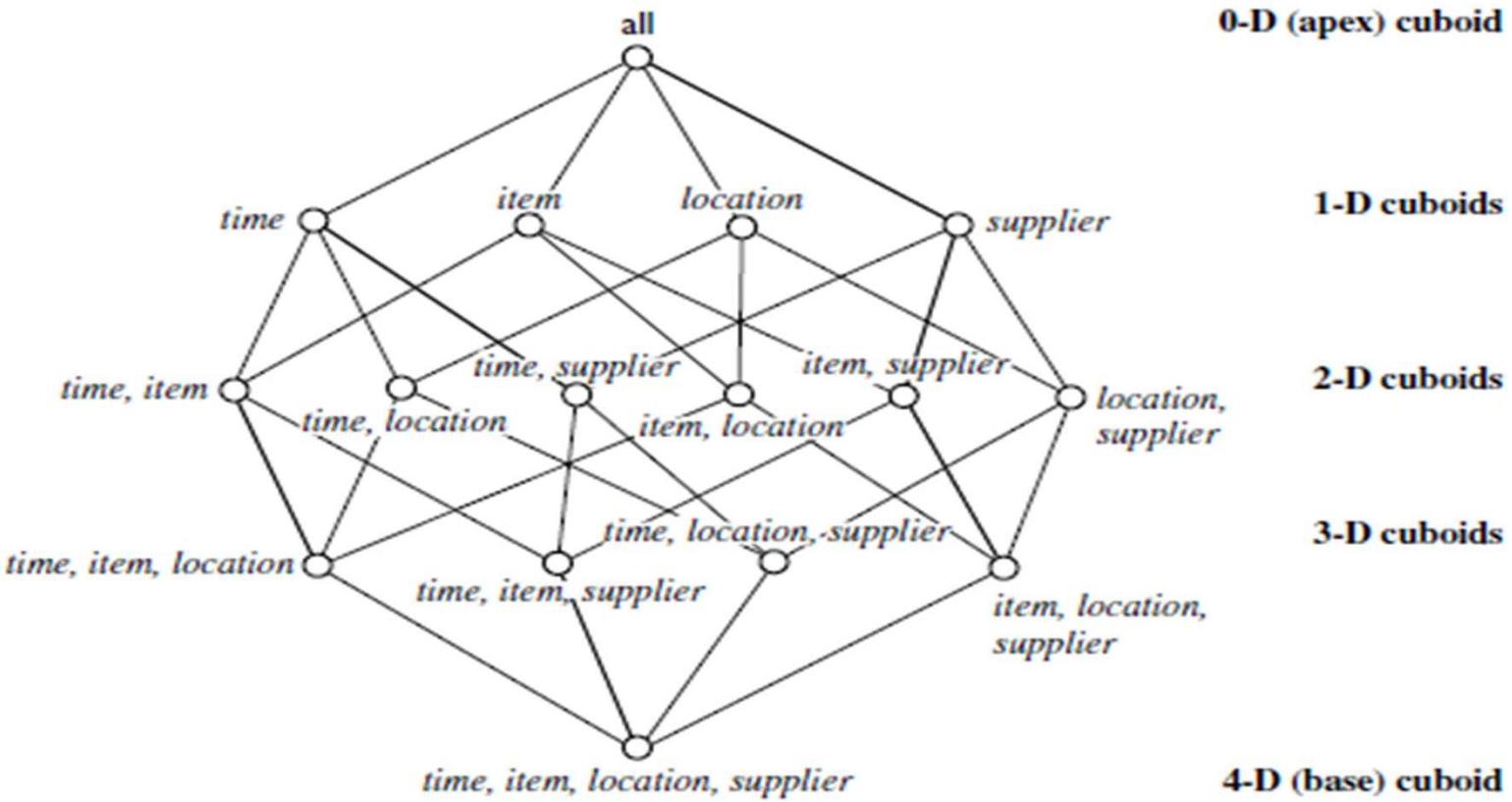
- The important thing to remember is that data cubes are n-dimensional and do not confine data to 3-D.

- Given a set of dimensions, we can generate a cuboid for each of the possible subsets of the given dimensions. The result would form a lattice of cuboids, each showing the data at a different level of summarization, or group-by.

- The n-D cuboid that holds the lowest level of summarization is called the base cuboid(Least generalized i.e most specific).

- The 0-D cuboid, which holds the highest level of summarization, is called the apex cuboid(Most generalized i.e least specific).

Lattice of Cuboids



Lattice of cuboids, making up a 4-D data cube for *time*, *item*, *location*, and *supplier*. Each cuboid represents a different degree of summarization.

Metadata Repository

When used in a data warehouse, metadata are the data that define warehouse objects.

A metadata repository should contain the following:

- A description of the data warehouse structure, which includes the warehouse schema, view, dimensions, hierarchies, and derived data definitions, as well as data mart locations and contents

- Operational metadata, which include data lineage (history of migrated data and the sequence of transformations applied to it), currency of data (active, archived, or purged), and monitoring information (warehouse usage statistics, error reports, and audit trails).

- The algorithms used for summarization, which include measure and dimension definition algorithms, data on granularity, partitions, subject areas, aggregation, summarization, and predefined queries and reports.

Metadata Repository

- Mapping from the operational environment to the data warehouse, which includes source databases and their contents, gateway descriptions, data partitions, data extraction, cleaning, transformation rules and defaults, data refresh and purging rules, and security (user authorization and access control).
- Data related to system performance, which include indices and profiles that improve data access and retrieval performance, in addition to rules for the timing and scheduling of refresh, update, and replication cycles.
- Business metadata, which include business terms and definitions, data ownership information, and charging policies.

Why Metadata is Important??

- Metadata are used as a directory to help the decision support system analyst locate the contents of the data warehouse, and as a guide to the data mapping when data are transformed from the operational environment to the data warehouse environment.

- Metadata also serve as a guide to the algorithms used for summarization between the current detailed data and the lightly summarized data, and between the lightly summarized data and the highly summarized data.

- Metadata should be stored and managed persistently (i.e., on disk).

Dimensional Modelling

- **Dimensional Modeling (DM)** is a logical design technique optimized for data storage in a Data warehouse.The purpose of dimensional modeling is to optimize the database for faster retrieval of data.
- **The concept of Dimensional Modelling consists of "fact" and "dimension" tables**.
- **A dimensional model contains same information as ER model but packages the data in a symmetric format whose design goals are easy understandability, query performance, and resilience to change. Why ER is not suitable for DWH?**
- **A dimensional model in data warehouse is designed to read, summarize, analyze numeric information like values, balances, counts, weights, etc. in a data warehouse. In contrast, relation models are optimized for addition, updating and deletion of data in a real-time Online Transaction System.**
- Dimensional models are used in data warehouse systems and not a good fit for relational systems.

Dimensional Modelling

- **ER modelling aims to optimize performance for transaction processing. It is also hard to query ER models because of the complexity; Therefore ER models are not suitable for high performance retrieval of data.**

- Data warehouse contain huge information. Data can't be fetched by normal technique so it requires special techniques.
- **Dimensional Modeling (DM)** is a logical design technique optimized for data storage in a Data warehouse.The purpose of dimensional modeling is to optimize the database for faster retrieval of data.
- **A dimensional model contains same information as ER model but packages the data in a symmetric format whose design goals are easy understandability, query performance, and resilience to change.**

- A dimensional model in data warehouse is designed to read, summarize, analyze numeric information like values, balances, counts, weights, etc. in a data warehouse.
  In contrast, relation models are optimized for addition, updating and deletion of data in a real-time OLTP.

Dimensional Modelling

- ER models are not suitable for high performance retrieval of data.
- **Dimensional Modeling (DM)** is a logical design technique optimized for data storage in a Data warehouse.
- The purpose of dimensional modeling is **to optimize the database for faster retrieval of data**.
- A dimensional model contains same information as ER model but packages the data in a symmetric format whose design goals are easy understandability, query performance, and resilience to change.
- A dimensional model in data warehouse is designed to read, summarize, analyze numeric information (like values, balances, counts, weights, etc.) in a data warehouse.
  In contrast, relation models are optimized for addition, updating and deletion of data in a real-time OLTP.

Elements of Dimensional Modelling

Fact

- Facts are the measurements/metrics from your business process.
  —For a Sales business process, a measurement would be quarterly sales number

Dimension

- A category of information. For example, the time dimension.
- In simple terms, they give who, what, where of a fact.
  –E.g.In the Sales business process, for the fact quarterly sales number, dimensions would be

  Who - Customer Names
  Where - Location
  What - Product Name
  When - Time Dimension
- In other words, a dimension is a window to view information in the facts.

AttributesThe Attributes are the various characteristics of the dimension
—E.g. In the Location dimension, the attributes can be State, Country , Zipcode etc.

- Attributes are used to search, filter, or classify facts. Dimension Tables contain Attributes.

Elements of Dimensional Modelling

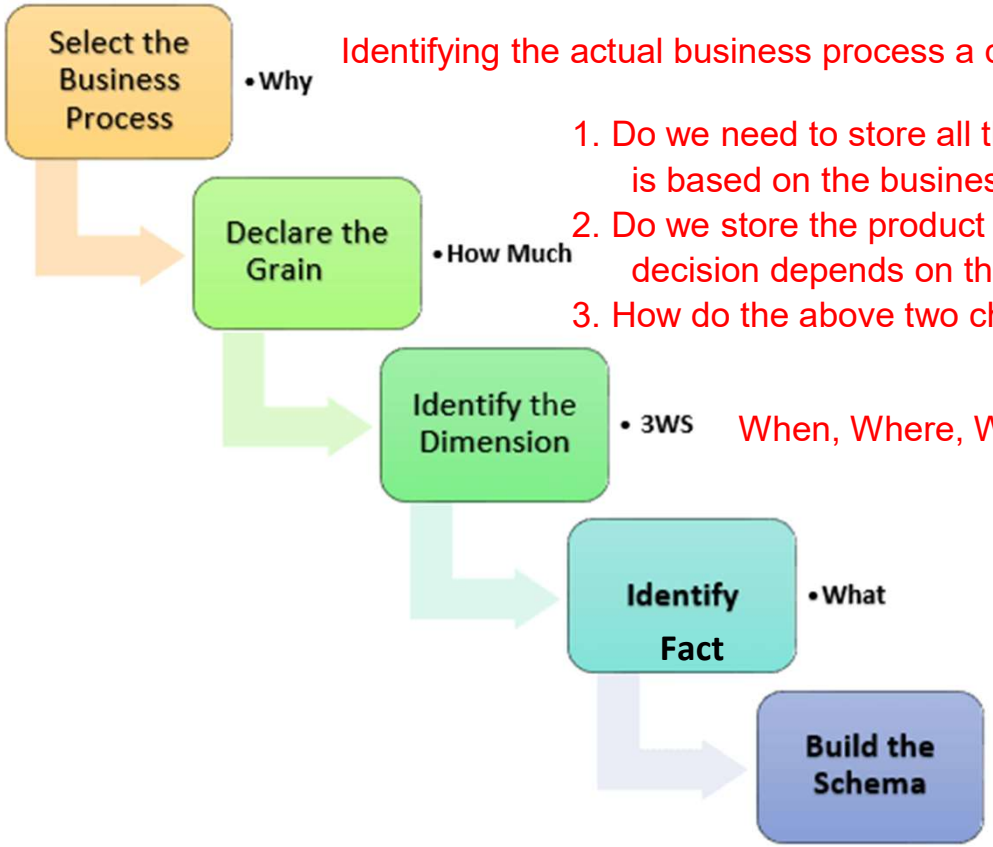
Fact Table

- A fact table is a primary table in dimension modelling.
- Fact table consists of the measurements, metrics or facts of a business process.
  Eg. Monthly sales volume, Average Customer Balance etc...
- A Fact Table contains
  1. Measurements/facts
  2. Foreign key to dimension table

Dimension Table

- A dimension table contains dimensions of a fact.
- They are joined to fact table via a foreign key.
- Dimension tables are denormalized tables.
- The Dimension Attributes are the various columns in a dimension table
- Dimensions offers descriptive characteristics of the facts with the help of their attributes
- No limit set for number of dimensions
- The dimension can also contain one or more hierarchical relationships

Steps of Dimensional Modelling



Identifying the actual business process a data warehouse should cover.

1. Do we need to store all the available products or just a few types of products? This decision is based on the business processes selected for DWH
2. Do we store the product sale information on a monthly, weekly, daily or hourly basis? This decision depends on the nature of reports requested by executives
3. How do the above two choices affect the database size?

