UNIT-I
Data Warehousing

Contents:

- Basic Concepts
- Data Warehousing Components
- OLAP and OLTP
- Multidimensional Data Model
- Data Warehouse Schemas for Decision Support
- Concept Hierarchies
- Typical OLAP Operations
- 3-tier DW Architecture

Basic Concepts:

- Data Warehouse:

1.A data warehouse refers to a data repository that is maintained separately from an organization's operational databases.

2. Data warehouse systems allow for integration of a variety of application systems.

3. They support information processing by providing a solid platform of consolidated historic data for analysis

Basic Concepts:

- 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”.

NOTE: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

Basic Concepts:

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

  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**.**

- 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.

Basic Concepts:

- 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.

NOTE: ***To discover trends in business***

- Nonvolatile: A data warehouse is always a physically separate store of data transformed from the application data found in the operational environment. Due to this separation, a data warehouse does not require ***transaction processing, recovery, and concurrency control mechanisms***. It usually requires only two operations in data accessing: initial loading of data and access of data.

- NOTE:- ***Non-volatile means the previous data is not erased when new data is added to it.***

“*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

“Differences between Operational Database Systems and Data Warehouses

The major task of online operational database systems is to perform online transaction and query processing. These systems are called online transaction processing (OLTP) systems.

They cover most of the day-to-day operations of an organization such as purchasing,

- inventory,
- manufacturing,
- banking,
- payroll,
- registration,
- and accounting

“Differences between Operational Database Systems and Data Warehouses

DW serve users or knowledge workers in the role of data analysis and decision making. Such systems can organize and present data in various formats in order to accommodate the diverse needs of different users. These systems are known as online analytical processing (OLAP) systems.

Users and system orientation: An OLTP system is customer-oriented and is used for transaction and query processing by clerks, clients, and information technology professionals. An OLAP system is market-oriented and is used for data analysis by knowledge workers, including managers, executives, and analysts.

Data contents: An OLTP system manages current data that, typically, are too detailed to be easily used for decision making. An OLAP system manages large amounts of historic data, provides facilities for summarization and aggregation, and stores and manages information at different levels of granularity.

“Differences between Operational Database Systems and Data Warehouses

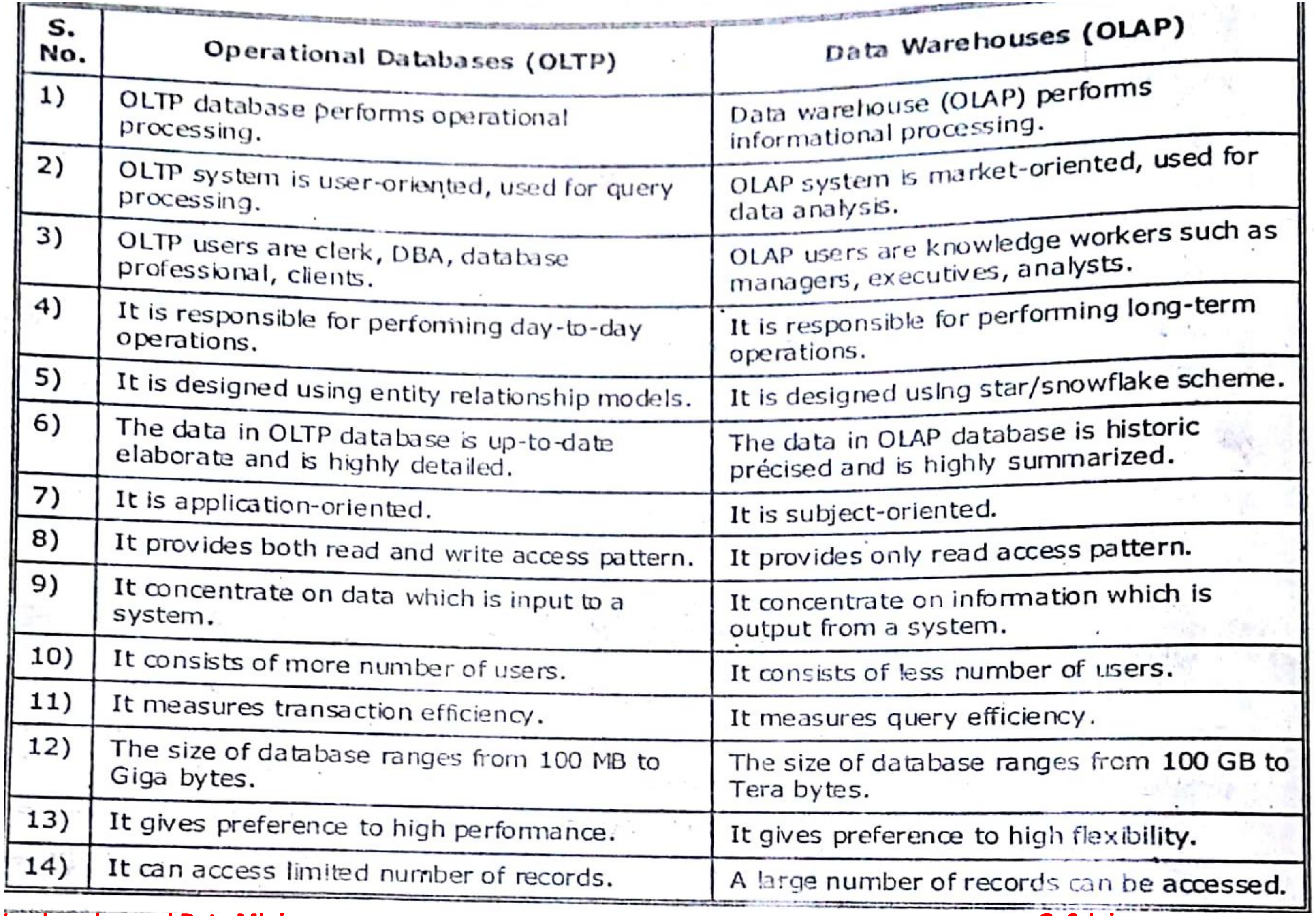
Database design: An OLTP system usually adopts an entity-relationship (ER) data model and an application-oriented database design. An OLAP system typically adopts either a star or a snowflake model and a subject-oriented database design.

View: An OLTP system focuses mainly on the current data within an enterprise or department, without referring to historic data or data in different organizations.

. In contrast, an OLAP system often spans multiple versions of a database schema, due to the evolutionary process of an organization.

Access patterns: The access patterns of an OLTP system consist mainly of short, atomic transactions. Such a system requires concurrency control and recovery mechanisms'. However, accesses to OLAP systems are mostly read-only operations (because most data warehouses store historic rather than up-to-date information),

“Differences between Operational Database Systems and Data Warehouses



S. No.	Operational Databases (OLTP)	Data Warehouses (OLAP)
1)	OLTP database performs operational processing.	Data warehouse (OLAP) performs informational processing.
2)	OLTP system is user-oriented, used for query processing.	OLAP system is market-oriented, used for data analysis.
3)	OLTP users are clerk, DBA, database professional, clients.	OLAP users are knowledge workers such as managers, executives, analysts.
4)	It is responsible for performing day-to-day operations.	It is responsible for performing long-term operations.
5)	It is designed using entity relationship models.	It is designed using star/snowflake scheme.
6)	The data in OLTP database is up-to-date elaborate and is highly detailed.	The data in OLAP database is historic précised and is highly summarized.
7)	It is application-oriented.	It is subject-oriented.
8)	It provides both read and write access pattern.	It provides only read access pattern.
9)	It concentrate on data which is input to a system.	It concentrate on information which is output from a system.
10)	It consists of more number of users.	It consists of less number of users.
11)	It measures transaction efficiency.	It measures query efficiency.
12)	The size of database ranges from 100 MB to Giga bytes.	The size of database ranges from 100 GB to Tera bytes.
13)	It gives preference to high performance.	It gives preference to high flexibility.
14)	It can access limited number of records.	A large number of records can be accessed.

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

Dimensions are the perspectives or entities with respect to which an organization wants to keep records.

For example, All Electronics may create a sales data warehouse in order to keep records of the store's sales with respect to the dimensions time, item, branch, and location.

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. Think of them as the quantities by which we want to analyze relationships between dimensions

Data Warehouse Modeling: Data Cube and OLAP
Example:2D View of Automobile sales data

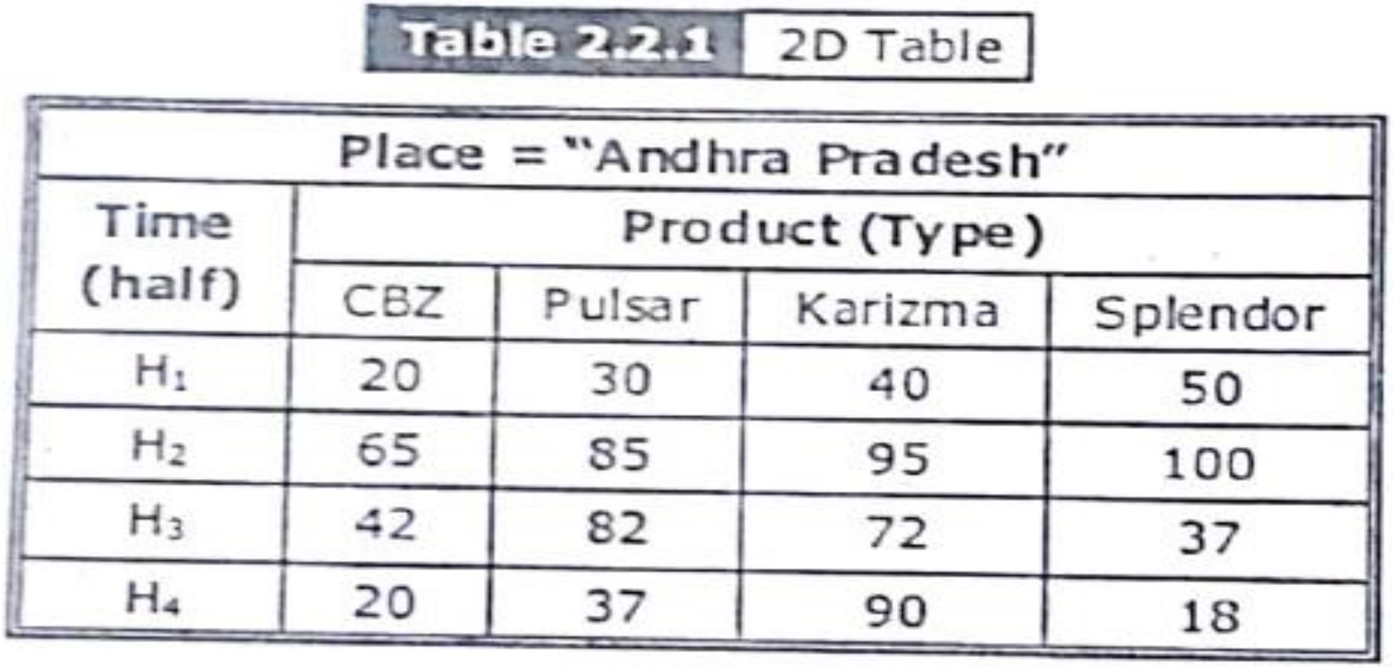


Table 2.2.1 2D Table

Place = "Andhra Pradesh"				
Time (half)	Product (Type)			
	CBZ	Pulsar	Karizma	Splendor
H_1	20	30	40	50
H_2	65	85	95	100
H_3	42	82	72	37
H_4	20	37	90	18