When, Where, Who

Benefits of Dimensional Modeling

- Standardization of dimensions allows easy reporting across areas of the business.
- Dimension tables store the history of the dimensional information.
- It allows to introduce entirely new dimension without major disruptions to the fact table.
- Dimensional models also store data in such a fashion that it is easier to retrieve the information from the data once the data is stored in the database.
- Compared to the normalized model dimensional table are easier to understand.
- Information is grouped into clear and simple business categories.
- The dimensional model is very understandable by the business. This model is based on business terms, so that the business knows what each fact, dimension, or attribute means.
- Dimensional models are deformalized and optimized for fast data querying. Many relational database platforms recognize this model and optimize query execution plans to aid in performance.
- Dimensional modelling in data warehouse creates a schema which is optimized for high performance. It means fewer joins and helps with minimized data redundancy.
- The dimensional model also helps to boost query performance. It is more denormalized therefore it is optimized for querying.
- Dimensional models can comfortably accommodate change. Dimension tables can have more columns added to them without affecting existing business intelligence applications using these tables.

Types of DWH Schema

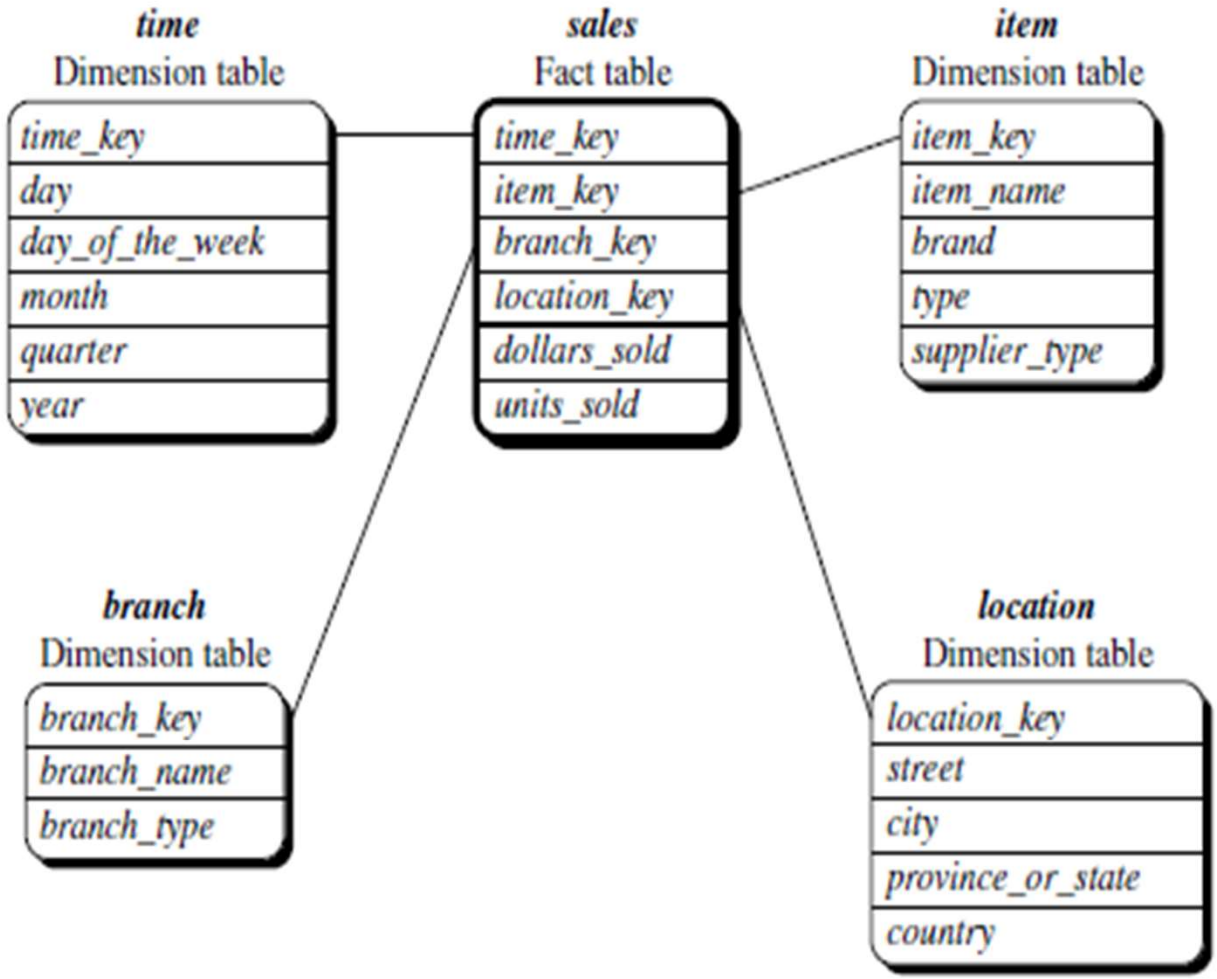
- Star Schema
- Snowflake Schema

The star schema and snowflake schema are two different ways of organizing data warehouses.

Both schemas use dimension tables that describe the information contained within a fact table

Star Schema

- In the **star schema design, the fact table sits in the middle and is connected to dimension lookup tables like a star.**
- Each dimension is represented as a single table.
- The primary key in each dimension table is related to a foreign key in the fact table.
- All measures in the fact table are related to all the dimensions that fact table is related to. In other words, they all have the same level of granularity.
- A star schema can be simple or complex. A simple star consists of one fact table; a complex star can have more than one fact table



Star schema of *sales* data warehouse.

- A star schema for AllElectronics sales is shown in Figure .
- Sales are considered along four dimensions: time, item, branch, and location.
- The schema contains a central fact table for sales that contains keys to each of the four dimensions, along with two measures: dollars sold and units sold.
- To minimize the size of the fact table,dimension identifiers (e.g., time key and item key) are system-generated identifiers.

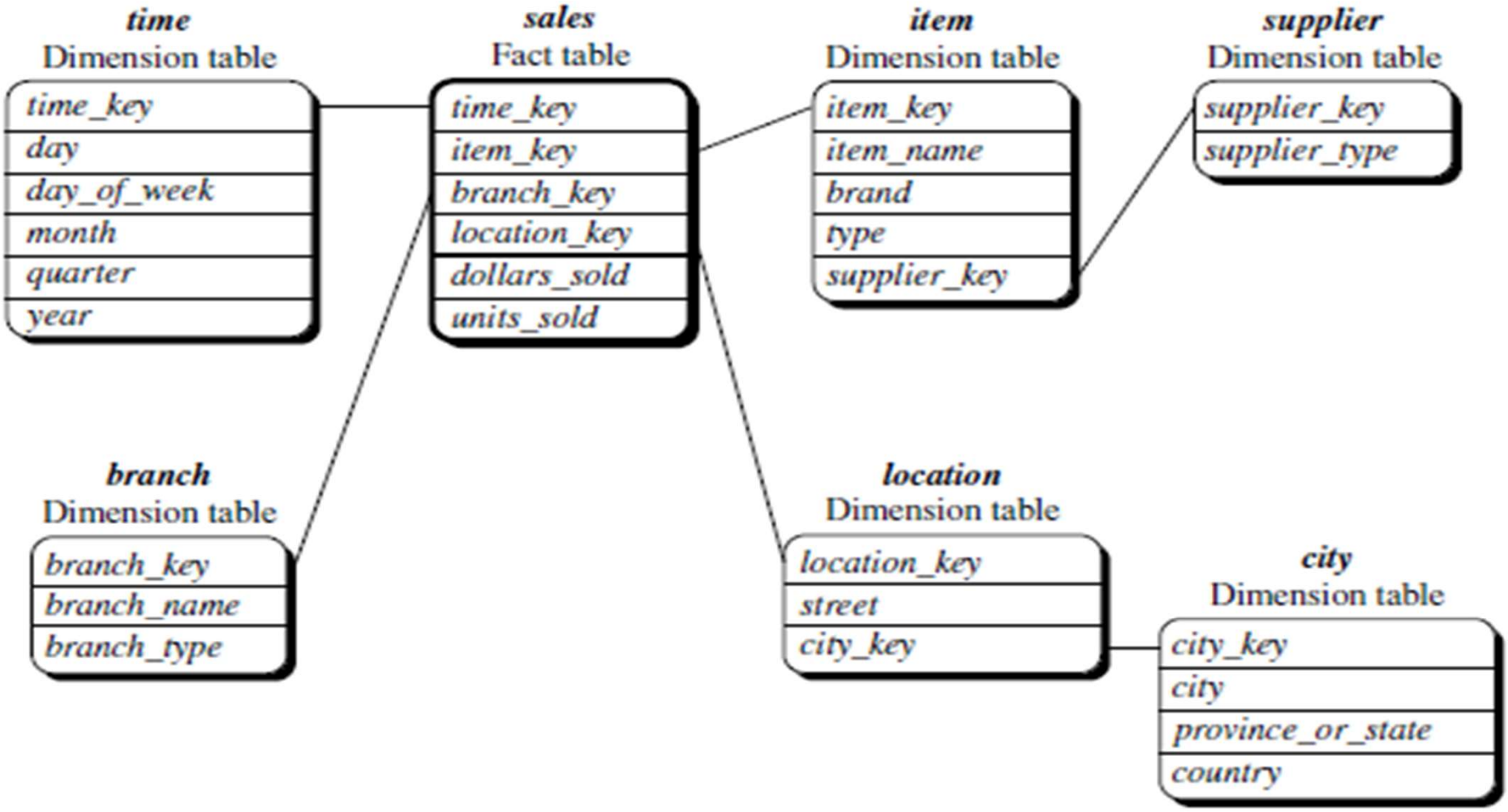
Characteristics of Star Schema

- Every dimension in a star schema is represented with the only one-dimension table.
- The dimension table should contain the set of attributes.
- The dimension table is joined to the fact table using a foreign key
- The dimension tables are not joined to each other
- Fact table would contain keys and measures
- The Star schema is easy to understand and provides optimal disk usage.
- The dimension tables are not normalized.
- The schema is widely supported by BI Tools

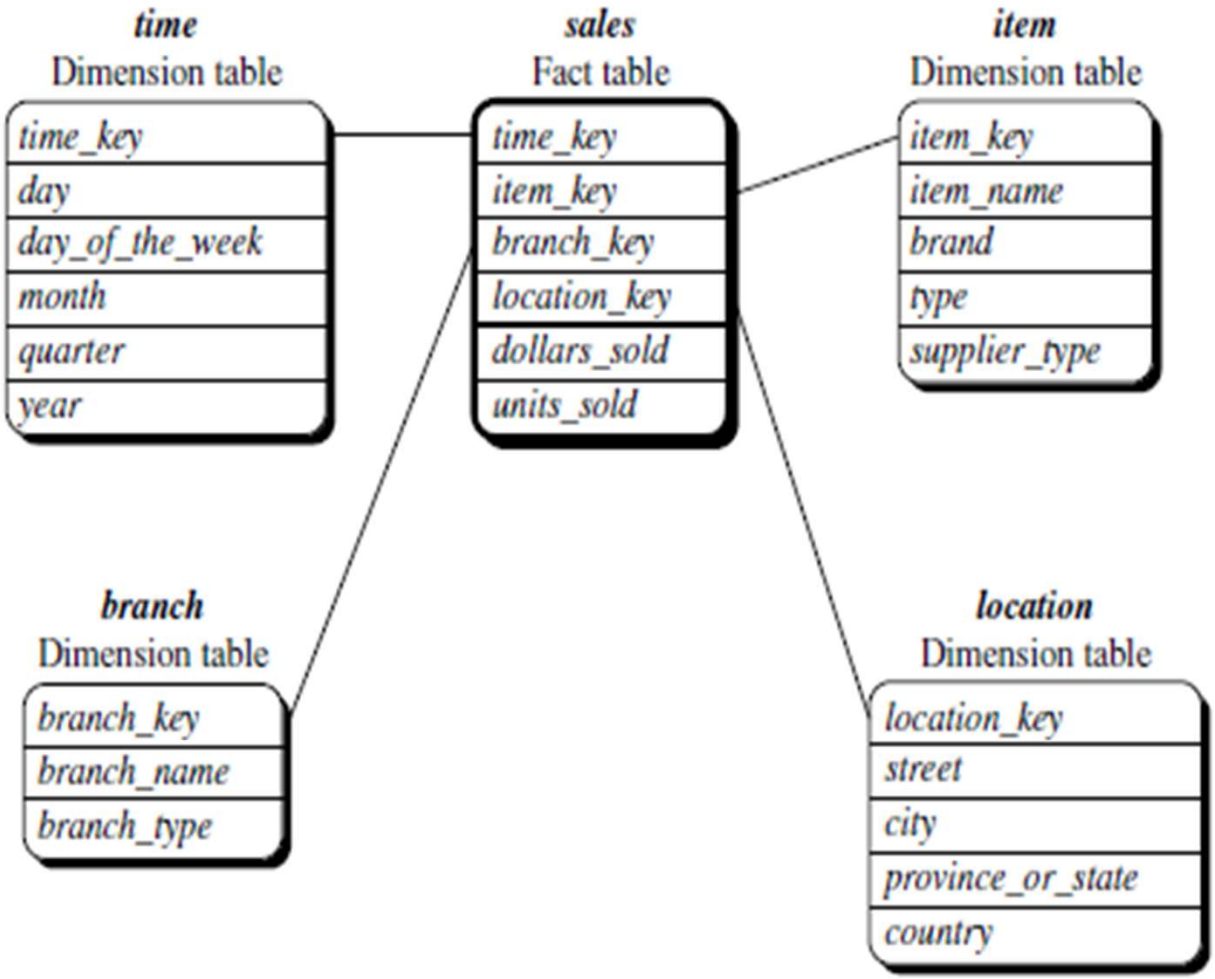
Snowflake Schema

- The **snowflake schema is an extension of the star schema which have multiple levels of dimension tables.**
- Snowflaking a dimension means normalizing it and making it more manageable by reducing its size.
- Dimension tables are normalized which splits data into additional tables.
  - –Adv: This reduces redundancies.Such a table is easy to maintain and saves storage space.
  - –Disadv: Normalizing creates more dimension tables with multiple joins and reduces data integrity issues. However, querying is more challenging using the snowflake schema, because queries need to dig deeper to access the relevant data.

    Also the snowflake structure can reduce the effectiveness of browsing, since more joins will be needed to execute a query. Consequently, the system performance may be adversely impacted.
- Hence, although the snowflake schema reduces redundancy, it is not as popular as the star schema in data warehouse design.



Snowflake schema of a *sales* data warehouse.

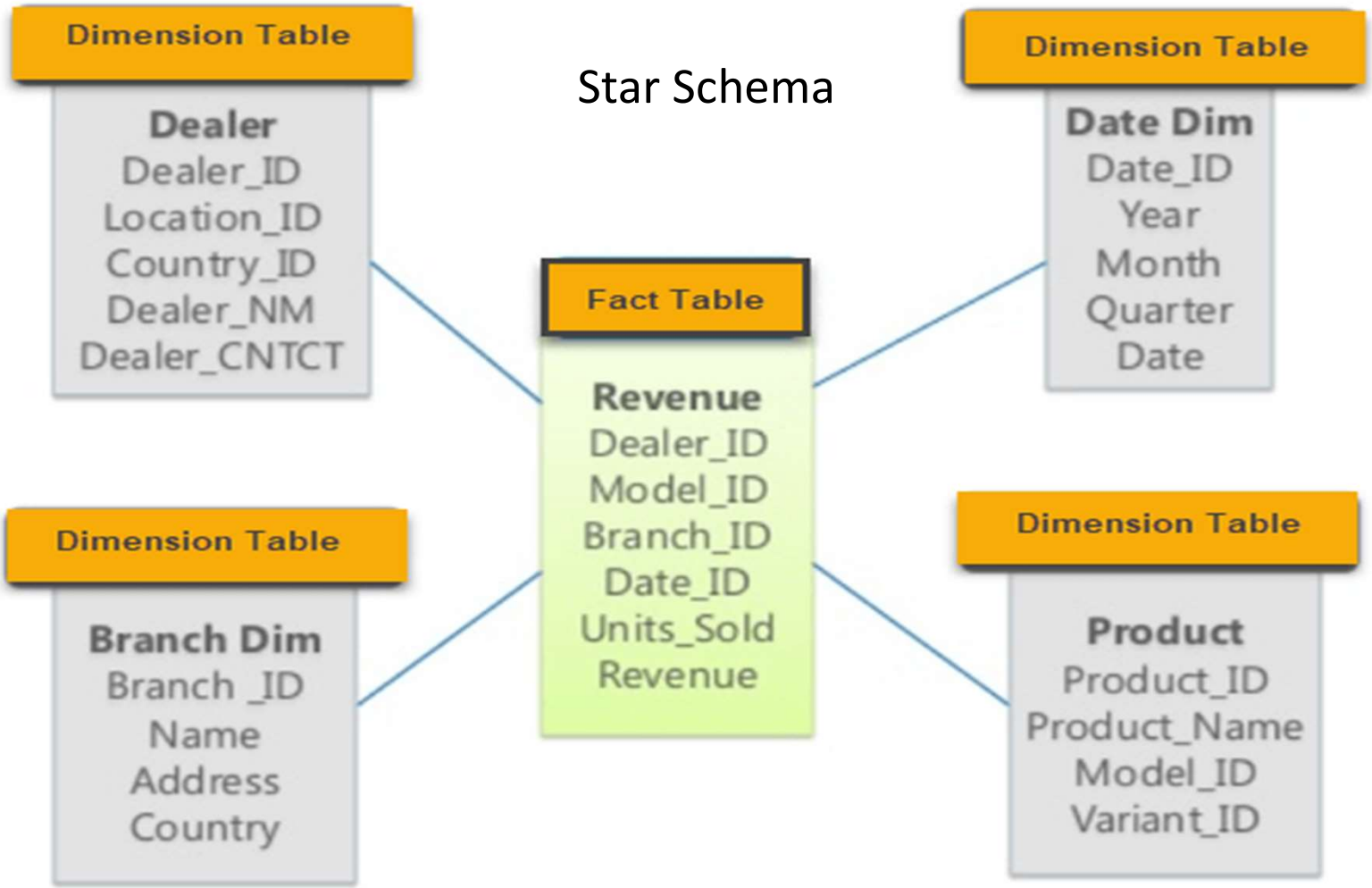


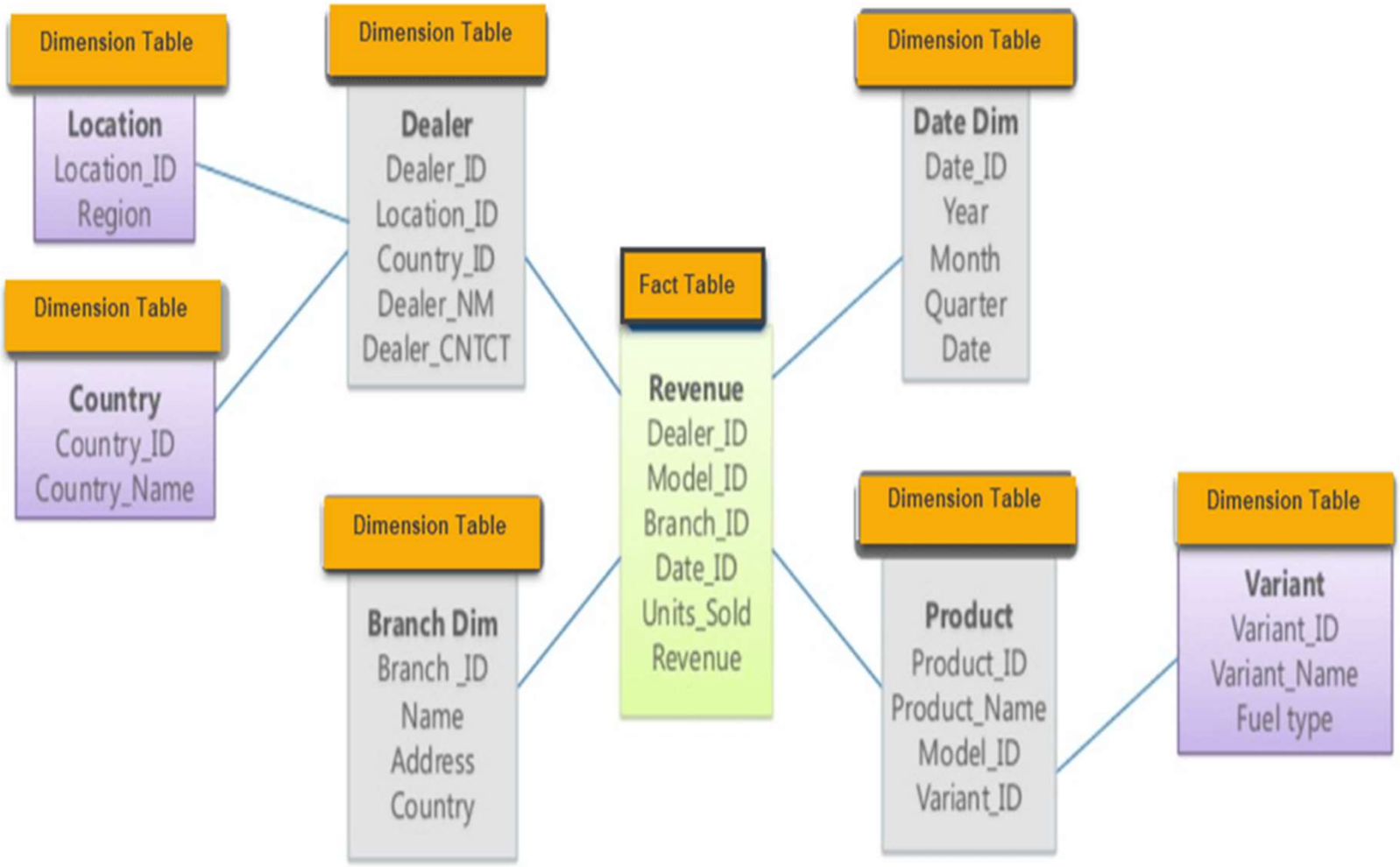
Star schema of *sales* data warehouse.

Characteristics of Snowflake Schema:

- The main benefit of the snowflake schema it uses smaller disk space.
- Easier to implement when a dimension is added to the Schema
- Due to multiple tables query performance is reduced
- The primary challenge that you will face while using the snowflake Schema is that you need to perform more maintenance efforts because of the more lookup tables.

Star Schema


Dimension Table
Dealer
Dealer_ID
Location_ID
Country_ID
Dealer_NM
Dealer_CNTCT
Dimension Table
Date Dim
Date_ID
Year
Month
Quarter
Date
Fact Table
Revenue
Dealer_ID
Model_ID
Branch_ID
Date_ID
Units_Sold
Revenue
Dimension Table
Branch Dim
Branch _ID
Name
Address
Country
Dimension Table
Product
Product_ID
Product_Name
Model_ID
Variant_ID



- Country is further normalized into an individual table

Snowflake Schema

The Mumbai university wants you to help design a star schema to record grades for course completed by students. There are four dimensional tables namely course_section, professor, student, period with attributes as follows :

Course_section Attributes: Course_Id, Section_number, Course_name, Units, Room_id, Roomcapacity. During a given semester the college offers an average of 500 course sections

Professor Attributes: Prof_id, Prof_Name, Title, Department_id, department_name

Student Attributes: Student_id, Student_name, Major. Each Course section has an average of 60 students

Period Attributes: Semester_id, Year. The database will contain Data for 30 months periods. The only fact that is to be recorded in the fact table is course Grade