Data Warehouse Modeling: Data Cube and OLAP

Example:3D View of Automobile sales data

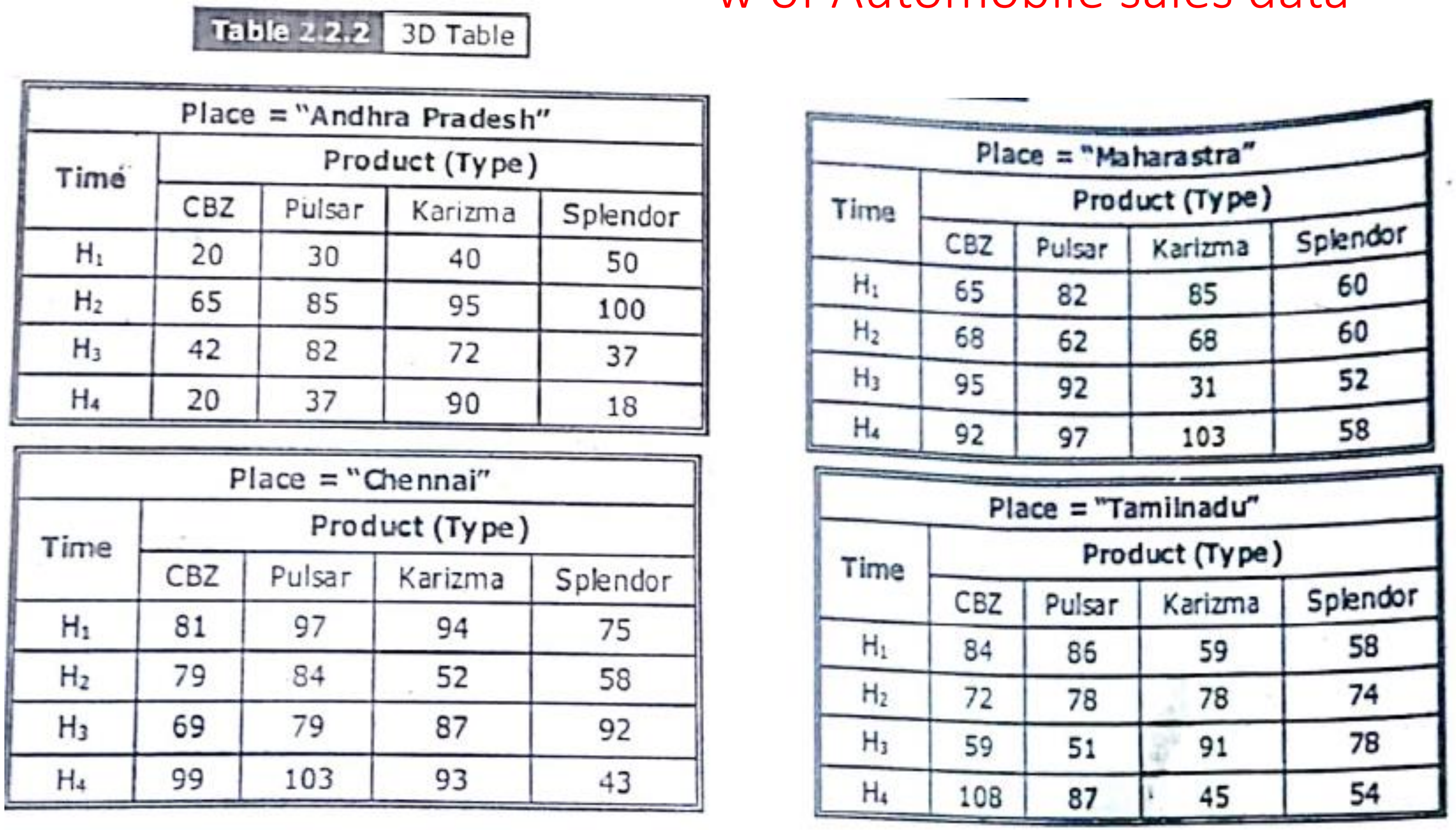


Table 2.2.2 3D Table

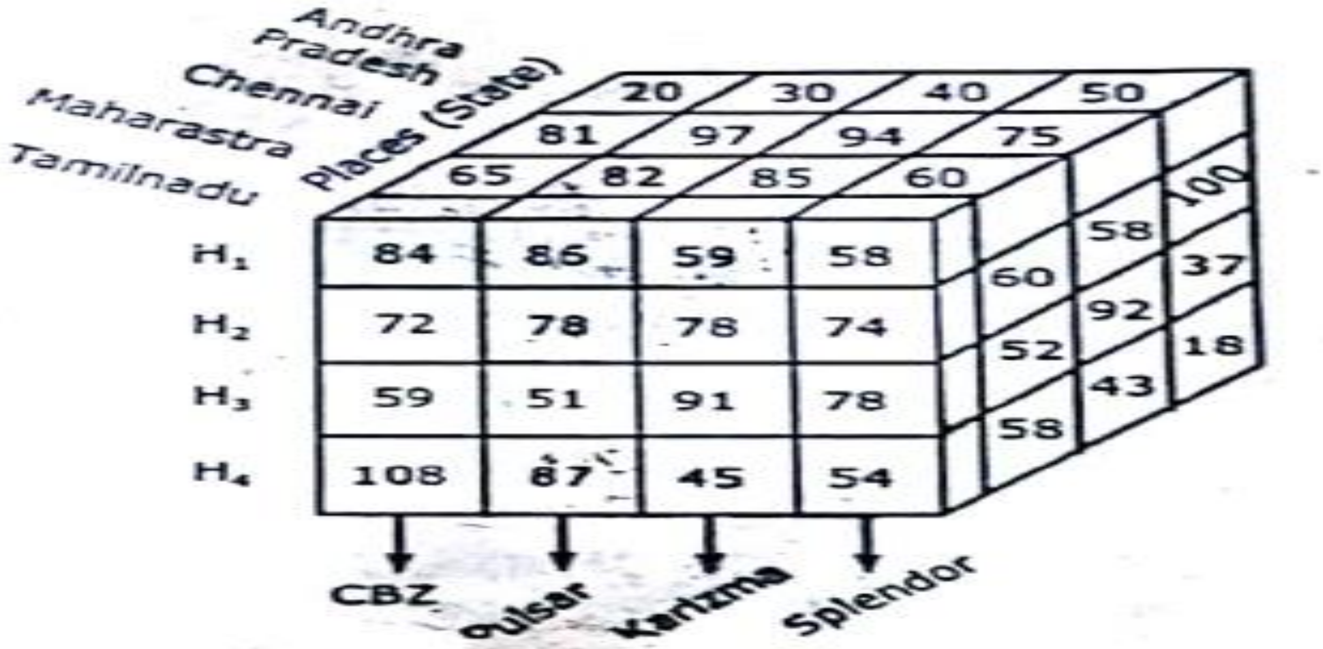
Place = "Andhra Pradesh"				
Time	Product (Type)			
	CBZ	Pulsar	Karizma	Splendor
H_1	20	30	40	50
H_2	65	85	95	100
H_3	42	82	72	37
H_4	20	37	90	18

Place = "Chennai"				
Time	Product (Type)			
	CBZ	Pulsar	Karizma	Splendor
H_1	81	97	94	75
H_2	79	84	52	58
H_3	69	79	87	92
H_4	99	103	93	43

Place = "Maharastra"				
Time	Product (Type)			
	CBZ	Pulsar	Karizma	Splendor
H_1	65	82	85	60
H_2	68	62	68	60
H_3	95	92	31	52
H_4	92	97	103	58

Place = "Tamilnadu"				
Time	Product (Type)			
	CBZ	Pulsar	Karizma	Splendor
H_1	84	86	59	58
H_2	72	78	78	74
H_3	59	51	91	78
H_4	108	87	45	54

Data Warehouse Modeling: Data Cube and OLAP
Example:3D View of Automobile sales data cube



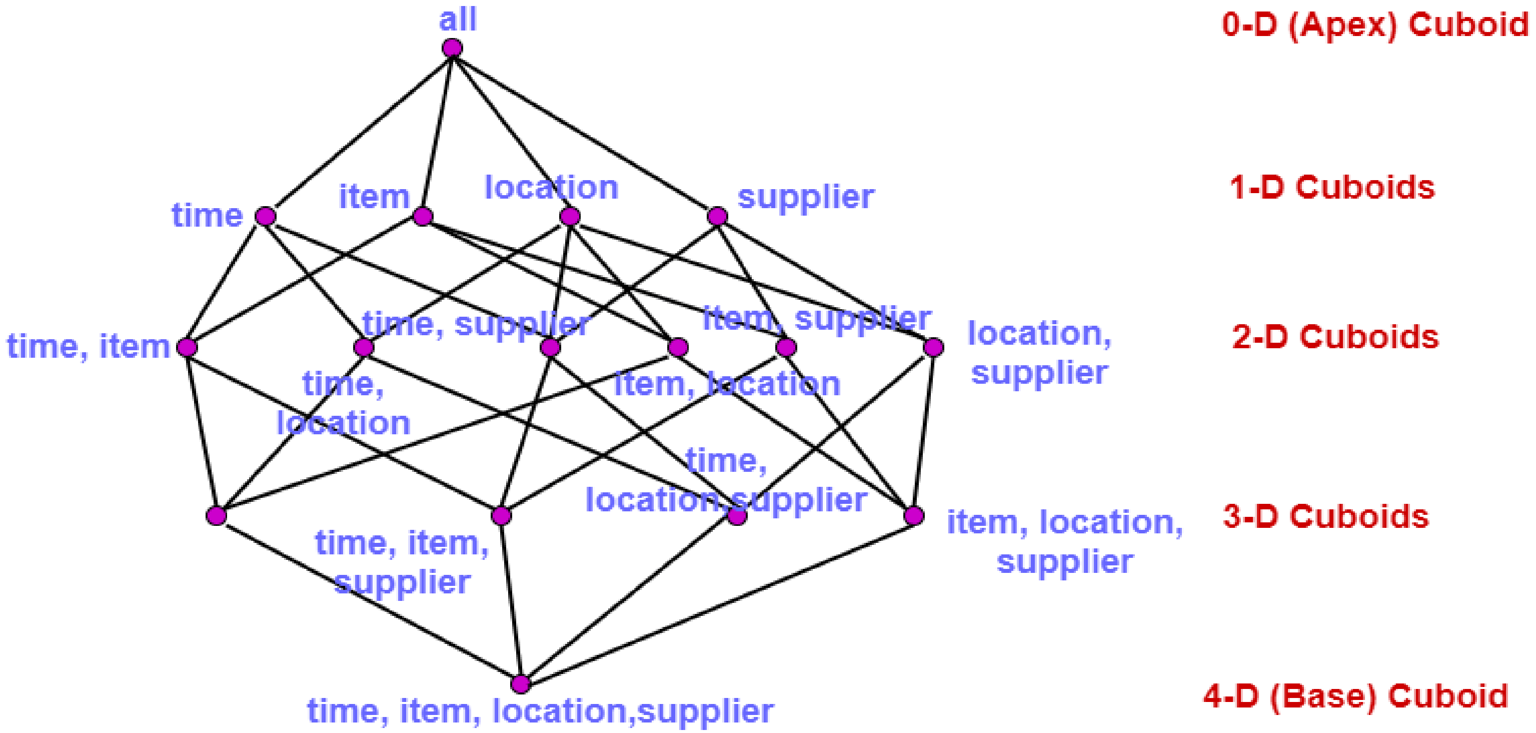
	CBZ	Pulsar	Karizma	Splendor
H₁	84	86	59	58
H₂	72	78	78	74
H₃	59	51	91	78
H₄	108	87	45	54

Fig. 2.2.1 3D Data Cube

- The cuboid that holds the lowest level of summarization is called the base cuboid.
- The 0-D cuboid, which holds the highest level of summarization, is called the apex cuboid.

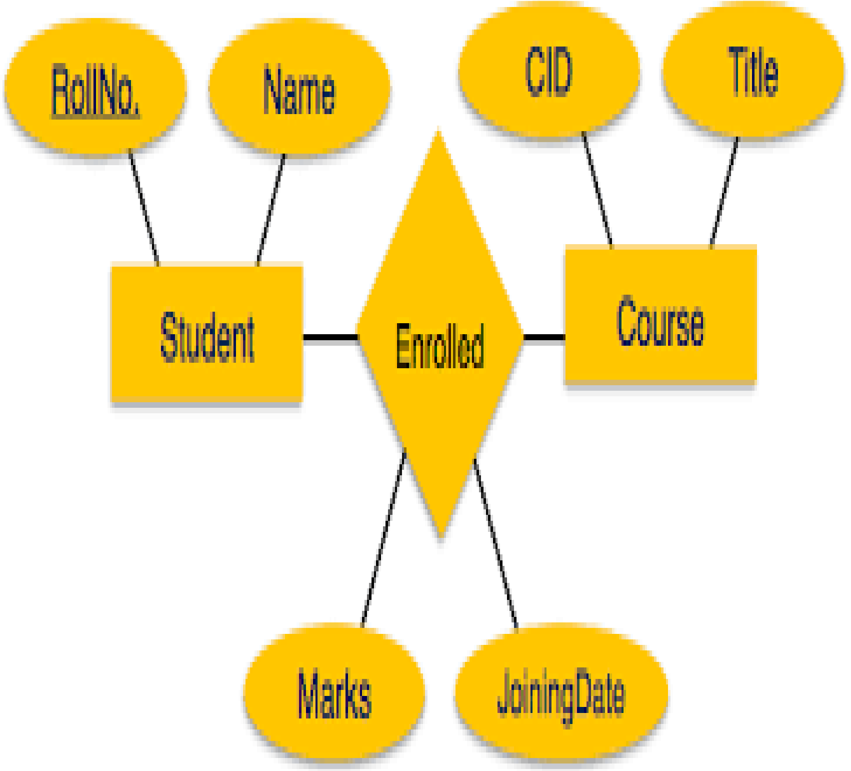
Lattice of cuboids, making up a 4-D data cube for time, item, location, and supplier. Each cuboid represents a different degree of summarization.

- The apex cuboid, or 0-D cuboid, refers to the case where the group-by is empty. It contains the total sum of all sales. The apex cuboid is the most generalized (least specific) of the cuboids, and is often denoted as all.
- The base cuboid is the least generalized (most specific) of the cuboids.


all
0-D (Apex) Cuboid
time
item
location
supplier
1-D Cuboids
time, item
time, location
time, supplier
item, location
item, supplier
location, supplier
2-D Cuboids
time, item, location
time, item, supplier
time, location, supplier
item, location, supplier
3-D Cuboids
time, item, location,supplier
4-D (Base) Cuboid

Stars, Snowflakes, and Fact Cor
Multidimensional Data Models.

The entity-relationship data model is commonly used in the design of relational databases, where a database schema consists of a set of entities and the relationships between them



The most popular data model for a data warehouse is a **multidimensional model,** which can exist in the form of a **star schema, a snowflake schema, or a fact constellation schema**.

Stars, Snowflakes, and Fact Constellations: Schemas for Multidimensional Data Models.

Star Schema:

The most common modeling paradigm is the star schema, in which the data warehouse contains

(1) a large central table (**fact table) containing the bulk of** the data, with no redundancy, and

(2) a set of smaller attendant tables (**dimension tables), one for each dimension.**

The schema graph resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.

Stars, Snowflakes, and Fact Cor Multidimensional Data Models.

- **Star Scheme:** Sales are considered along four dimensions: time, place, division, and product. 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_id and product_id) are system-generated identifiers.