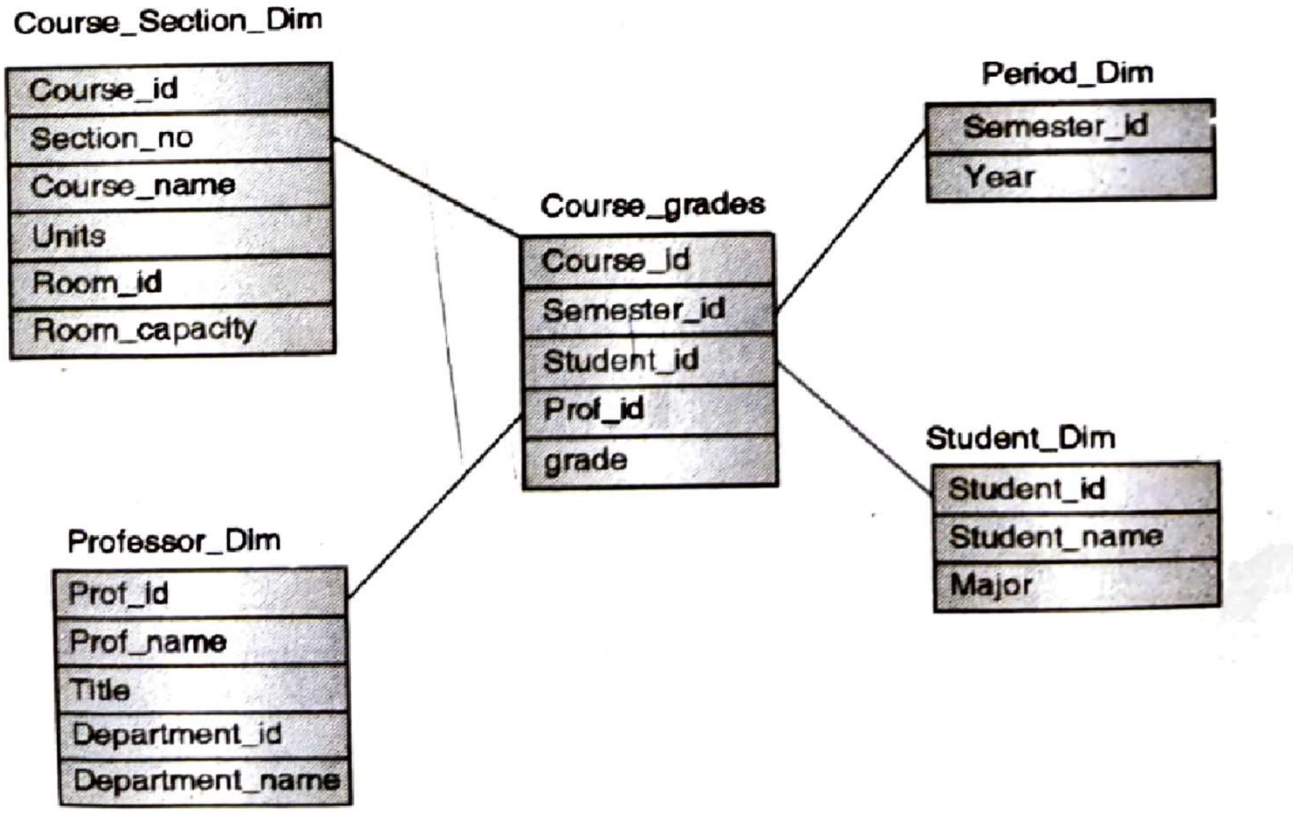
Answer the following Questions

(a) Design the star schema for this problem

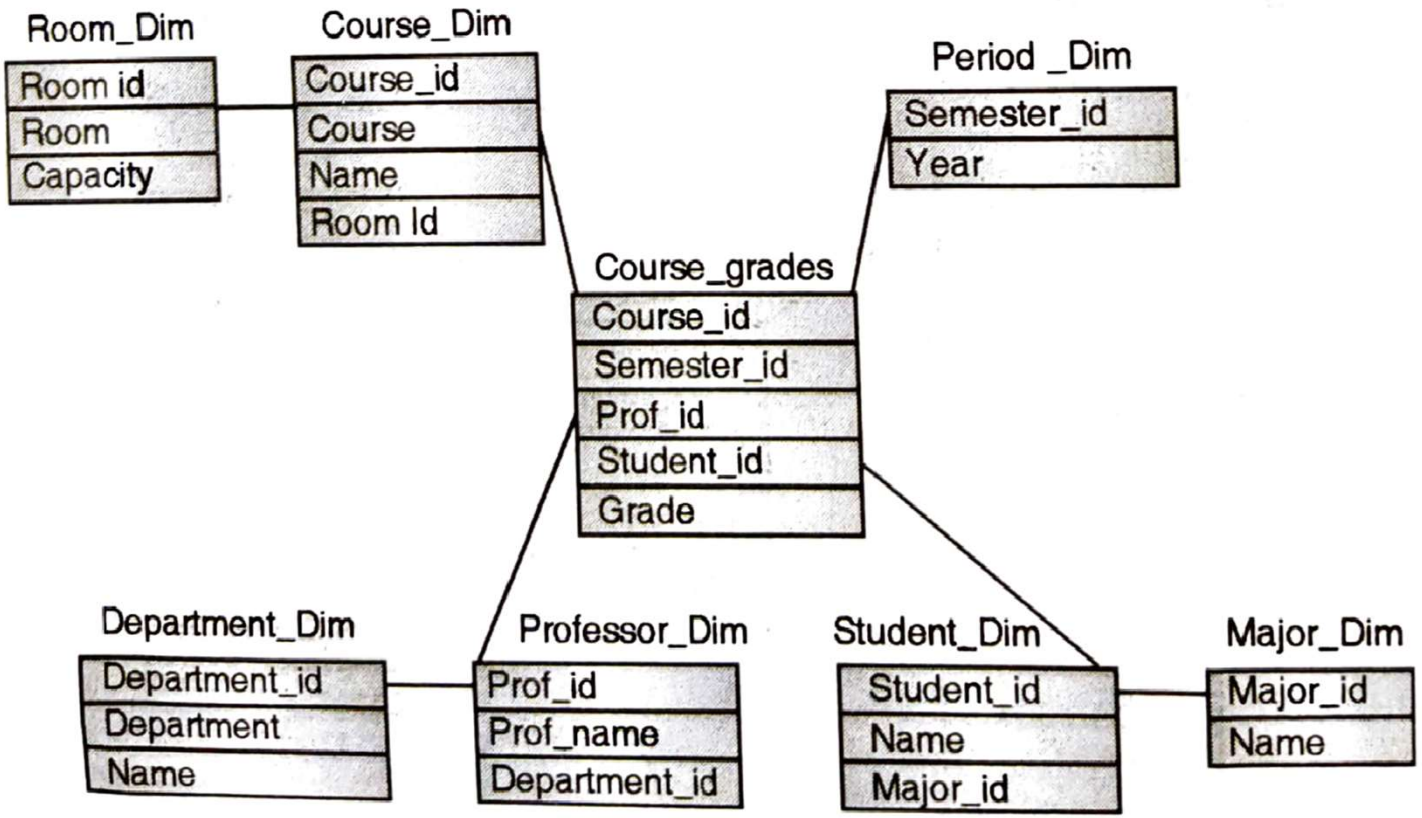
(b) Estimate the number of rows in the fact table, using the assumptions stated above and also estimate the total size of the fact table (in bytes) assuming that each field has an average of 5 bytes.

(c) Can you convert this star schema to a snowflake schema ? Justify your answer and design a snowflake schema if it is possible.

- Star Schema


Course_Section_Dim
Course_id
Section_no
Course_name
Units
Room_id
Room_capacity
Period_Dim
Semester_id
Year
Course_grades
Course_id
Semester_id
Student_id
Prof_id
grade
Student_Dim
Student_id
Student_name
Major
Professor_Dim
Prof_id
Prof_name
Title
Department_id
Department_name

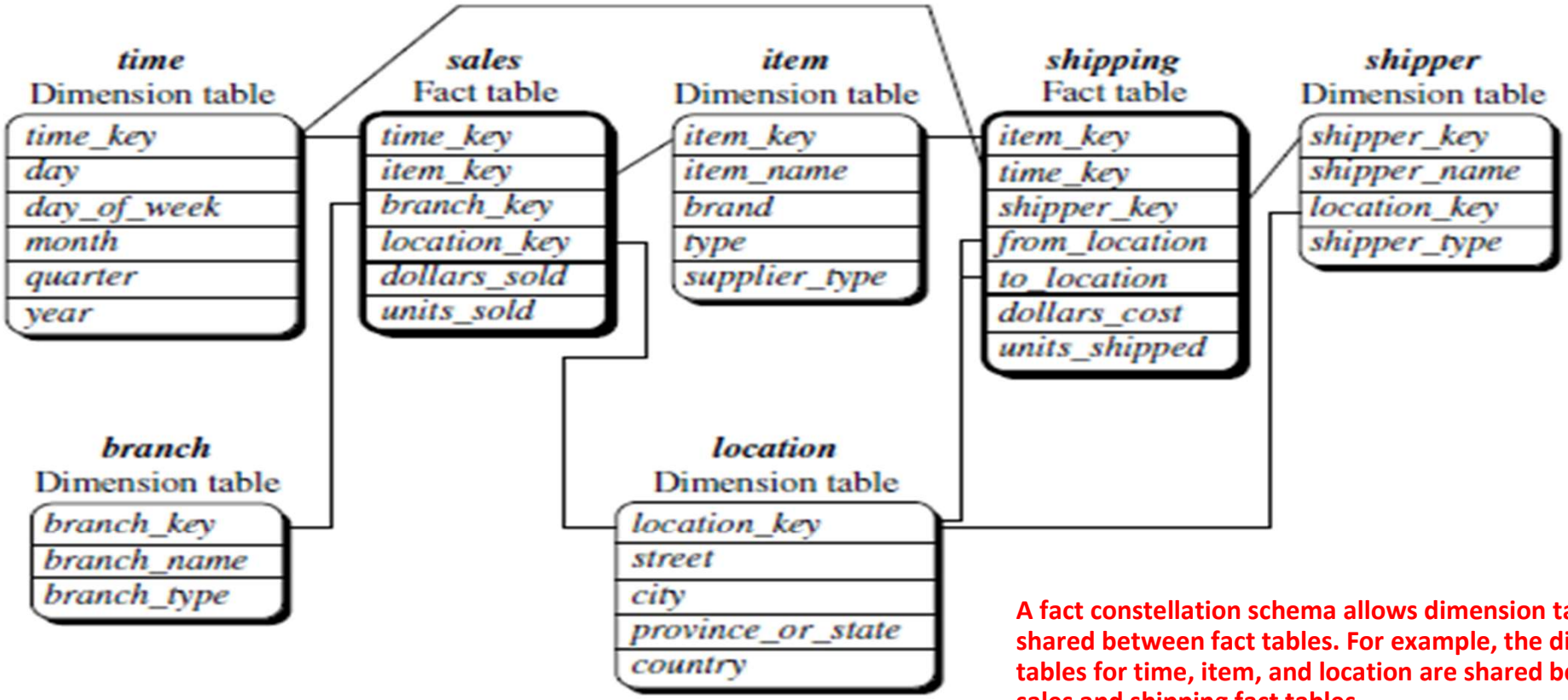
- Snowflake Schema



Star Vs Snowflake Schema

Star Schema	Snowflake Schema
Star schema is a **top-down model.**	While it is a **bottom-up model.**
It contains a fact table surrounded by dimension tables.	One fact table surrounded by dimension table which are in turn surrounded by dimension table
In a star schema, only single join creates the relationship between the fact table and any dimension tables.	A snowflake schema requires many joins to fetch the data.
Simple DB Design.	Very Complex DB Design.
Denormalized Data structure and query also run faster.	Normalized Data Structure.
It takes less time for the execution of queries.	While it takes more time than star schema for the execution of queries.
Single Dimension table contains aggregated data.	Data Split into different Dimension Tables.
It has less number of foreign keys.	While it has more number of foreign keys.
Offers higher performing queries using Star Join Query Optimization. **Tables may be connected with multiple**	The Snowflake schema is represented by centralized fact table which is unlikely connected with multiple dimensions.

Fact constellation: Sophisticated applications may require multiple fact tables to share dimension tables. This kind of schema can be viewed as a collection of stars, and hence is called a galaxy schema or a fact constellation.



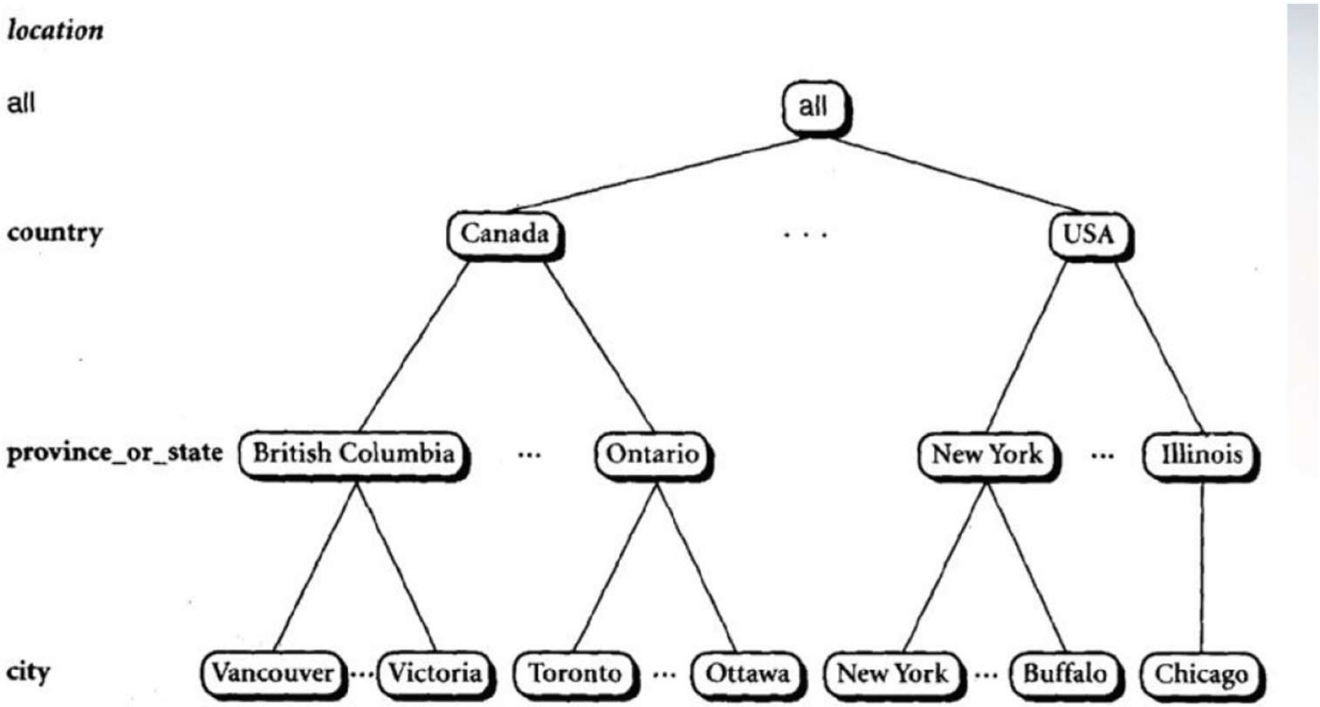
A fact constellation schema allows dimension tables to be shared between fact tables. For example, the dimensions tables for time, item, and location are shared between the sales and shipping fact tables.