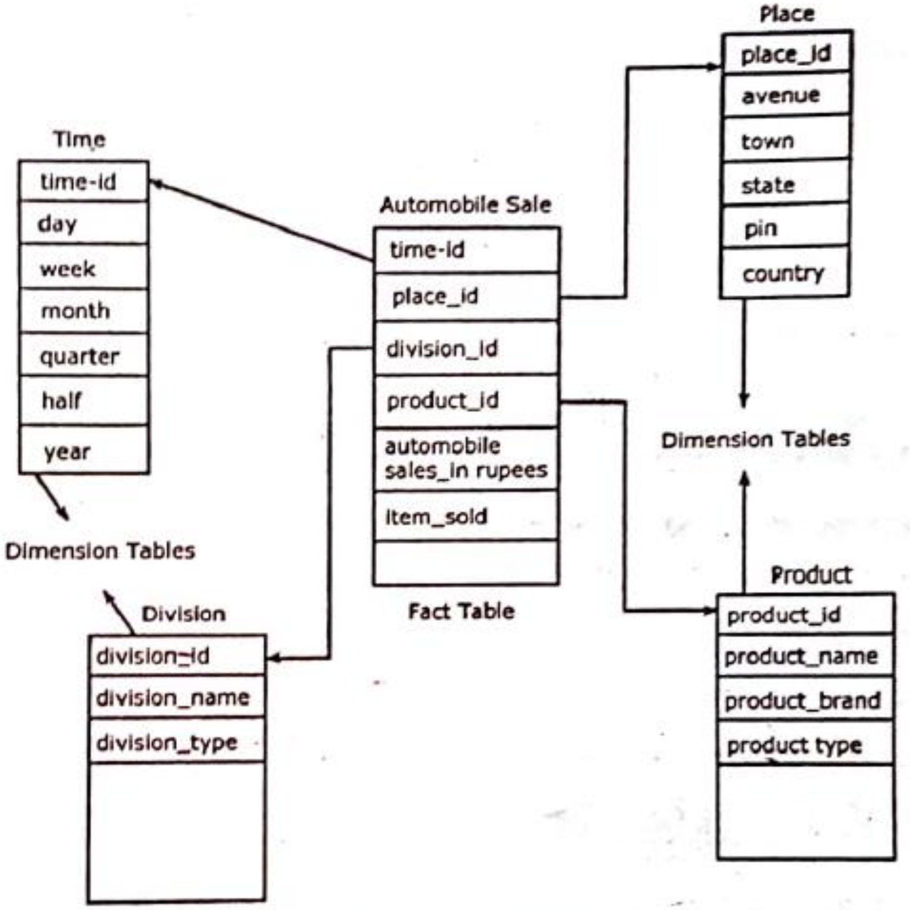
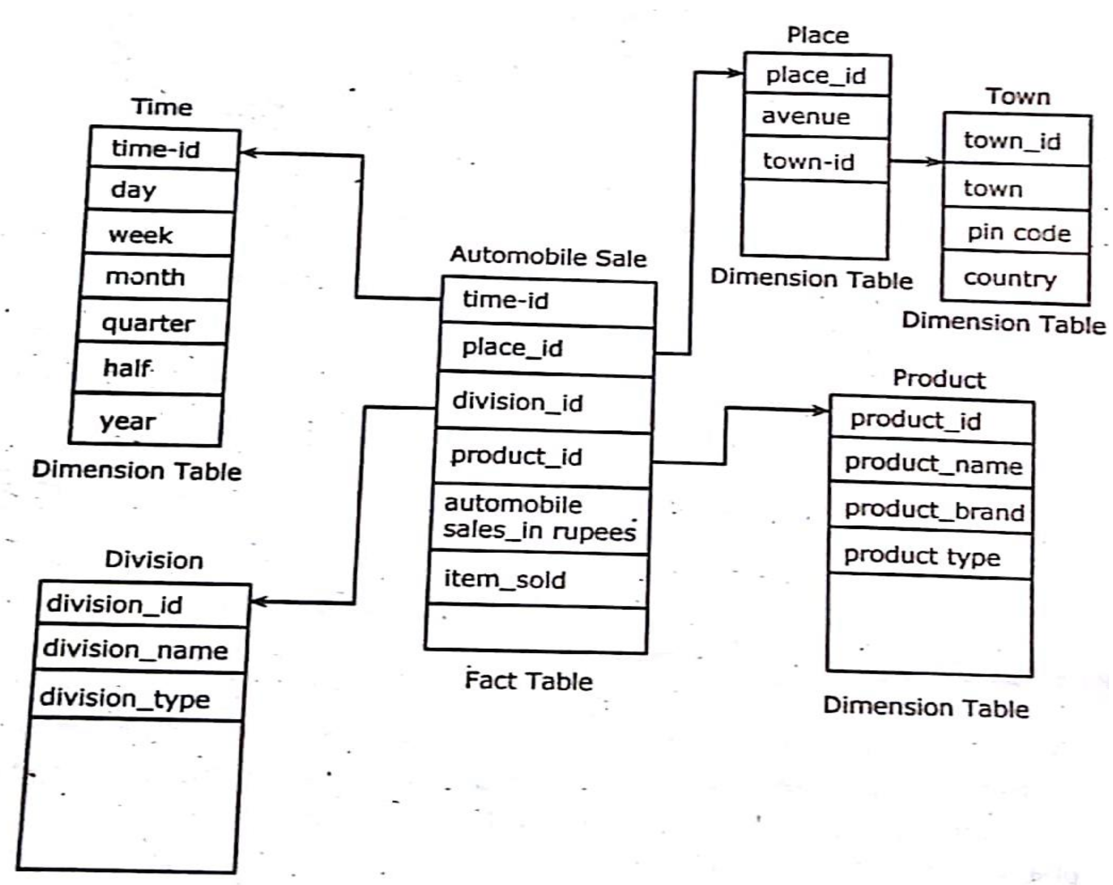


Fig. 2.2.3 Star Schema of Data Warehouse for Automobile Sales

Stars, Snowflakes, and Fact Multidimensional Data Mo

- **Snowflake schema: The snowflake schema is a variant of the star schema model,** where some dimension tables are normalized, thereby further splitting the data into additional tables.
- The resulting schema graph forms a shape similar to a snowflake..



Time: time-id, day, week, month, quarter, half, year — Dimension Table

Division: division_id, division_name, division_type

Automobile Sale: time-id, place_id, division_id, product_id, automobile sales_in rupees, item_sold — Fact Table

Place: place_id, avenue, town-id — Dimension Table

Town: town_id, town, pin code, country — Dimension Table

Product: product_id, product_name, product_brand, product type — Dimension Table

Fig. 2.2.4 Snowflake Schema of Data Warehouse for Automobile Sales

- **Snowflake schema: The snowflake schema is a variant of the star schema model,**

Stars, Snowflakes, and Fact Constellations: Schemas for Multidimensional Data Models:

1. The major difference between the snowflake and star schema models is that the dimension tables of the snowflake model may be kept in normalized form to reduce redundancies. Such a table is easy to maintain and saves storage space.
2. The snowflake structure can reduce the effectiveness of browsing, since more joins will be needed to execute a query.
3. The single dimension table for *place in the star* schema can be normalized into two new tables: *place and town*. *The city key in the* new *place table links to the town dimension*.

Stars, Snowflakes, and Fact Constellations: Schemas for Multidimensional Data Models.

- 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.
- This schema specifies two fact tables, sales and inventory. The sales table definition is identical to that of the star schema .
- The shipping table has three dimensions, or keys—product_id, time_id, store_id—and two measures—dollars cost and units shipped.
- A fact constellation schema allows dimension tables to be shared between fact tables. For example, the dimensions tables for time, product, and place are shared between the sales and shipping fact tables.

Sta
Mu

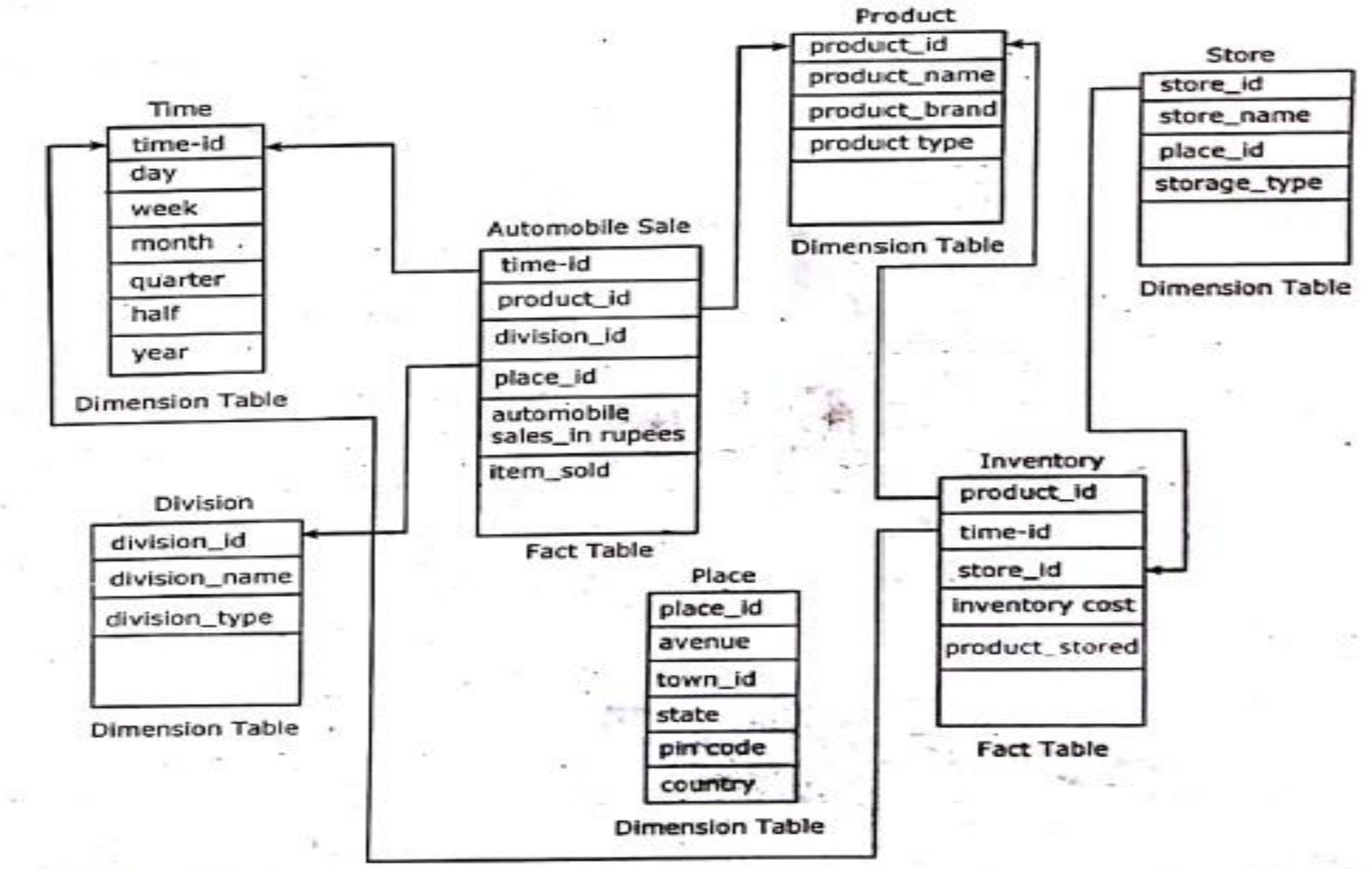


Fig. 2.2.5 Fact Constellation Schema of Data Warehouse for Automobile Sales

Stars, Snowflakes, and Fact Constellations: Schemas for Multidimensional Data Models.

- 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 department wide. For data marts, the star or snowflake schema is commonly used, since both are geared toward modeling single subjects

Dimensions: The Role of Concept Hierarchies

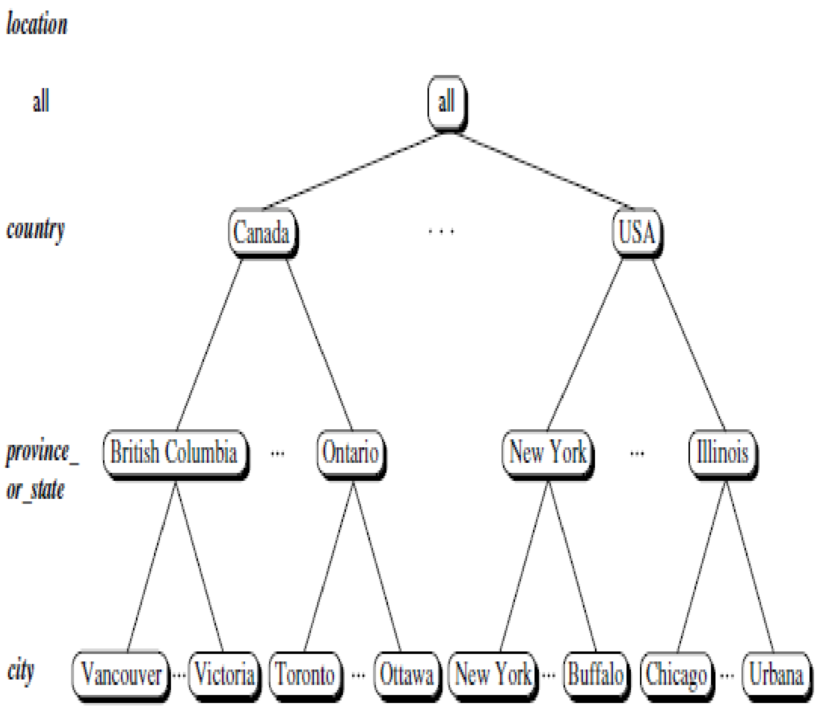
- A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts.
- Example: Consider a concept hierarchy City values for location include Vancouver, Toronto, New York, and Chicago. Each city, however, can be mapped to the province or state to which it belongs. For example, Vancouver can be mapped to British Columbia, and Chicago to Illinois.
- The provinces and states can in turn be mapped to the country (e.g., Canada or the United States) to which they belong for the dimension location.
- Many concept hierarchies are implicit within the database schema. For example, suppose that the dimension location is described by the attributes number, street, city, province or state, zip code, and country. These attributes are related by a total order, forming a concept hierarchy such as “street < city < province or state < country.”