Fact constellation schema of a sales and shipping data warehouse.

- In data warehousing, there is a distinction between a data warehouse and a data mart.
- A data warehouse collects information about subjects that span the entire organization,such as customers, items, sales, assets, and personnel, and thus its scope is enterprise-wide.
- For data warehouses, the fact constellation schema is commonly used, since it can model multiple, interrelated subjects. A data mart, on the other hand, is a department subset of the data warehouse that focuses on selected subjects, and thus its scope is departmentwide.
- For data marts, the star or snowflake schema is commonly used, since both are geared toward modeling single subjects, although the star schema is more popular and efficient.

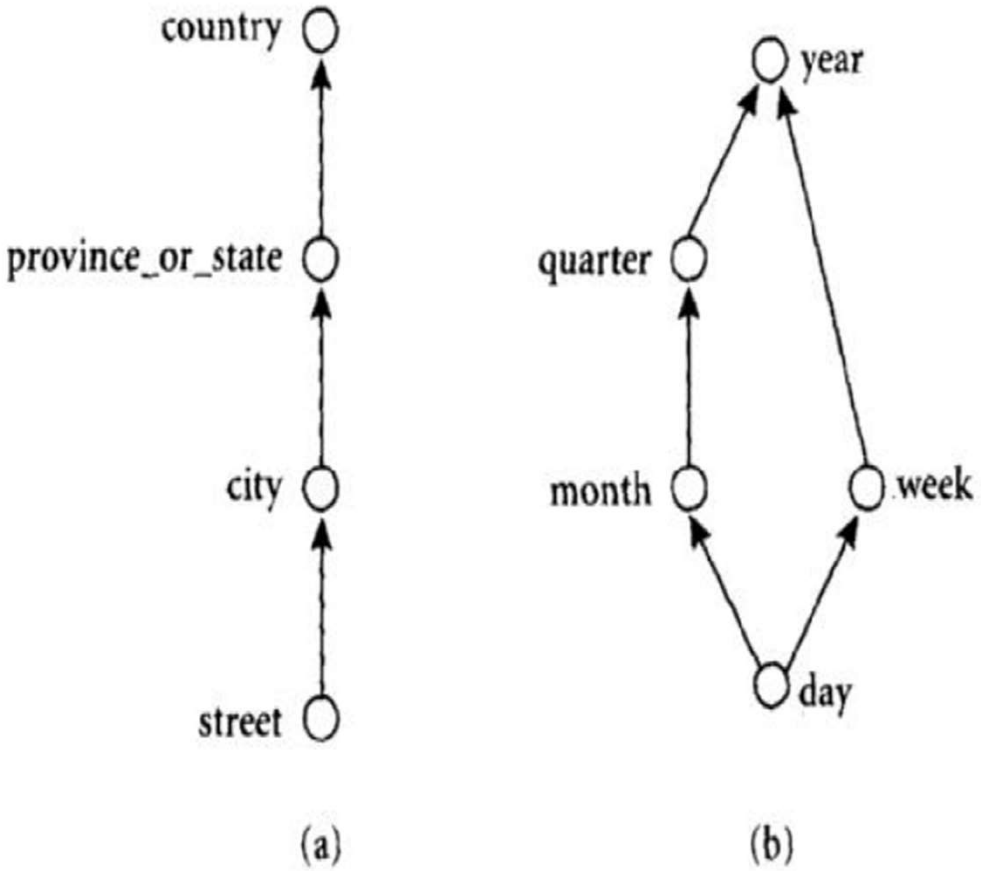
Concept Hierarchy

It is a sequence of mappings from a set of low-level concepts to higher-level, more general concepts

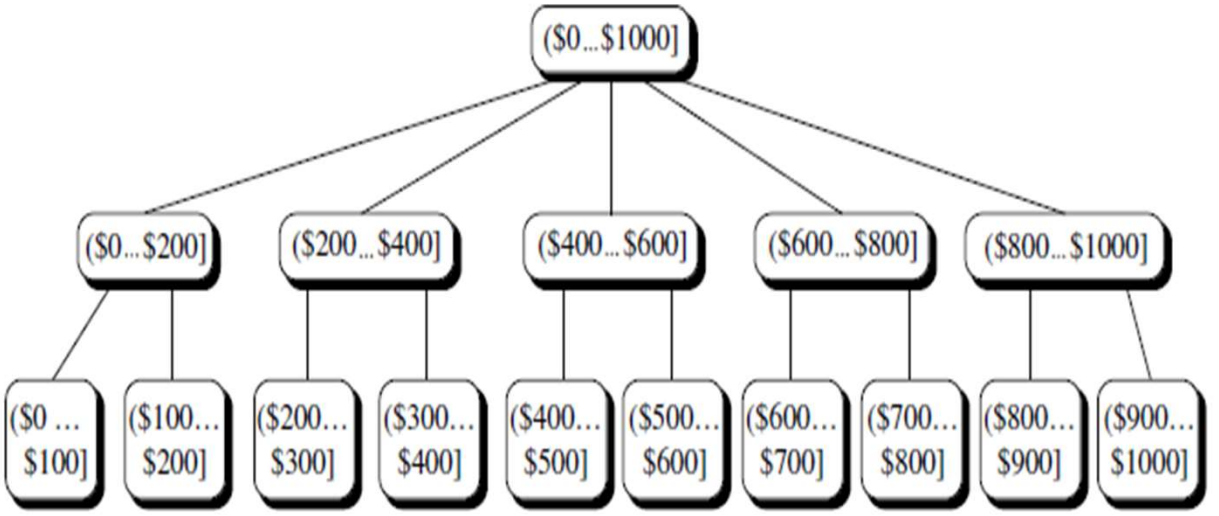


Concept Hierarchy

- A Concept Hierarchy may also be a total order or partial order among attributes in a database schema
- It may also be defined by discretizing or grouping values for a given dimension or attribute, resulting in a **set-grouping** hierarchy
- Concept Hierarchies may be provided manually by
  - System users
  - Domain Experts
  - Knowledge Engineers
  - Automated Statistical Analysis



Set Grouping Hierarchy Example



A concept hierarchy for *price*.

Basic Analytical Operations of OLAP for multidimensional data

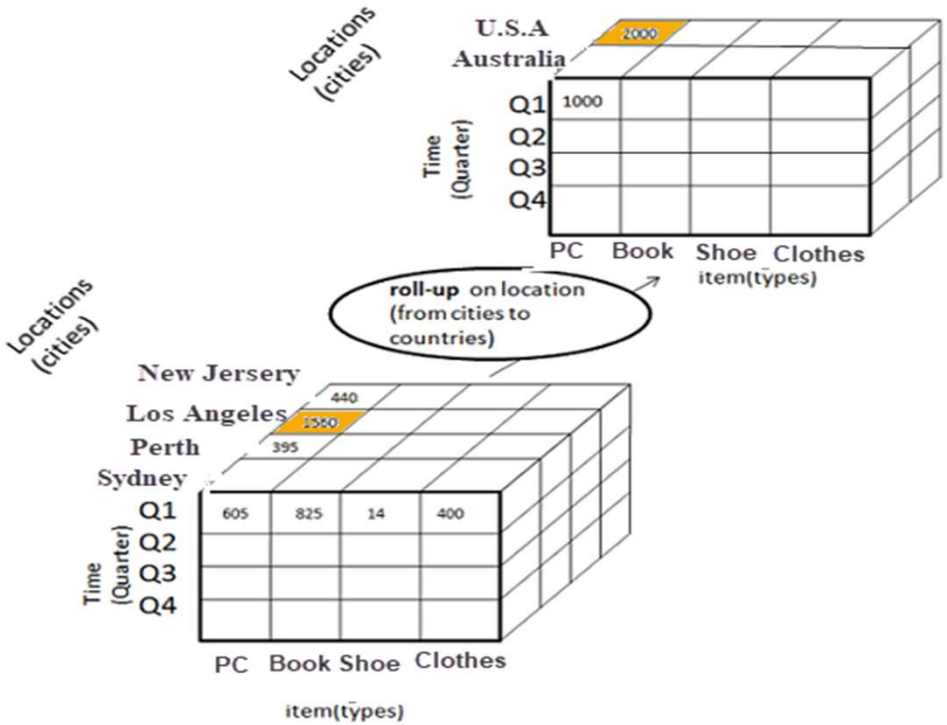
- In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies.
- Thus organization provides users with the flexibility to view data from different perspectives.
- A number of OLAP data cube operations exist to materialize these different views, allowing interactive querying and analysis of the data at hand.
- Hence, OLAP provides a user-friendly environment for interactive data analysis.

1. **Roll-up**
2. **Drill-down**
3. **Slice and dice**
4. **Pivot(rotate)**

Basic Analytical Operations of OLAP

1. Roll-up :

- Roll-up is also known as "**consolidation" or "aggregation**." The Roll-up operation can be performed in 2 way
  - Reducing dimensions
  - Climbing up concept hierarchy.
- In the roll-up process at least one or more dimensions need to be removed.

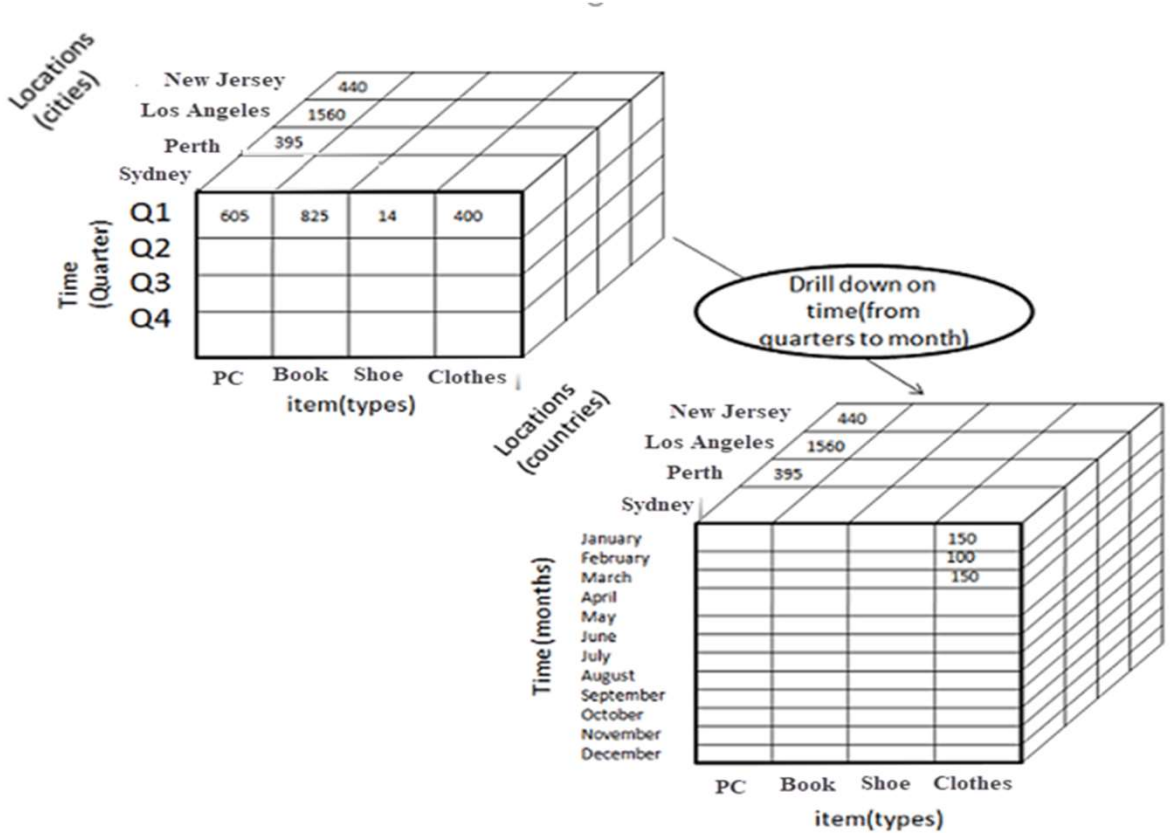


- In this example, cities New jersey and Los Angels are rolled up into country USA
- The sales figure of New Jersey and Los Angels are 440 and 1560 respectively. They become 2000 after roll-up
- In this aggregation process, data in location hierarchy moves up from city to the country.
- In this example, Cities dimension is removed.