Dimensions: The Role of Concept Hierarchies

- Alternatively, the attributes of a dimension may be organized in a partial order, forming a lattice. An example of a partial order for the time dimension based on the attributes day, week, month, quarter, and year is “day <month < quarter; week < year.”1 .This lattice structure is shown in Figure.
- NOTE:A concept hierarchy that is a total or partial order among attributes in a database schema is called a **schema hierarchy.**

- Concept hierarchies may also be defined by discretizing or grouping values for a given dimension or attribute, resulting in a set-grouping hierarchy. A total or partial order can be defined among groups of values.
- Example:

The dimension price, where an interval [$X : : :$Y] denotes the range from $X (exclusive) to $Y (inclusive).



4.9 A concept hierarchy for *location*. Due to space limitations, not all of the hierarchy nodes are shown, indicated by ellipses between nodes.

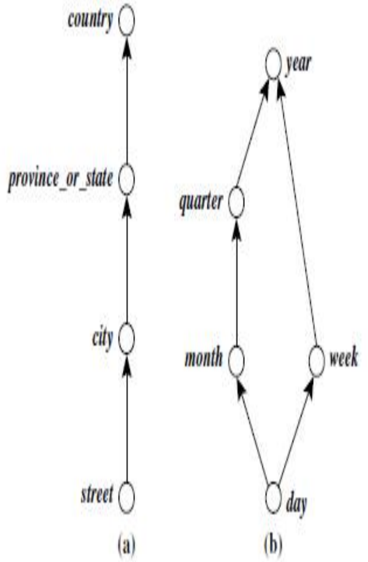
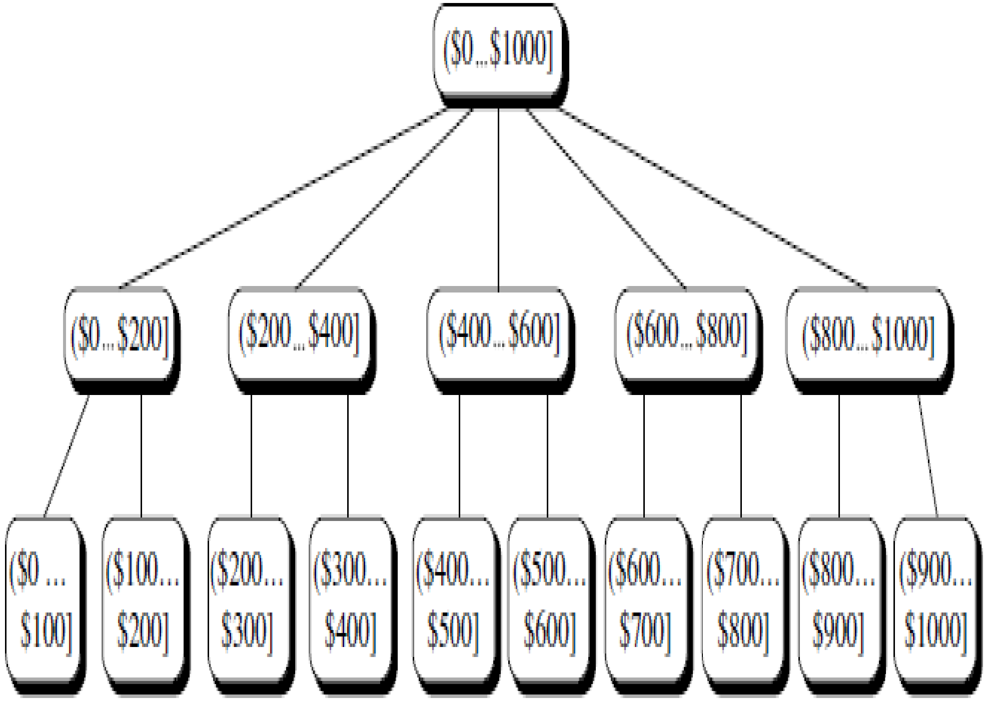


Figure 4.10 Hierarchical and lattice structures of attributes in warehouse dimensions: (a) a hierarchy for *location* and (b) a lattice for *time*.

Dime



4.11 A concept hierarchy for *price.*

- NOTE:

Concept hierarchies may be provided manually by system users, domain experts, or knowledge engineers, or may be automatically generated based on statistical analysis of the data distribution

- Multidimensional Model 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.

OLAP Operations

- Example: The cube contains the dimensions place, time, and product, where location is aggregated with respect to city values, time is aggregated with respect to quarters, and item is aggregated with respect to item types.

OLAP Operations

1.**Roll-up: The roll-up operation (also called the drill-up operation by some vendors)** performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension or by dimension reduction.

- Example: The roll-up operation shown aggregates the data by ascending the location hierarchy from the level of city to the level of country.
- NOTE: When roll-up is performed by dimension reduction, one or more dimensions are removed from the given cube.

2. **Drill-down: Drill-down is the reverse of roll-up. It navigates from less detailed data** to more detailed data. Drill-down can be realized by either stepping down a concept hierarchy for a dimension or introducing additional dimensions.

- Drill-down occurs by descending the time hierarchy from the level of quarter to the more detailed level of month. The resulting data cube details the total sales per month rather than summarizing them by quarter.

3. Slice and dice: The slice operation performs a selection on one dimension of the given cube, resulting in a subcube.

- The dice operation defines a subcube by performing a selection on two or more dimensions

OLAP Operations

4.Pivot (rotate): Pivot (also called rotate) is a visualization operation that rotates the data axes in view to provide an alternative data presentation

- Other OLAP operations: Some OLAP systems offer additional drilling operations. For example, drill-across executes queries involving (i.e., across) more than one fact table.
- The drill-through operation uses relational SQL facilities to drill through the bottom level of a data cube down to its back-end relational tables.

Drill-up

$\text{Roll-up}_{\text{location}}$ C[Half, State, Product] = C[Half, Country, Product]

Here, roll-up operation is performed on location (i.e., from states to countries).

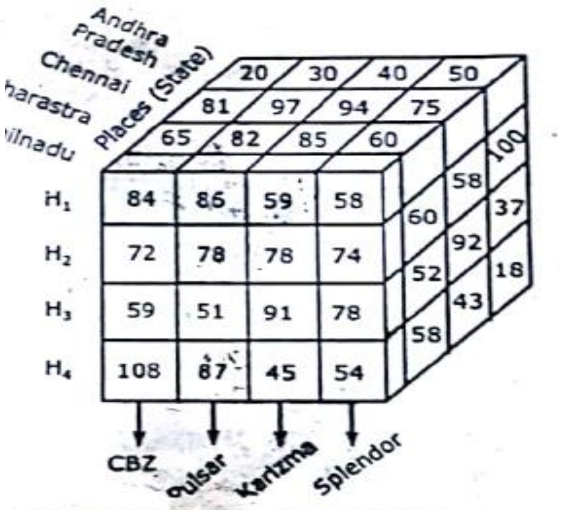
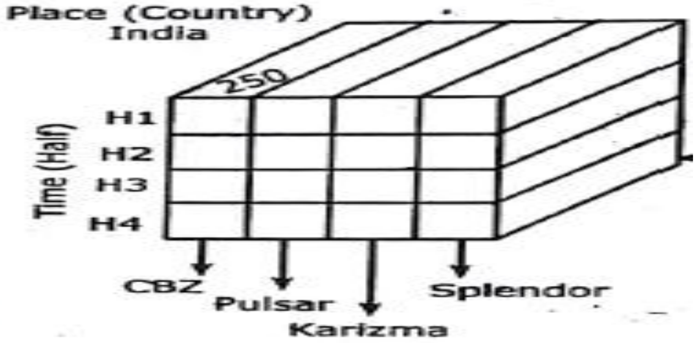


Fig. 2.2.1 3D Data Cube



Drill-down

$\text{Drill-down}_{\text{time}}$ C[Half, Product] = C[Quarter, Product]

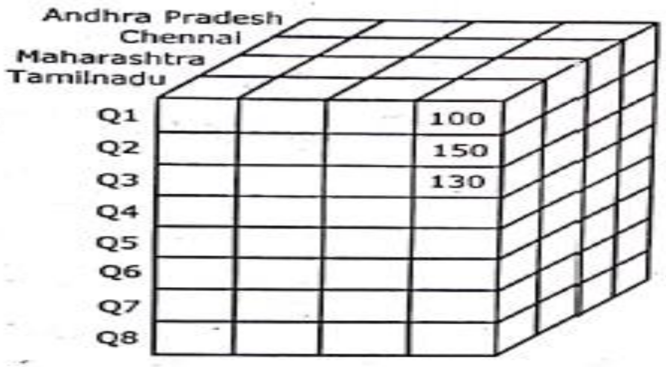


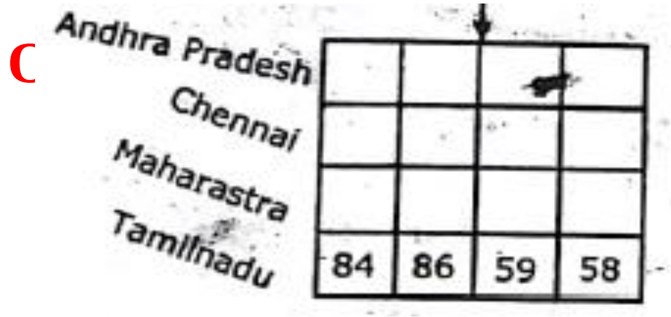
Fig. 2.2.7 Drill-down Operation

Here, drill-down operation is performed on time (i.e., from half to quarter).

Slice

Slice$_{time\ =\ 'H1'}$ C[Half, State, Product] = C[State, Product]

Here, slicing is performed on one dimension i.e., H1 as shown in Fig. 2.2.6.



Dice

Dice$_{time='H1'\ or\ 'H3'}$ = "Chennai" or "Tamilnadu" C[Half, State, Product]

= C[Half, State', Product']

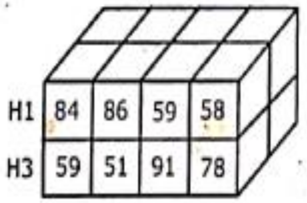


Fig. 2.2.8 Dicing Operation

Here, dicing is performed on two dimension H1 and H3.

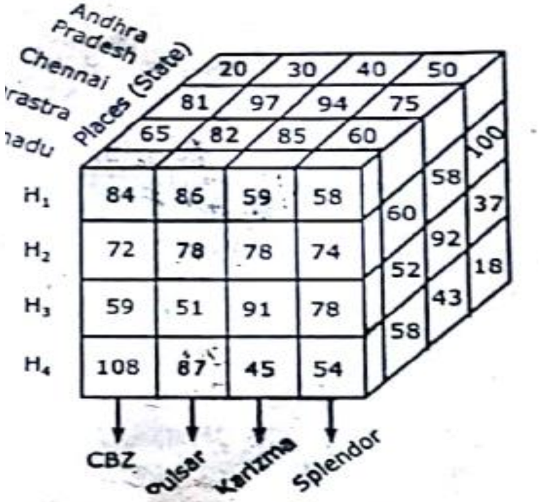


Fig. 2.2.1 3D Data Cube

Rotate

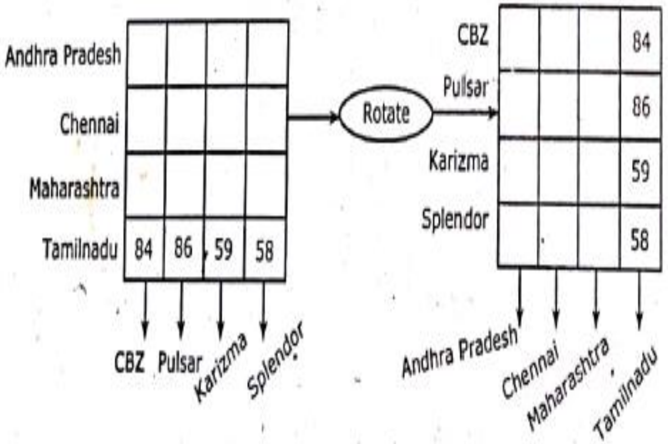


Fig. 2.2.9 Rotate Operation

Starnet Query Model for Querying Multidimensional Databases

A starnet query model is a model for querying multidimensional databases. It consists of multiple "radial lines" all emitting from a center point and representing the concept hierarchies for dimensions in the data warehouse. These radial lines have multiple "footprints" that represent the abstraction levels (high or low) of the dimensions.

Example

Consider a All Automobile data warehouse with dimensions place, product and time. The starnet model for the All Automobile data warehouse is shown in Fig. 2.2.10. It has three radial lines, one for each dimension. The radial lines product has three footprints, product_name, product_brand and product_type, "place" has five footprints avenue, town, state, pin and country, "time" has six footprints, day, week, month, quarter, half and year.

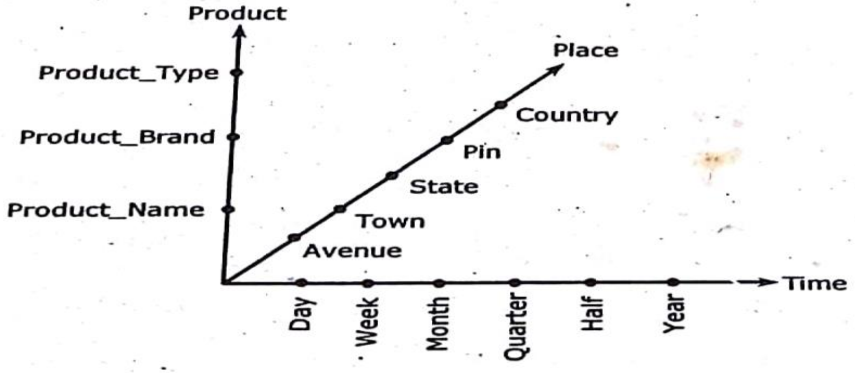


Fig. 2.2.10 A Starnet Model of Student Data Warehouse

Data Warehousing: A Multitiered Architecture

Data warehouse can be visualized as three-tier architecture as shown in Fig. 2.3.1.

1) Tier 1 (or bottom tier) represents data warehouse server.

2) Tier 2 (or middle tier) represents the OLAP engine or OLAP server.

3) Tier 3 (or top tier) represents the front-end client layer.

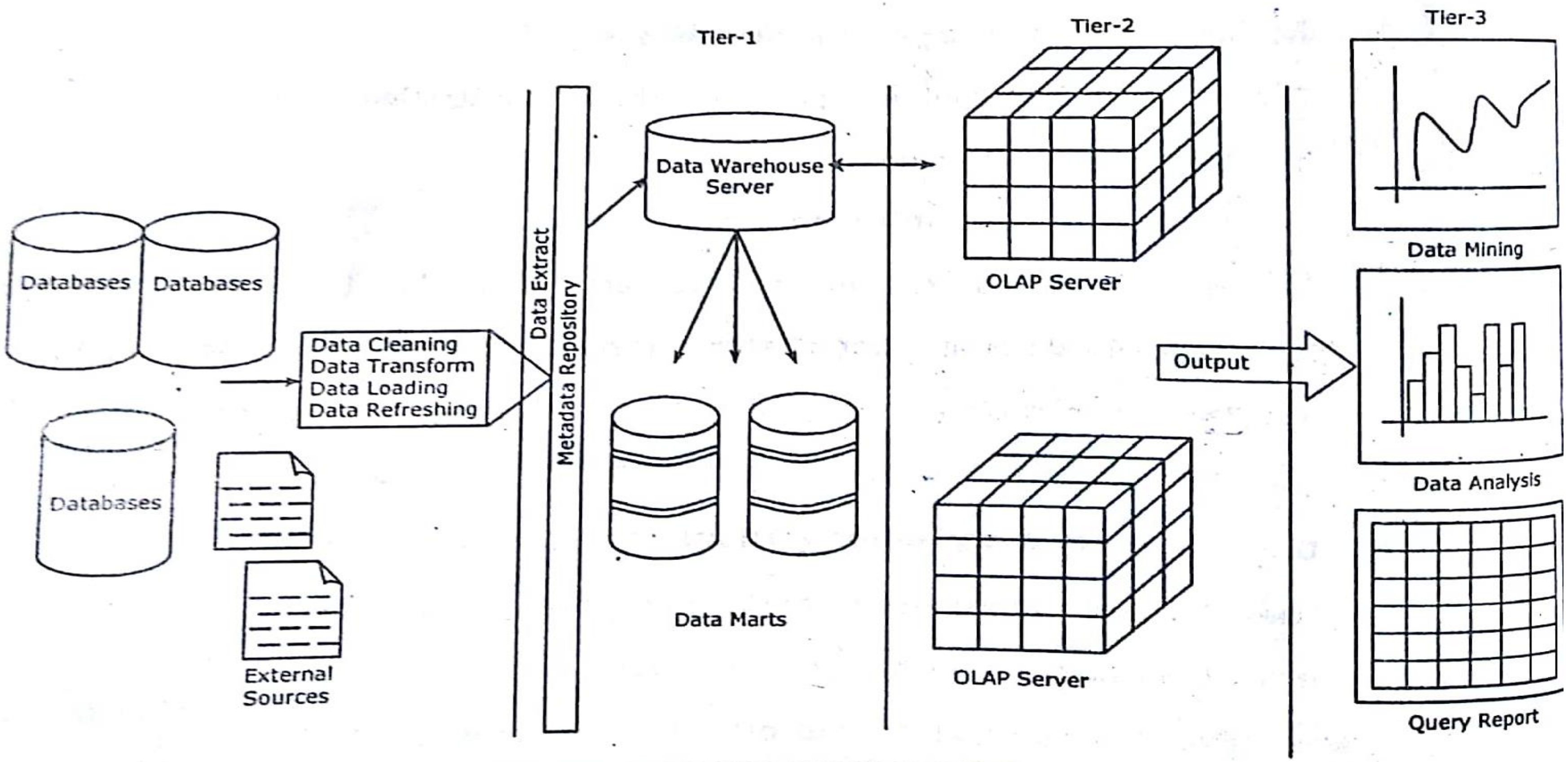
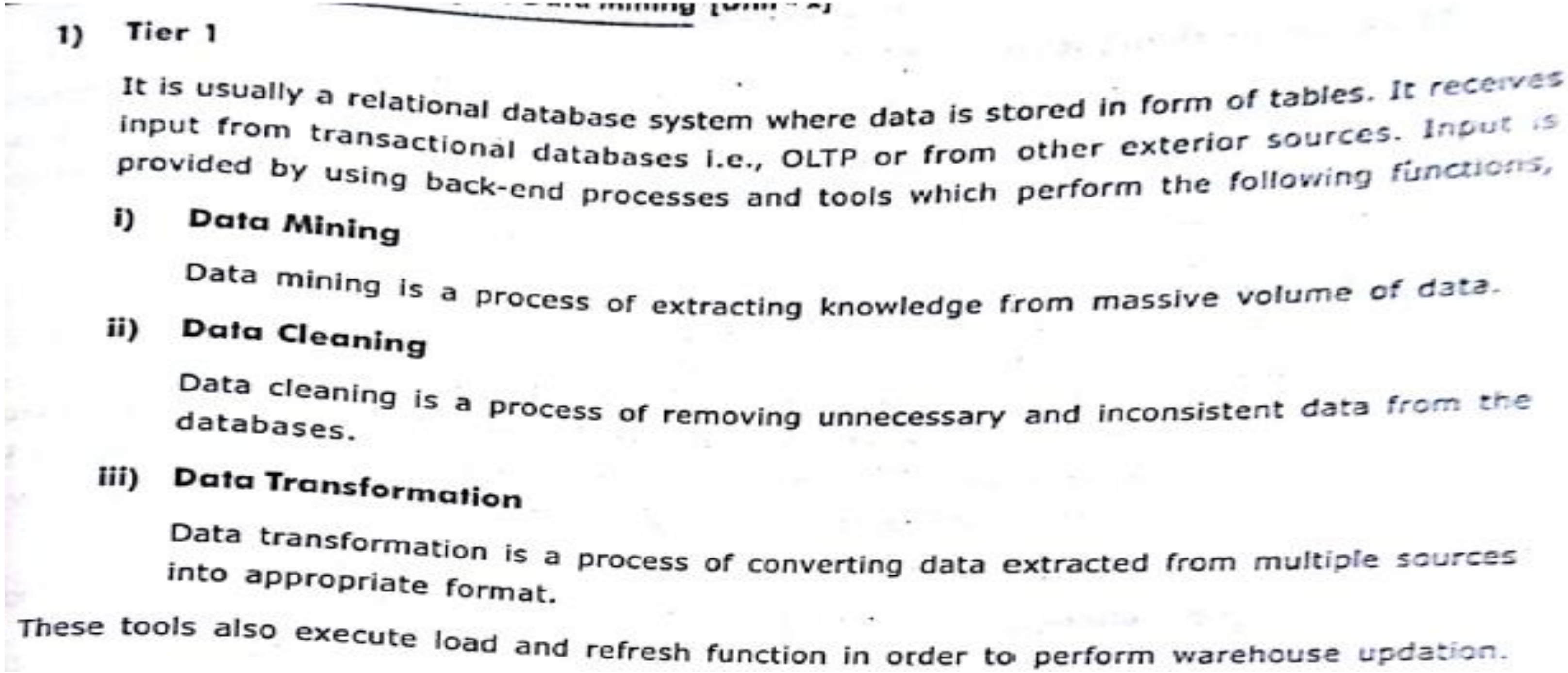


Fig. 2.3.1 Data Warehouse Architecture

Data Warehousing: A Multitiered Architecture



1) **Tier 1**

It is usually a relational database system where data is stored in form of tables. It receives input from transactional databases i.e., OLTP or from other exterior sources. Input is provided by using back-end processes and tools which perform the following functions,

i) **Data Mining**

Data mining is a process of extracting knowledge from massive volume of data.

ii) **Data Cleaning**

Data cleaning is a process of removing unnecessary and inconsistent data from the databases.

iii) **Data Transformation**

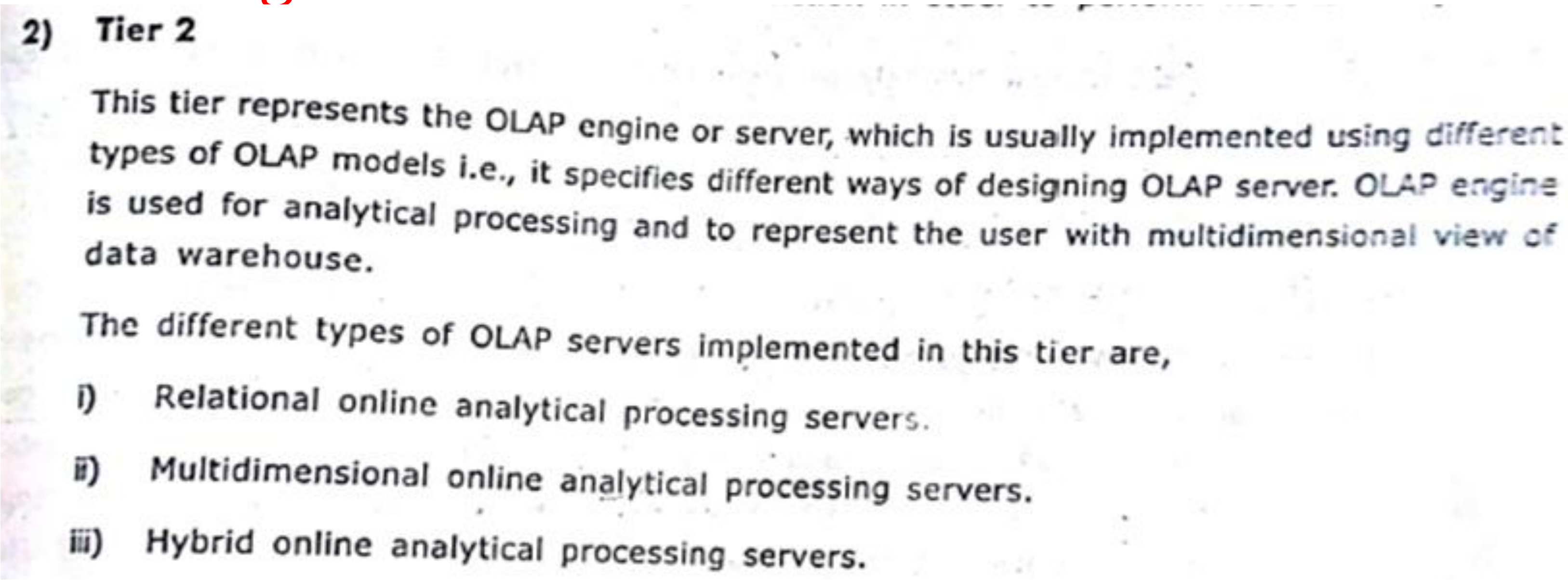
Data transformation is a process of converting data extracted from multiple sources into appropriate format.

These tools also execute load and refresh function in order to perform warehouse updation.

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 Warehousing: A Multitiered Architecture



2) **Tier 2**

This tier represents the OLAP engine or server, which is usually implemented using different types of OLAP models i.e., it specifies different ways of designing OLAP server. OLAP engine is used for analytical processing and to represent the user with multidimensional view of data warehouse.

The different types of OLAP servers implemented in this tier are,

i) Relational online analytical processing servers.

ii) Multidimensional online analytical processing servers.

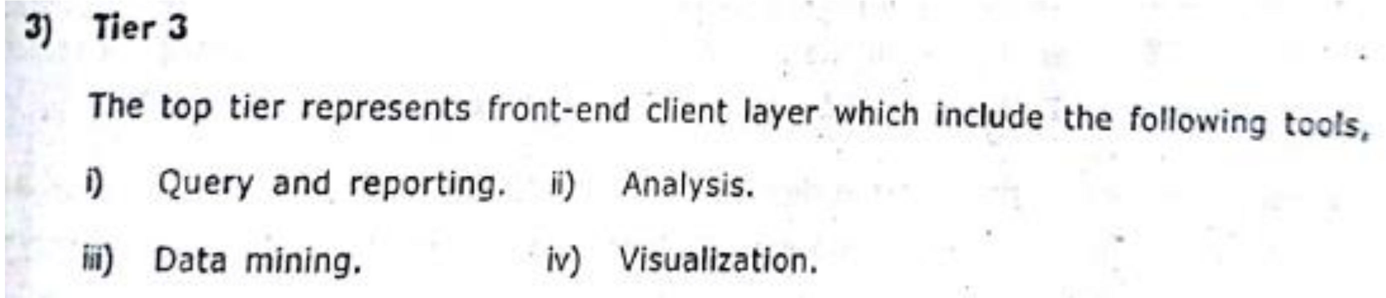
iii) Hybrid online analytical processing servers.

(1) a relational OLAP (ROLAP) model (i.e., an extended relational DBMS that maps operations on multidimensional data to standard relational operations);

(2) a multidimensional OLAP (MOLAP) model (i.e., a special-purpose server that directly implements multidimensional data and operations).

(3) HOLAP approach combines ROLAP & MOLAP technology.