Basic Analytical Operations of OLAP

2. Drill-down

- In drill-down, **data is fragmented into smaller parts**. It is the opposite of the rollup process. It can be done via
  - Moving down the concept hierarchy
  - Increasing a dimension



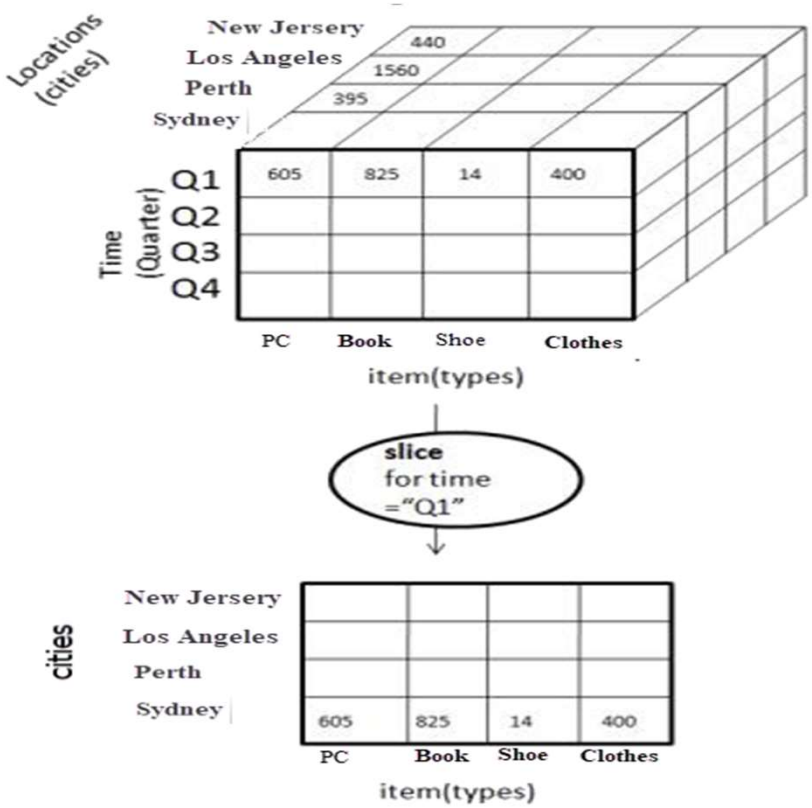
In this Example,

- Quarter Q1 is drilled down to months
- Here dimension Months is added.

Basic Analytical Operations of OLAP

3. Slice

- Here, one dimension is selected, and **a new sub-cube is created.**



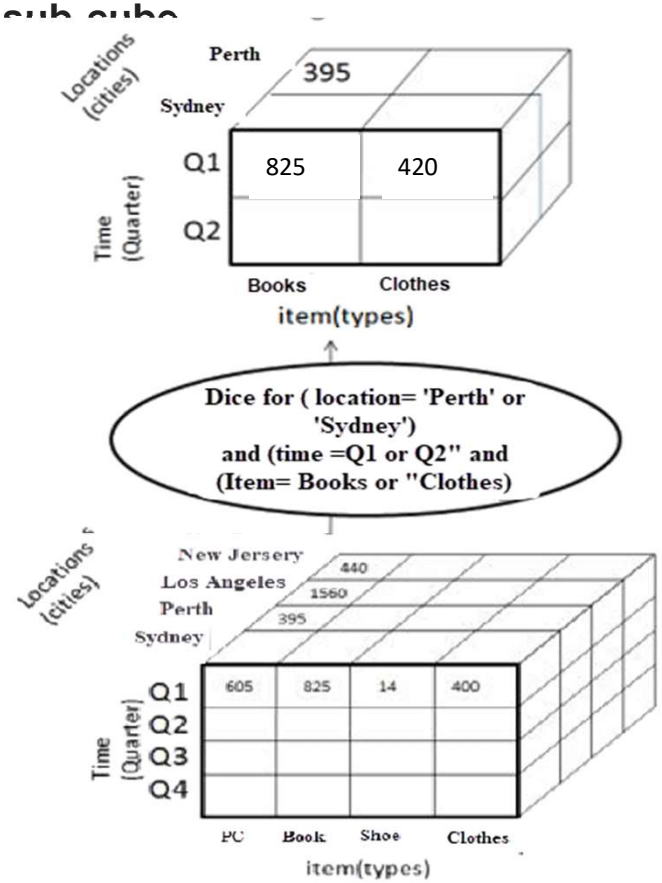
In this Example,

- Dimension Time is sliced with Q1 as the filter.
- A new cube is created altogether.

Basic Analytical Operations of OLAP

4. Dice

- This operation is similar to a slice. The difference in dice is you select **2 or more dimensions that result in the creation of a sub-cube**

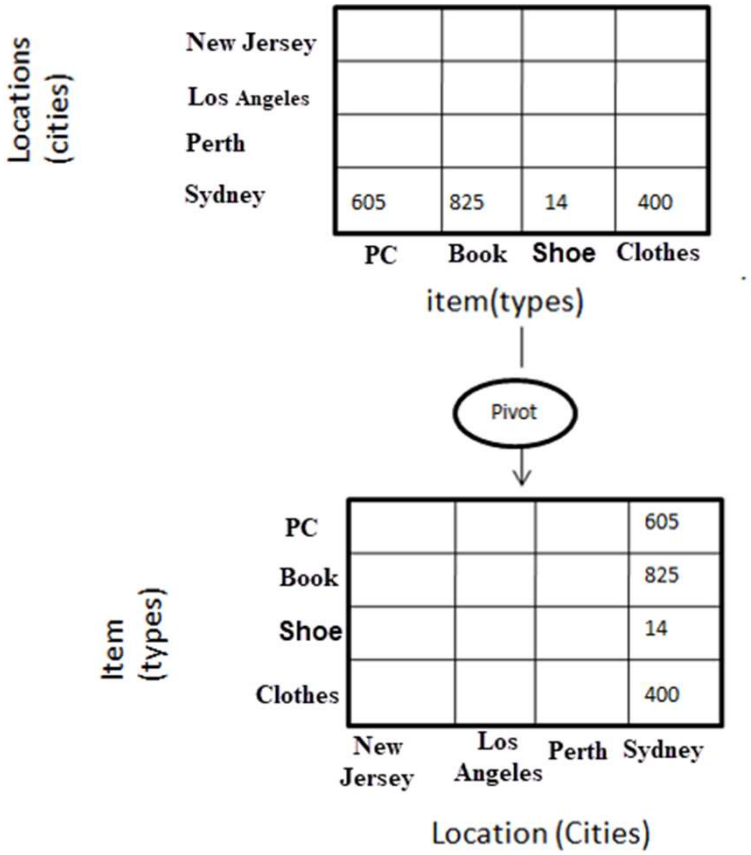


Dice operation in OLAP

Basic Analytical Operations of OLAP

5. Pivot

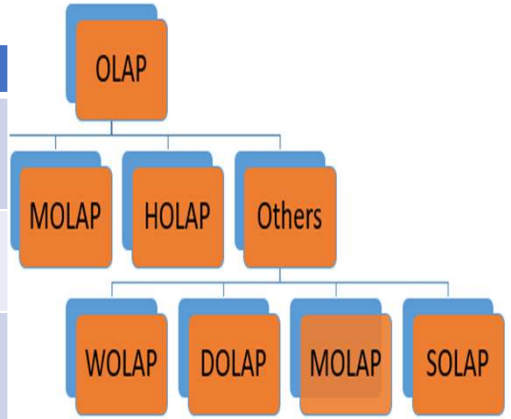
- In Pivot, you **rotate the data axes** to provide a substitute presentation of data.



Pivot operation in OLAP
Here the pivot is based on item types

Types of OLAP systems

Type of OLAP	Explanation
Relational OLAP(ROLAP):	ROLAP is an extended RDBMS along with multidimensional data mapping to perform the standard relational operation.
Multidimensional OLAP (MOLAP)	MOLAP Implementes operation in multidimensional data.
Hybrid OnlineAnalytical Processing (HOLAP)	In HOLAP approach the aggregated totals are stored in a multidimensional database while the detailed data is stored in the relational database. This offers both data efficiency of the ROLAP model and the performance of the MOLAP model.
Desktop OLAP (DOLAP)	In Desktop OLAP, a user downloads a part of the data from the database locally, or on their desktop and analyze it. DOLAP is relatively cheaper to deploy as it offers very few functionalities compares to other OLAP systems.
Web OLAP (WOLAP)	Web OLAP which is OLAP system accessible via the web browser. WOLAP is a three-tiered architecture. It consists of three components: client, middleware, and a database server.
Mobile OLAP:	Mobile OLAP helps users to access and analyze OLAP data using their mobile devices
Spatial OLAP :	SOLAP is created to facilitate management of both spatial and non-spatial data in a Geographic Information system (GIS)



Consider a data warehouse for a hospital where there are three dimension

(a) Doctor (b) Patient (c) Time

And two measures i) count ii) charge where charge is the fee that the doctor charges a patient for a visit.

Using the above example describe the following OLAP operations

1) Slice 2) Dice 3) Rollup 4) Drill down 5) Pivot

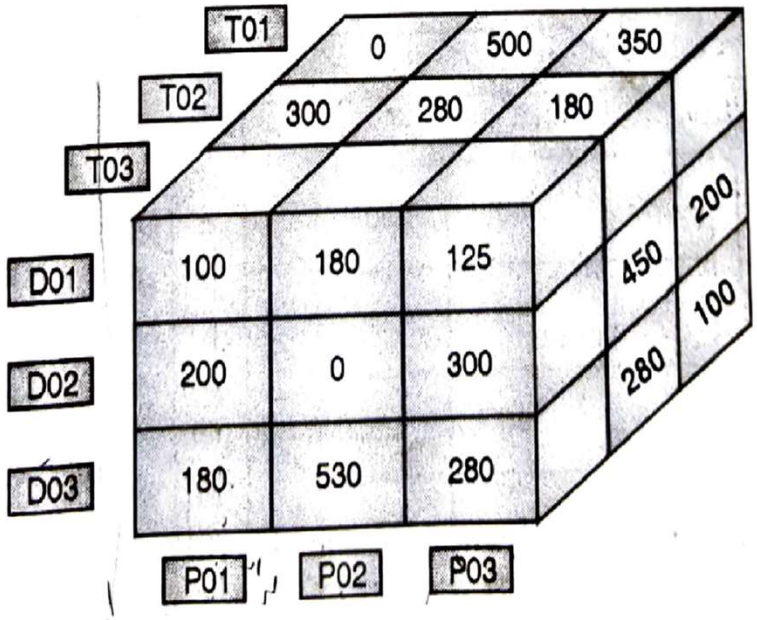
There are four tables, out of 3 dimension tables and 1 fact table

Dimension tables :

1. Doctor (DID, name, phone, location,pin,specialisation)
2. Patient (PID,name, phone , state, city,location, pin)
3. Time (TID,day, month, quarter , year)

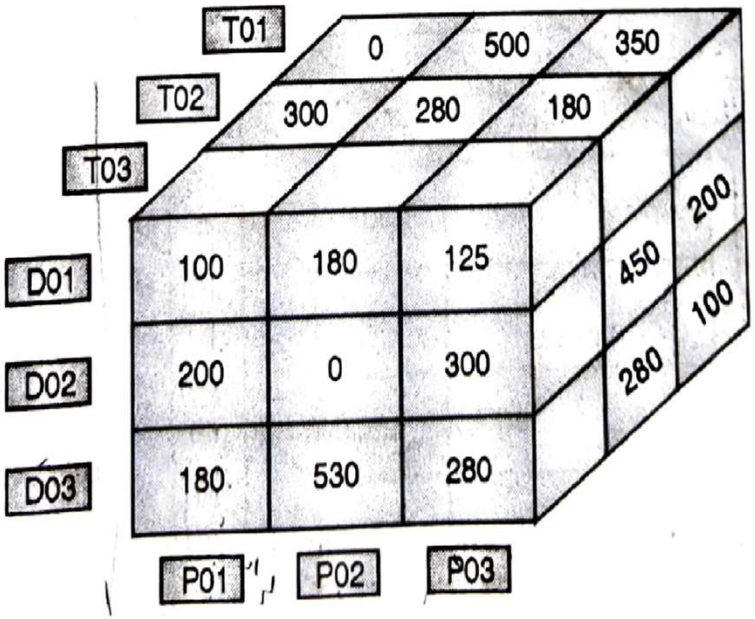
Fact Table :