Data Warehousing: A Multitiered Architecture



3) **Tier 3**

The top tier represents front-end client layer which include the following tools,

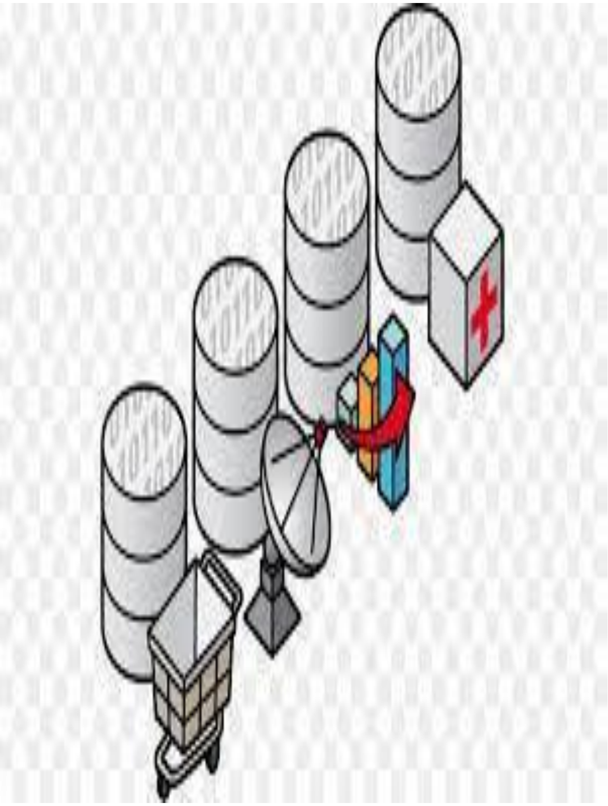
i) Query and reporting. ii) Analysis.

iii) Data mining. iv) Visualization.

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

Enterprise warehouse: An enterprise warehouse collects all of the information about subjects spanning the entire organization. It provides corporate-wide data integration, usually from one or more operational systems or external information providers.

An enterprise data warehouse may be implemented on traditional mainframes, computer super servers, or parallel architecture platforms.



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.

Examples: data mart may confine its subjects to customer, item, and sales.

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

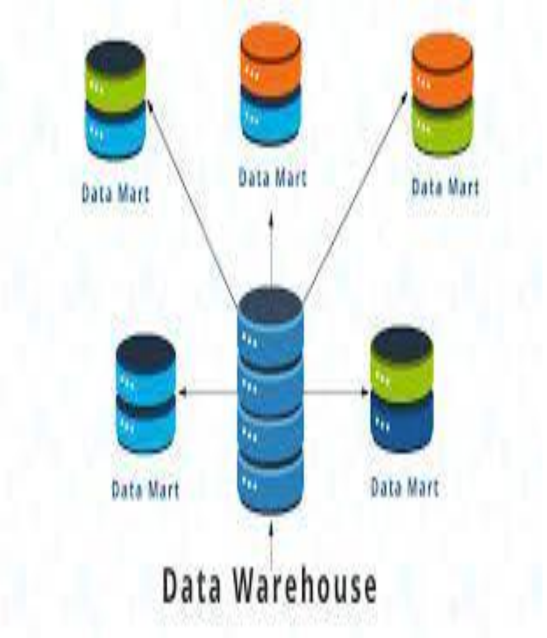
(or)

A Data Mart is focused on a single functional area of an organization and contains a subset of data stored in a Data Warehouse.

Depending on the source of data, data marts can be categorized as independent or dependent.

An independent data mart is a stand-alone system—created without the use of a data warehouse—that focuses on one subject area or business function.

Dependent data marts are sourced directly from enterprise data warehouses.



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.

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

	Data Mart	Data Warehouse
Size	< 100 GB	100 GB +
Subject	Single Subject	Multiple Subjects
Scope	Line-of-Business	Enterprise-wide
Data Sources	Few Sources	Many Source Systems
Data Integration	One Subject Area	All Business Data
Time to Build	Minutes, Weeks, Months	Many Months to Years

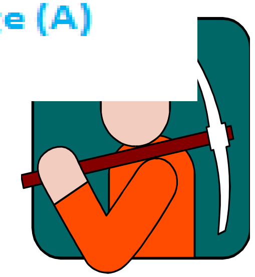
Outline

- Motivation: Why data mining?
- What is data mining?
- Data Mining: On what kind of data?
- Data mining functionality: What kinds of Patterns Can Be Mined?
- Classification of DM: Which technologies are used
- Which kinds of applications are targeted
- Major issues in data mining

1.1 Why Data Mining?

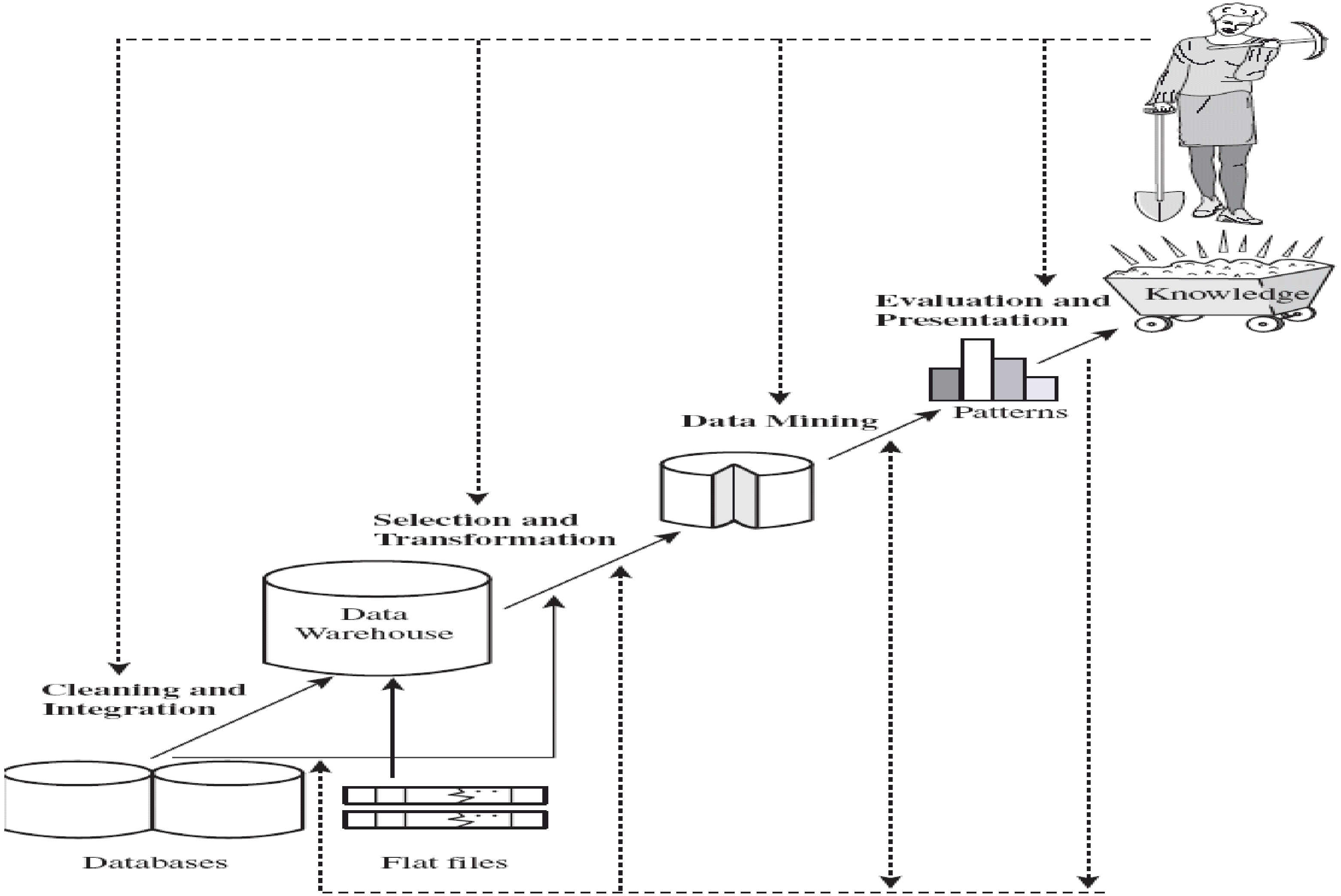
- The Explosive Growth of Data: from terabytes(1000^4) to yottabytes(1000^8)
  - Data collection and data availability
    - Automated data collection tools, database systems, web
  - Major sources of abundant data
    - Business: Web, e-commerce, transactions, stocks, …
    - Science: bioinformatics, scientific simulation, medical research …
    - Society and everyone: news, digital cameras, …
- Data rich but information poor!
  - What does those data mean?
  - How to analyze data?
- Data mining — Automated analysis of massive data sets

1.2 What Is Data Mining?



- Data mining (knowledge discovery from data)
  - Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data
  - Data mining: a misnomer?
- Alternative names
  - Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.

Knowledge Discovery (KDD) Process

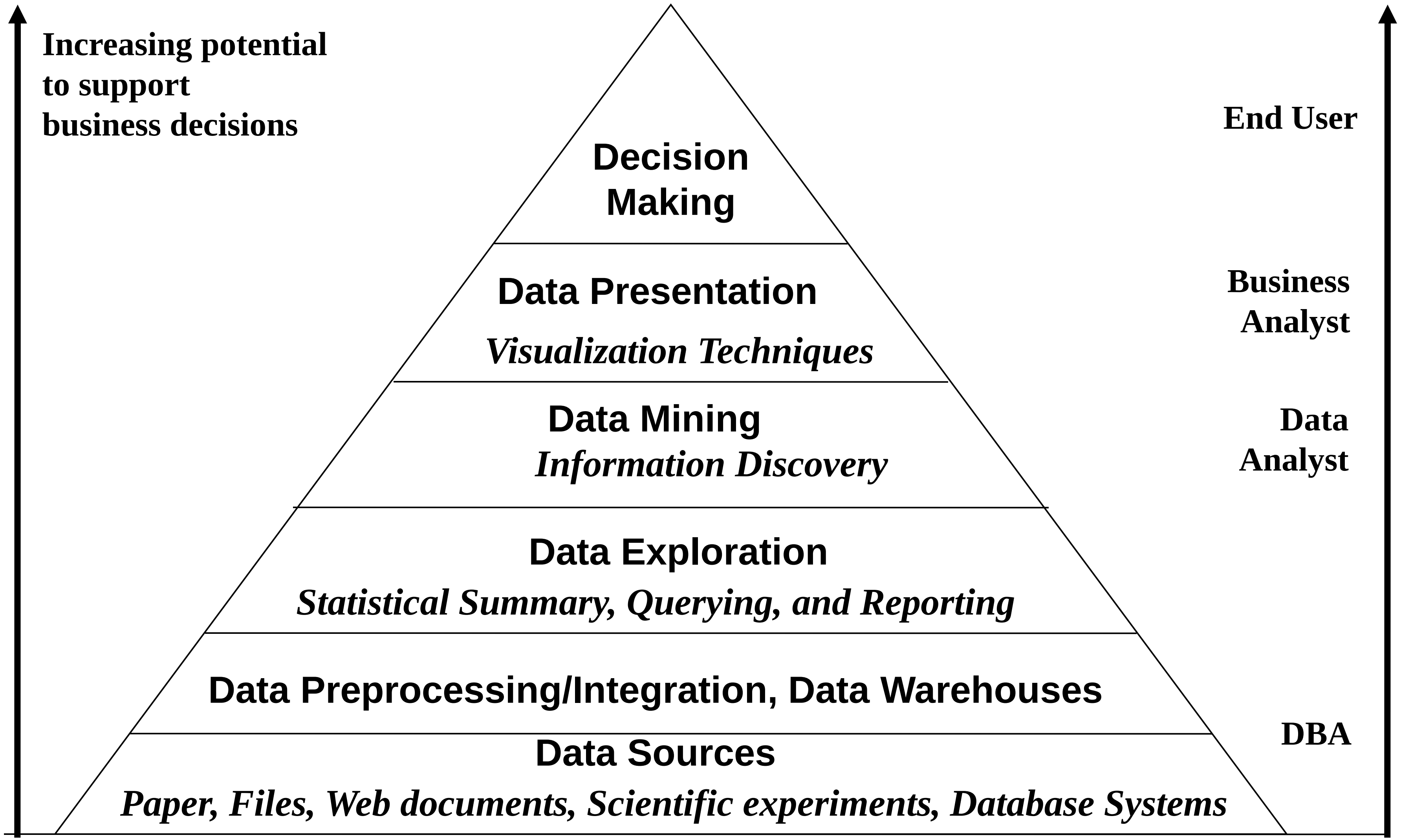


KDD Process: Several Key Steps

- Learning the application domain
  - relevant prior knowledge and goals of application
- Identifying a target data set: data selection
- Data processing
  - **Data cleaning** (remove noise and inconsistent data)
  - **Data integration** (multiple data sources maybe combined)
  - **Data selection** (data relevant to the analysis task are retrieved from database)
  - **Data transformation** (data transformed or consolidated into forms appropriate for mining)

    (Done with data preprocessing)
  - **Data mining** (an essential process where intelligent methods are applied to extract data patterns)
  - **Pattern evaluation** (identify the truly interesting patterns)
  - **Knowledge presentation** (mined knowledge is presented to the user with visualization or representation techniques)
- Use of discovered knowledge

Data Mining and Business Intelligence



Increasing potential to support business decisions

Decision Making

Data Presentation

Visualization Techniques

Data Mining

Information Discovery

Data Exploration

Statistical Summary, Querying, and Reporting

Data Preprocessing/Integration, Data Warehouses

Data Sources

Paper, Files, Web documents, Scientific experiments, Database Systems

End User

Business Analyst

Data Analyst

DBA

1.3 On What Kinds of Data?

- Database-oriented data sets and applications
  - Relational database, data warehouse, transactional database
- Advanced data sets and advanced applications
  - Object-Relational Databases
  - Temporal Databases, Sequence Databases, Time-Series databases
  - Spatial Databases and Spatiotemporal Databases
  - Text databases and Multimedia databases
  - Heterogeneous Databases and Legacy Databases
  - Data Streams
  - The World-Wide Web

Relational Databases

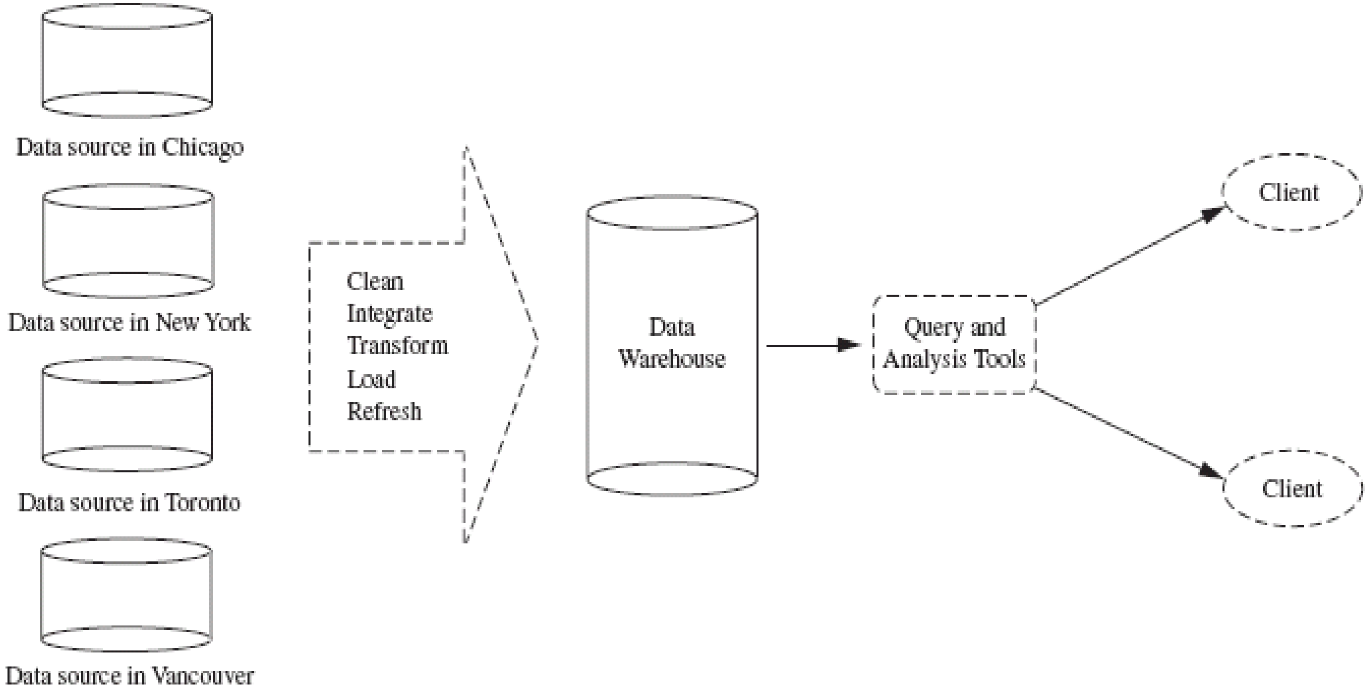
- DBMS – database management system, contains a collection of interrelated databases

  e.g. Faculty database, student database, publications database
- Each database contains a collection of tables and functions to manage and access the data.

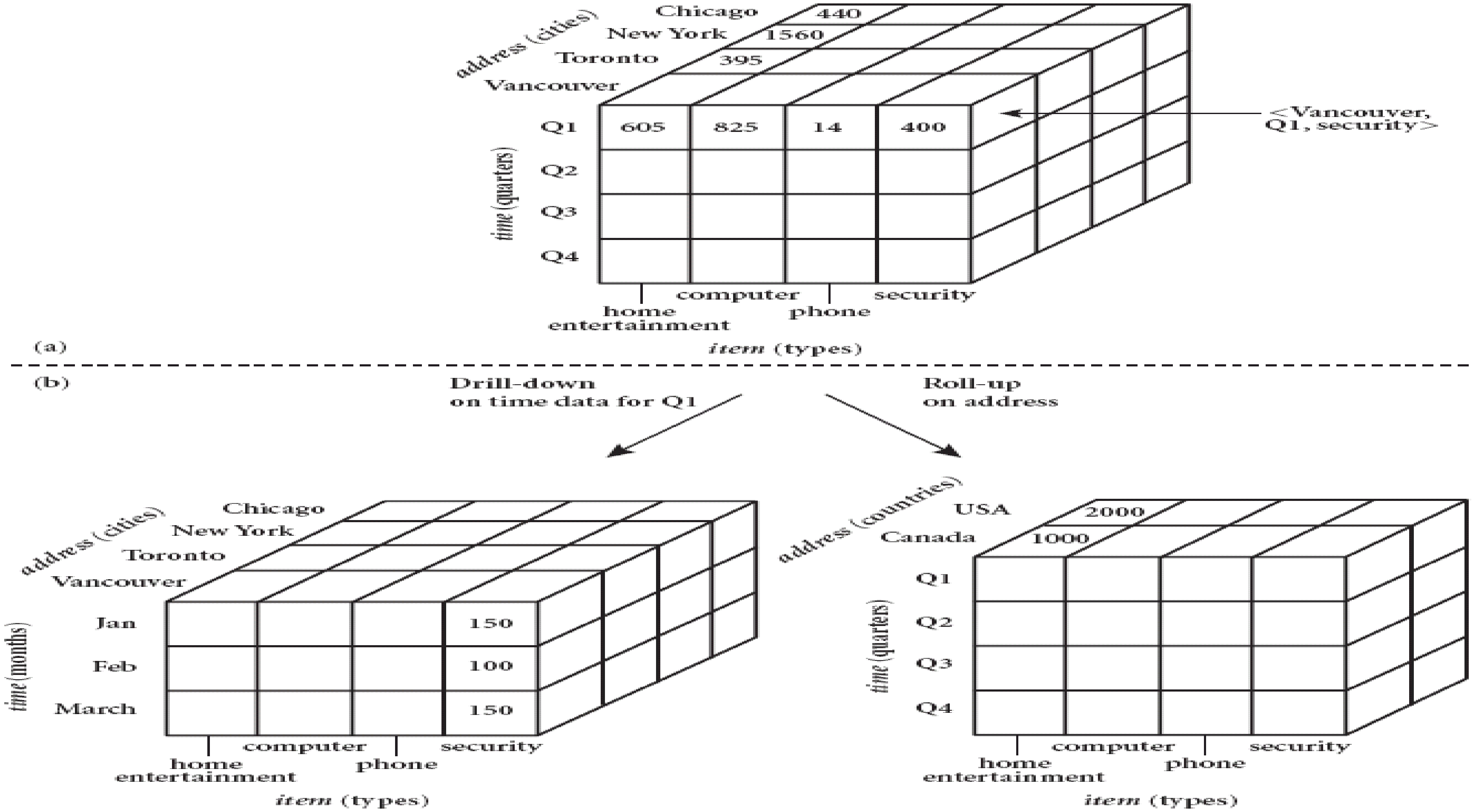
  e.g. student_bio, student_graduation, student_parking
- Each table contains columns and rows, with columns as attributes of data and rows as records.
- Tables can be used to represent the relationships between or among multiple tables.

Data Warehouses

- A repository of information collected from multiple sources, stored under a unified schema, and that usually resides at a single site.
- Constructed via a process of data cleaning, data integration, data transformation, data loading and periodic data refreshing.

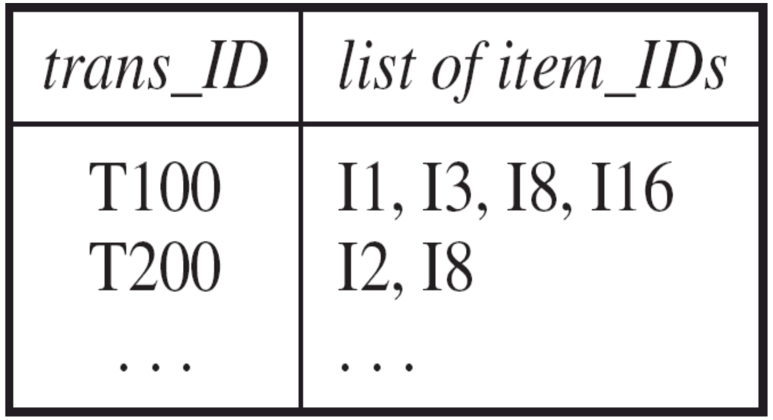


- Data are organized around major subjects, e.g. customer, item, supplier and activity.
- Provide information from a historical perspective (e.g. from the past 5 – 10 years)
- Typically summarized to a higher level (e.g. a summary of the transactions per item type for each store)
- User can perform drill-down or roll-up operation to view the data at different degrees of summarization


address (cities)
Chicago
New York
Toronto
Vancouver
440
1560
395
time (quarters)
Q1
Q2
Q3
Q4
605
825
14
400
<Vancouver, Q1, security>
home entertainment
computer
phone
security
item (types)
(a)
(b)
Drill-down on time data for Q1
Roll-up on address
address (cities)
Chicago
New York
Toronto
Vancouver
time (months)
Jan
Feb
March
150
100
150
home entertainment
computer
phone
security
item (types)
address (countries)
USA
Canada
2000
1000
time (quarters)
Q1
Q2
Q3
Q4
home entertainment
computer
phone
security
item (types)

Transactional Databases

- Consists of a file where each record represents a transaction
- A transaction typically includes a unique transaction ID and a list of the items making up the transaction.



trans_ID	*list of item_IDs*
T100	I1, I3, I8, I16
T200	I2, I8
. . .	. . .

- Either stored in a flat file or unfolded into relational tables
- Easy to identify items that are frequently sold together

Other Kinds of Data

- Besides relational database data, data warehouse data, and transaction data, there are many other kinds of data that have versatile forms and structures and rather different semantic meanings.
- time-related or sequence data (e.g., historical records, stock exchange data, and time-series and biological sequence data).
- data streams (e.g., video surveillance and sensor data, which are continuously transmitted).
- spatial data (e.g., whether data that is collected for a variety of geographical locations),
- engineering design data (e.g., the design of buildings, system components, or integrated circuits)
- hypertext and multimedia data (including text, image, video, and audio data)
- graph and networked data(e.g., social and information networks), and the Web (a huge, widely distributed information repository made available by the Internet).

1.4 Data Mining Functionalities

– What kinds of patterns can be mined?

- Concept/Class Description: Characterization and Discrimination
  - Data can be associated with classes or concepts.
    - E.g. classes of items – computers, printers, …

      concepts of customers – bigSpenders, budgetSpenders, …
    - How to describe these items or concepts?

– Descriptions can be derived via

- Data characterization – summarizing the general characteristics of a target class of data.
  - – E.g. summarizing the characteristics of customers who spend more than $1,000 a year

    at *AllElectronics*. Result can be a general profile of the customers, such as 40 – 50 years old, employed, have excellent credit ratings.
- Data discrimination – comparing the target class with one or a set of comparative classes
  - – E.g. Compare the general features of software products whole sales increase by 10% in the last year with those whose sales decrease by 30% during the same period
- Or both of the above

- Mining Frequent Patterns, Associations and Correlations
  - Frequent itemed: a set of items that frequently appear together in a transactional data set (e.g. milk and bread)
  - Frequent subsequence: a pattern that customers tend to purchase product A, followed by a purchase of product B
  - Association Analysis: find frequent patterns
    - E.g. a sample analysis result – an association rule:

      buys(X, “computer”) => buys(X, “software”) [support = 1%, confidence = 50%]

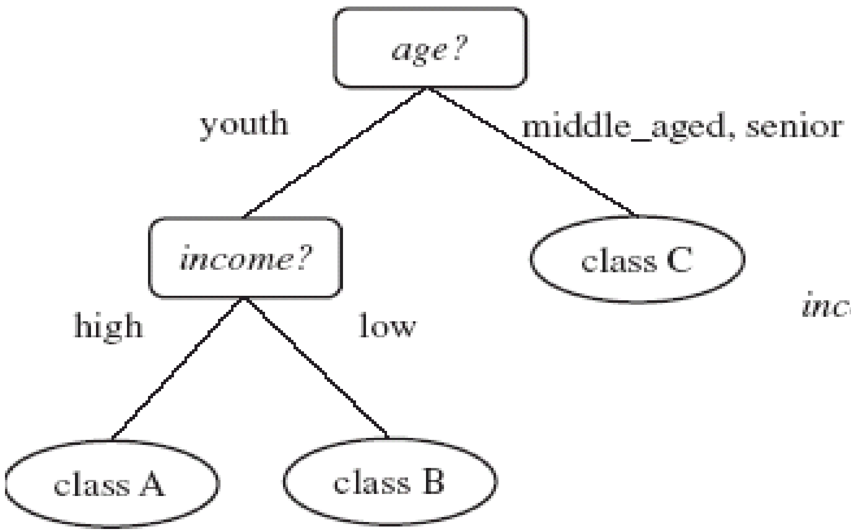
      (if a customer buys a computer, there is a 50% chance that she will buy software. 1% of all of the transactions under analysis showed that computer and software are purchased together.)
    - Associations rules are discarded as uninteresting if they do not satisfy both a minimum support threshold and a minimum confidence threshold.
  - Correlation Analysis: additional analysis to find statistical correlations between associated pairs

- Classification and Prediction
  - Classification
    - The process of finding a model that describes and distinguishes the data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown.
    - The derived model is based on the analysis of a set of training data (data objects whose class label is known).
    - The model can be represented in *classification (IF-THEN) rules,* decision trees, *neural networks,* etc.
  - Prediction
    - Predict missing or unavailable numerical data values