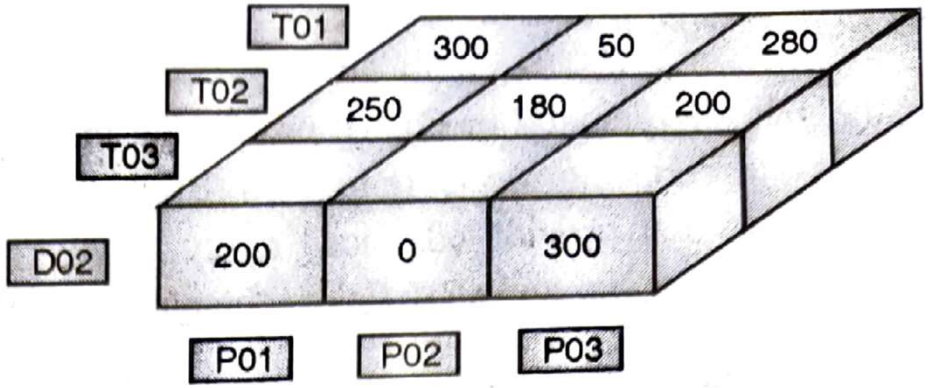
Fact_table (DID,PID,TID, count, charge)



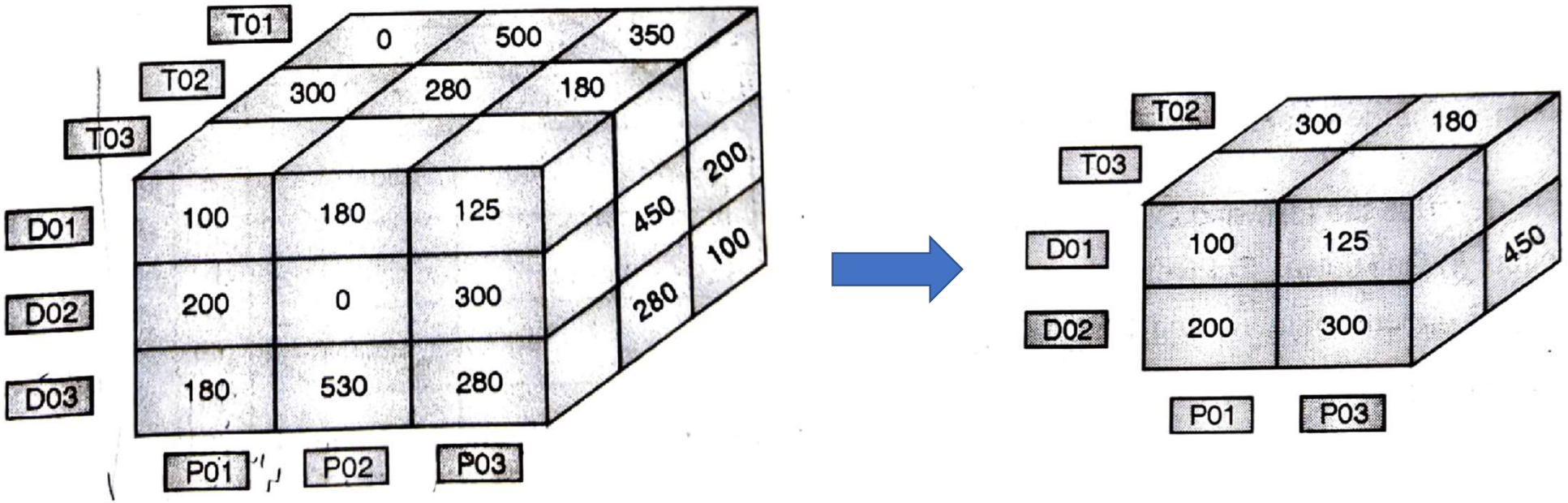
Operations :

1. **Slice :** Slice on fact table with DID = 2 , this cuts the cube at DID = 2 along the time and patient axis thus it will display a slice of cube, in which time on x and patient on y axis.

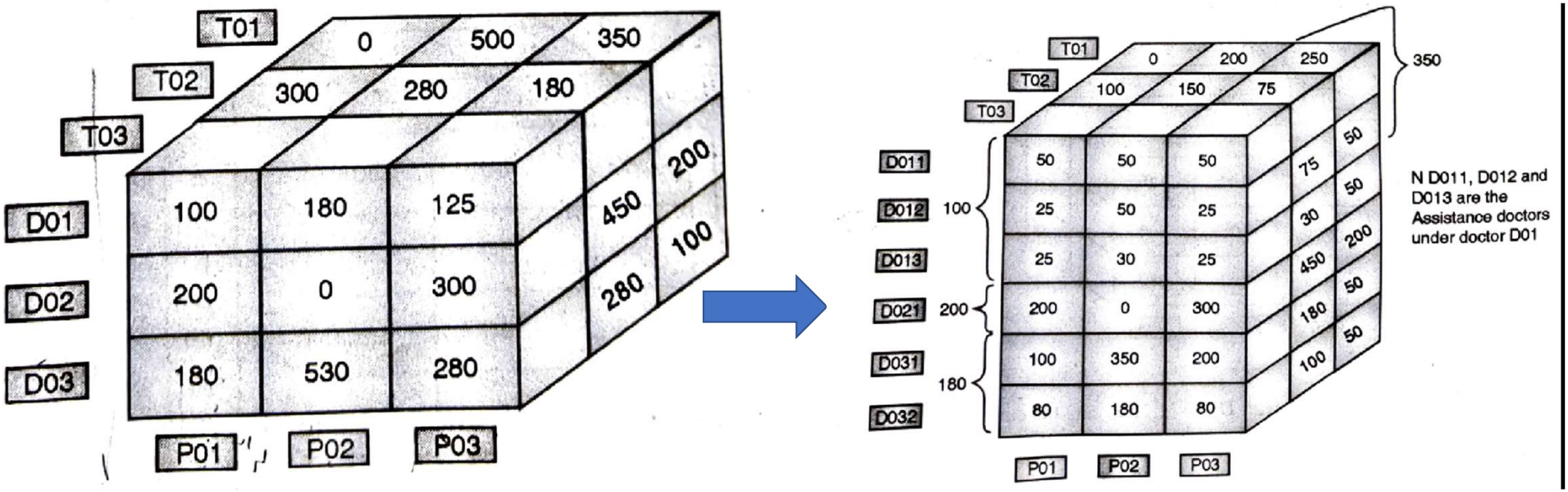




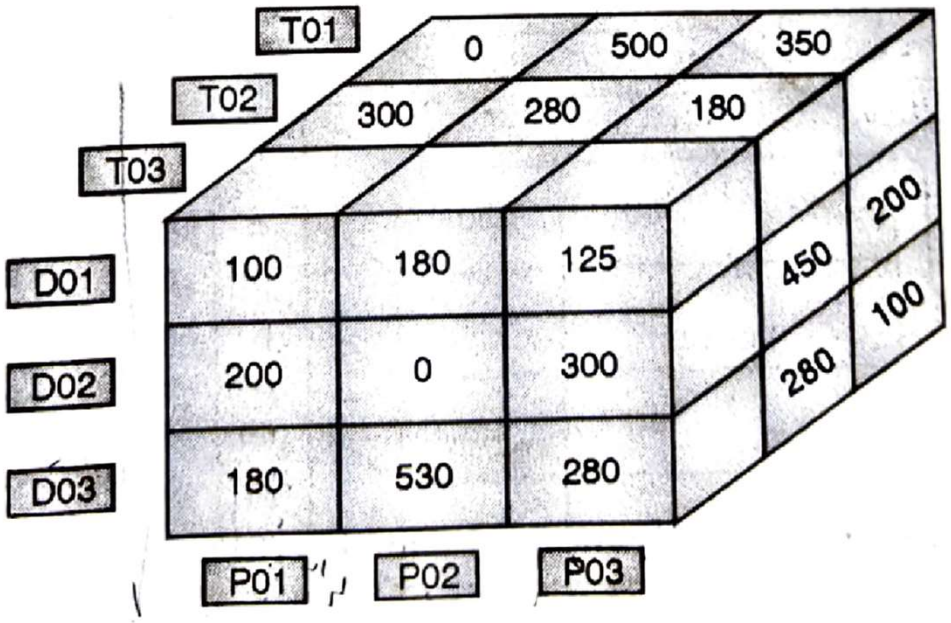
Dice : It is a sub cube of main cube. Thus it cuts the cube with more than predicate like dice on cube with DID = 2, and DID = 01 and PID = 01 PID = 03 and TID = 02, 03

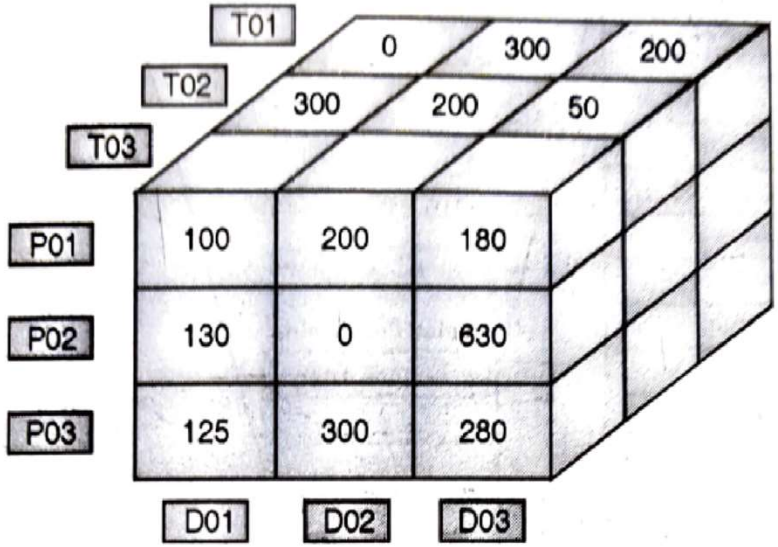


Roll up : it gives summary based on concept hierarchies. Assuming there e concept hierarchy in patient table as state->city->location. Then roll up summarise the charges or count in terms of city or further roll up will charges for a particular state etc.



Drill down : it is opposite to roll up that means if currently cube is summarised with respect to city then drill down will also show summarisation with respect to location



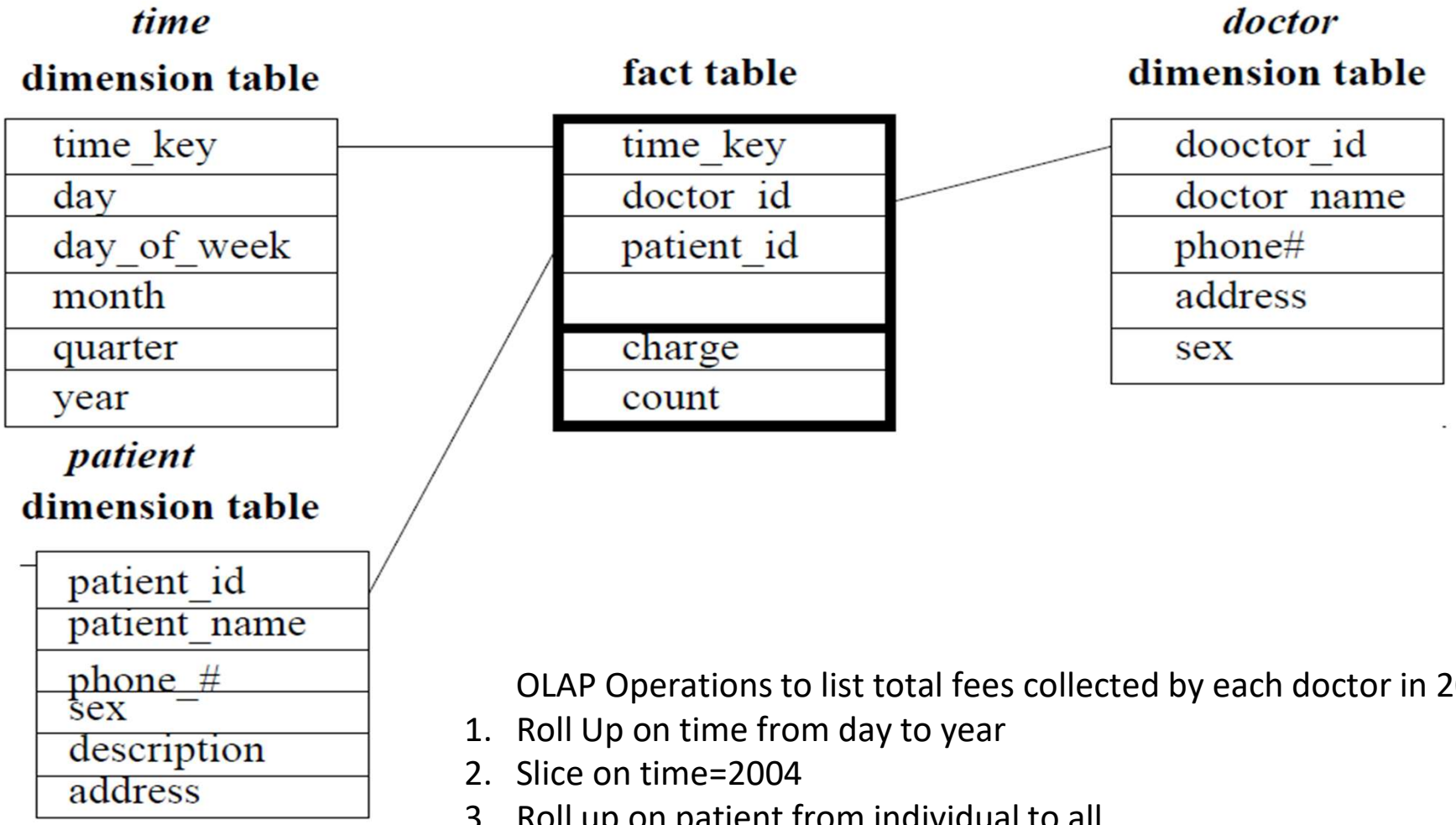


Suppose that a data warehouse consists of the three dimensions *time*, *doctor*, and *patient*, and the two measures *count* and *charge*, where *charge* is the fee that a doctor charges a patient for a visit.

(a) Enumerate three classes of schemas that are popularly used for modeling data warehouses.

(b) Draw a schema diagram for the above data warehouse using one of the schema classes listed in (a).

(c) Starting with the base cuboid [*day*, *doctor*, *patient*], what specific *OLAP operations* should be performed in order to list the total fee collected by each doctor in 2004?



OLAP Operations to list total fees collected by each doctor in 2004

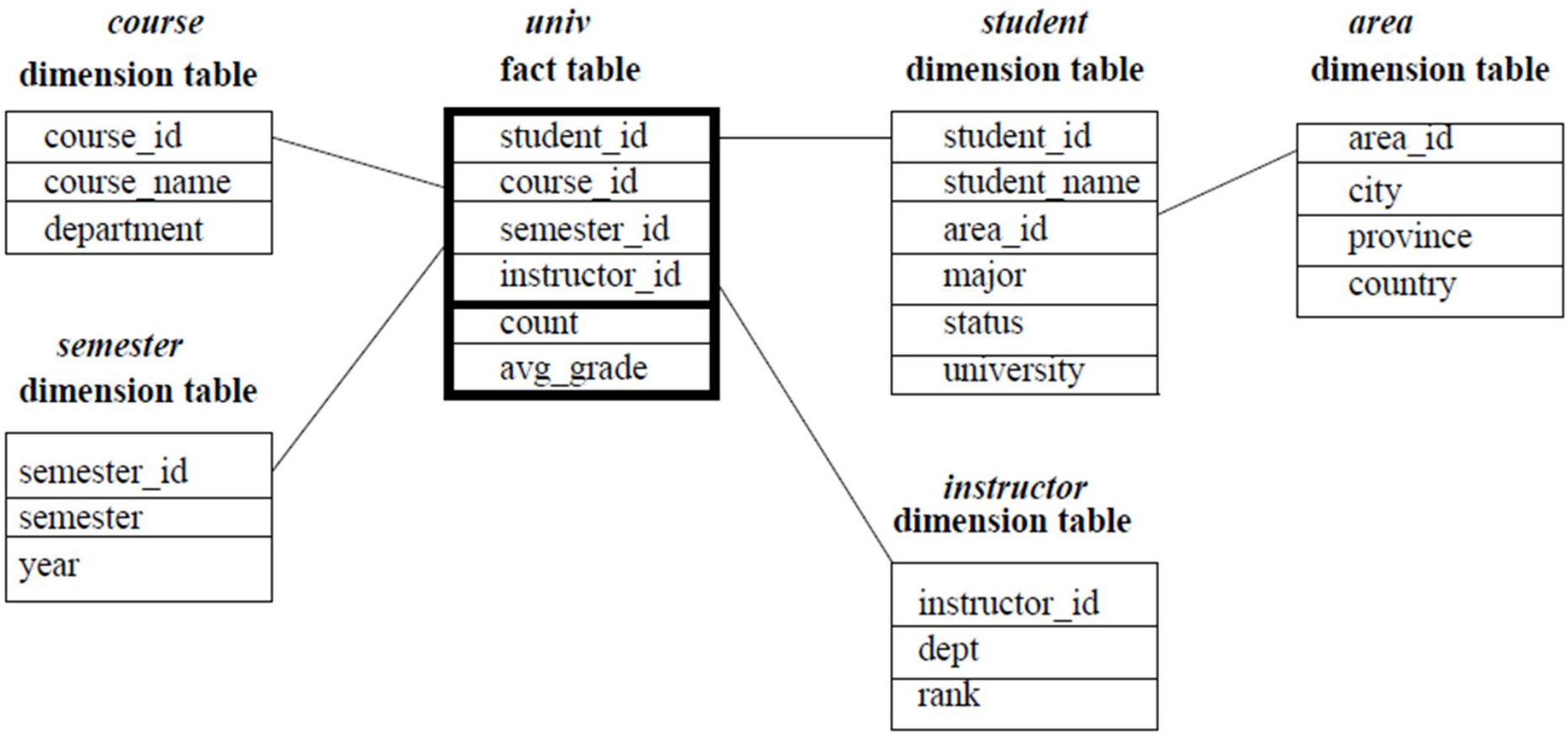
1. Roll Up on time from day to year
2. Slice on time=2004
3. Roll up on patient from individual to all.

Suppose that a data warehouse for *Big-University* consists of the following four dimensions: *student*, *course*, *semester*, and *instructor*, and two measures *count* and *avg_grade*. When at the lowest conceptual level (e.g., for a given student, course, semester, and instructor combination), the *avg_grade* measure stores the actual course grade of the student. At higher conceptual levels, *avg_grade* stores the average grade for the given combination.

(a) Draw a *snowflake schema* diagram for the data warehouse.

(b) Starting with the base cuboid [*student*, *course*, *semester*, *instructor*], what specific *OLAP operations* (e.g., roll-up from *semester* to *year*) should one perform in order to list the average grade of *CS* courses for each *Big-University* student.

(c) If each dimension has five levels (including all), such as "*student* < *major* < *status* < *university* < all", how many cuboids will this cube contain (including the base and apex cuboids)?



Starting with the base cuboid [student, course, semester, instructor], what specific OLAP operations (e.g., roll-up from semester to year) should one perform in order to list the average grade of CS courses for each Big-University student.

The specific OLAP operations to be performed are:

- Roll-up on course from course id to department.
- Roll-up on semester from semester id to all.
- Slice for course="CS" .

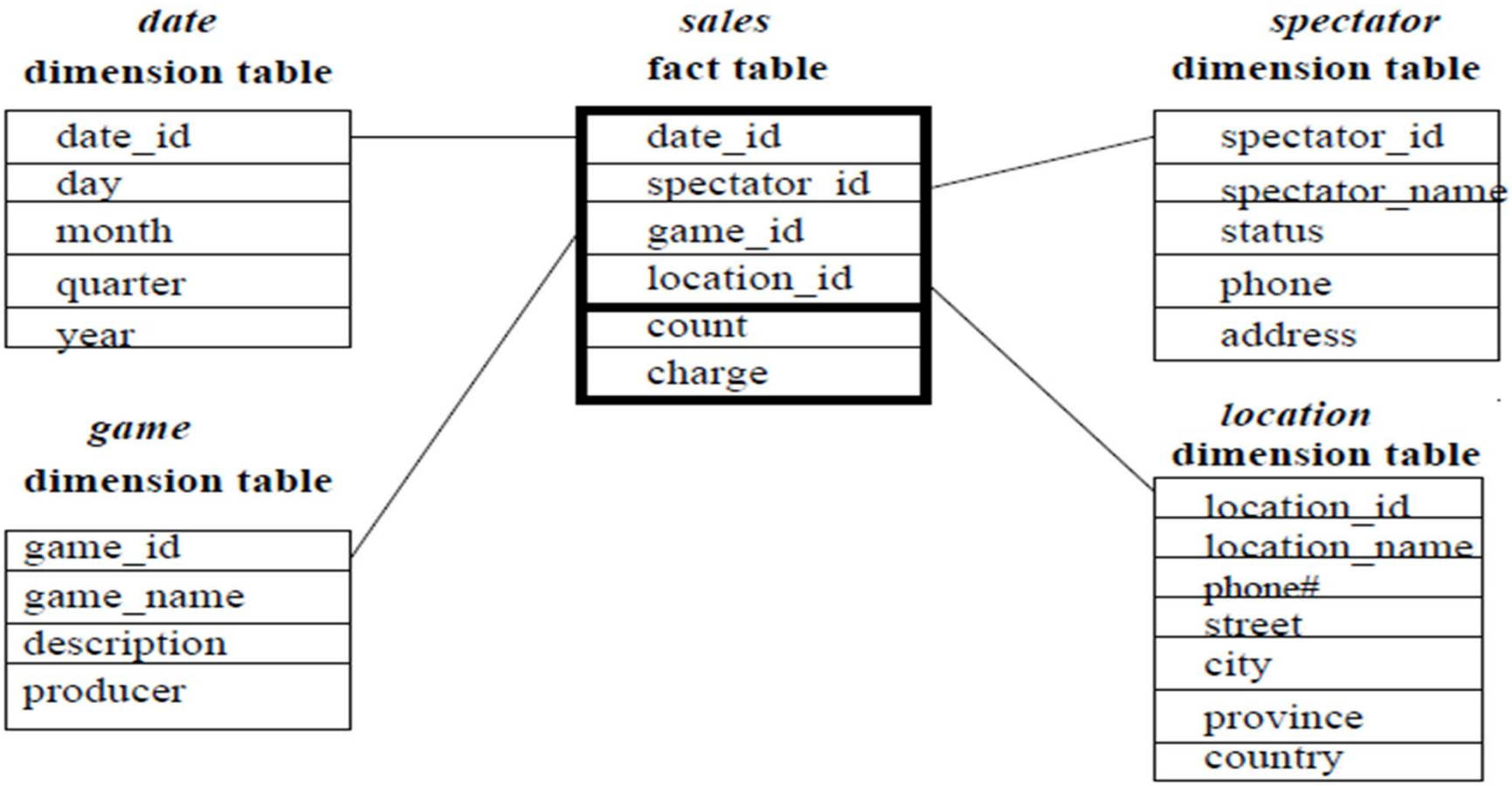
(c) If each dimension has five levels (including all), such as student < major < status < university < all, how many cuboids will this cube contain (including the base and apex cuboids)?

This cube will contain 5^4 = 625 cuboids.

Suppose that a data warehouse consists of the four dimensions, *date*, *spectator*, *location*, and *game*, and the two measures, *count* and *charge*, where *charge* is the fare that a spectator pays when watching a game on a given date. Spectators may be students, adults, or seniors, with each category having its own charge rate.

(a) Draw a *star schema* diagram for the data warehouse.

(b) Starting with the base cuboid [*date*, *spectator*, *location*, *game*], what specific *OLAP operations* should one perform in order to list the total charge paid by student spectators at GM_Place in 2010?



Que: Starting with the base cuboid [date, spectator, location, game], what specific OLAP operations should one perform in order to list the total charge paid by student spectators at GM Place in 2010?

Que: Starting with the base cuboid [date, spectator, location, game], what specific OLAP operations should one perform in order to list the total charge paid by student spectators at GM Place in 2010?

Ans:

The specific OLAP operations to be performed are:

- Roll-up on date from date id to year.
- Roll-up on game from game id to all.
- Roll-up on location from location id to location name.
- Roll-up on spectator from spectator id to status.
- Dice with status="students", location name="GM Place", and year = 2010.