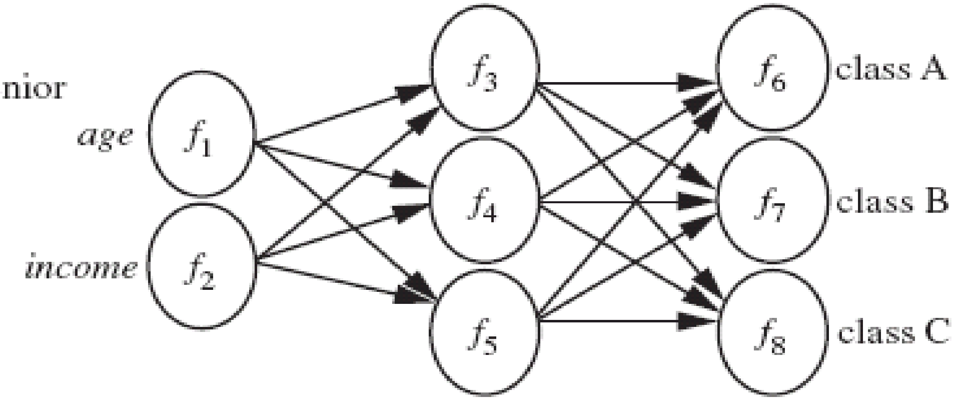
(a)

age(X, "youth") AND income(X, "high") → class(X, "A")
age(X, "youth") AND income(X, "low") → class(X, "B")
age(X, "middle_aged") → class(X, "C")
age(X, "senior") → class(X, "C")

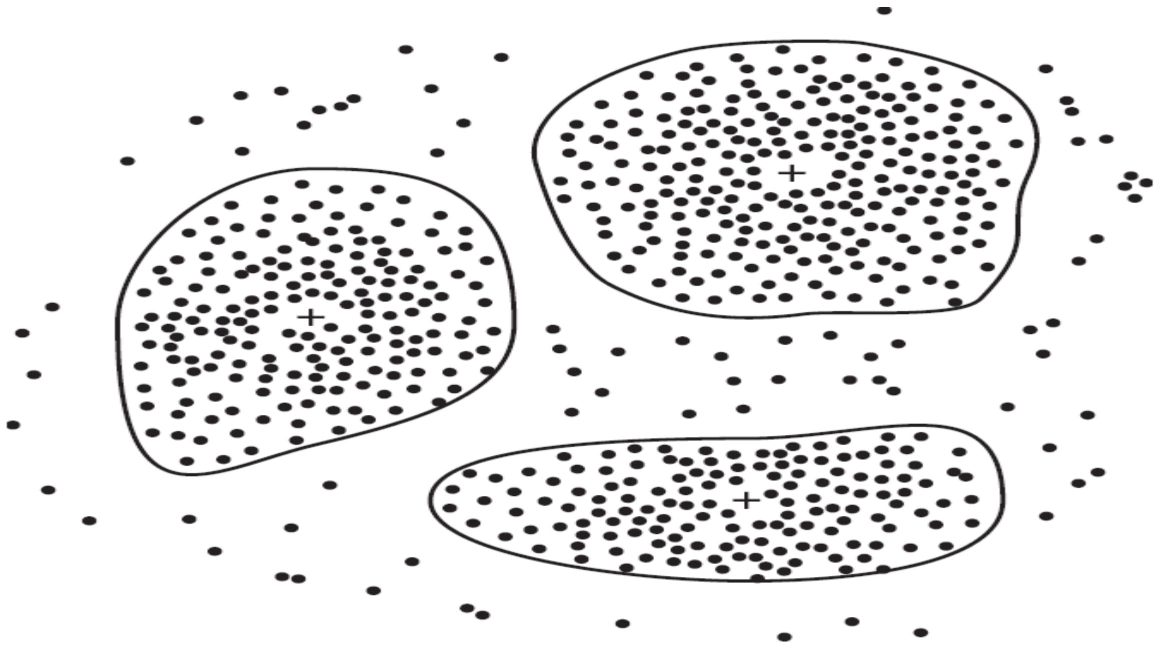
(b)



(c)

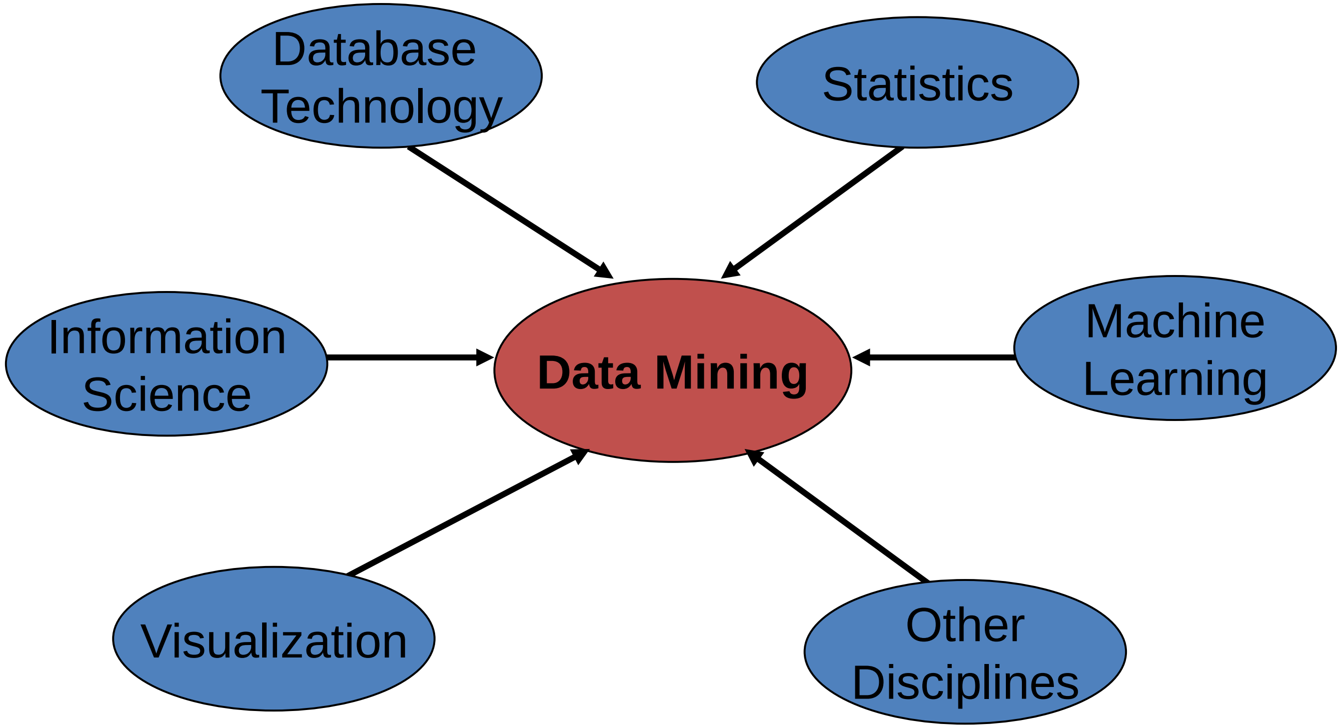


- Cluster Analysis
  - Class label is unknown: group data to form new classes
  - Clusters of objects are formed based on the principle of *maximizing intra-class similarity & minimizing interclass similarity*
    - E.g. Identify homogeneous subpopulations of customers. These clusters may represent individual target groups for marketing.



- Outlier Analysis
  - Data that do no comply with the general behavior or model.
  - Outliers are usually discarded as noise or exceptions.
  - Useful for fraud detection.
    - E.g. Detect purchases of extremely large amounts
- Evolution Analysis
  - Describes and models regularities or trends for objects whose behavior changes over time.
    - E.g. Identify stock evolution regularities for overall stocks and for the stocks of particular companies.

1.6 Classification of data mining systems



- **Data to be mined**
  - Relational, data warehouse, transactional, stream, object-oriented/relational, active, spatial, time-series, text, multi-media, heterogeneous, legacy, WWW
- **Knowledge to be mined**
  - Characterization, discrimination, association, classification, clustering, trend/deviation, outlier analysis, etc.
  - Multiple/integrated functions and mining at multiple levels
- **Techniques utilized**
  - Database-oriented, data warehouse (OLAP), machine learning, statistics, visualization, etc.
- **Applications adapted**
  - Retail, telecommunication, banking, fraud analysis, bio-data mining, stock market analysis, text mining, Web mining, etc.

Potential Applications

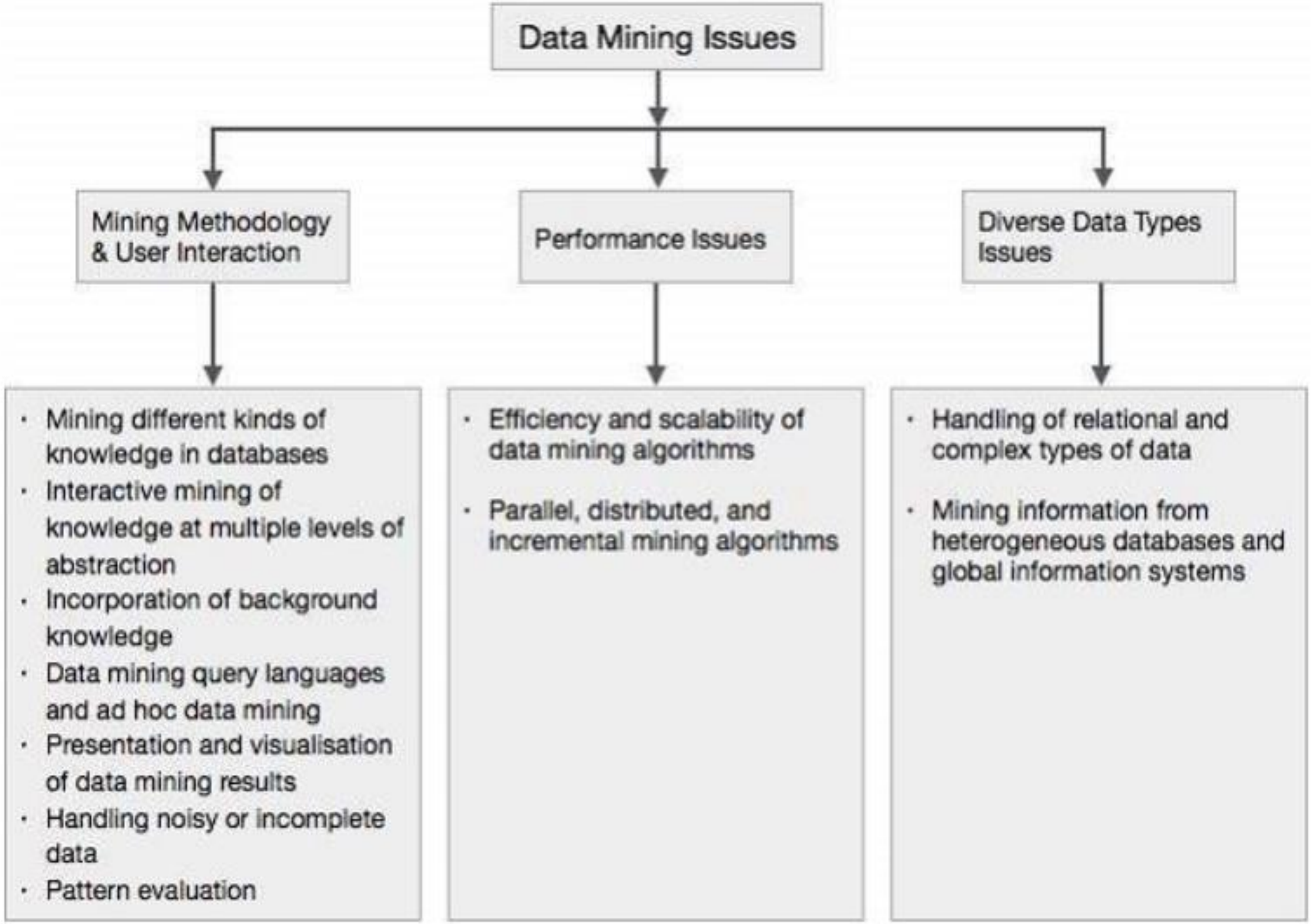
- Data analysis and decision support
  - Market analysis and management
    - Target marketing, customer relationship management (CRM), market basket analysis, cross selling, market segmentation
  - Risk analysis and management
    - Forecasting, customer retention, improved underwriting, quality control, competitive analysis
  - Fraud detection and detection of unusual patterns (outliers)
- Other Applications
  - Text mining (news group, email, documents) and Web mining
  - Stream data mining
  - Bioinformatics and bio-data analysis

Ex.: Market Analysis and Management

- Where does the data come from?—Credit card transactions, loyalty cards, discount coupons, customer complaint calls, surveys …
- Target marketing
  - Find clusters of "model" customers who share the same characteristics: interest, income level, spending habits, etc.,
    - E.g. Most customers with income level 60k – 80k with food expenses $600 - $800 a month live in that area
  - Determine customer purchasing patterns over time
    - E.g. Customers who are between 20 and 29 years old, with income of 20k – 29k usually buy this type of CD player
- Cross-market analysis—Find associations/co-relations between product sales, & predict based on such association
  - E.g. Customers who buy computer A usually buy software B

- Customer requirement analysis
  - Identify the best products for different customers
  - Predict what factors will attract new customers
- Provision of summary information
  - Multidimensional summary reports
    - E.g. Summarize all transactions of the first quarter from three different branches
      Summarize all transactions of last year from a particular branch
      Summarize all transactions of a particular product
  - Statistical summary information
    - E.g. What is the average age for customers who buy product A?
- Fraud detection
  - Find outliers of unusual transactions
- Financial planning
  - Summarize and compare the resources and spending

1.9 Major Issues in Data Mining



Data Mining Issues

Mining Methodology & User Interaction

- Mining different kinds of knowledge in databases
- Interactive mining of knowledge at multiple levels of abstraction
- Incorporation of background knowledge
- Data mining query languages and ad hoc data mining
- Presentation and visualisation of data mining results
- Handling noisy or incomplete data
- Pattern evaluation

Performance Issues

- Efficiency and scalability of data mining algorithms
- Parallel, distributed, and incremental mining algorithms

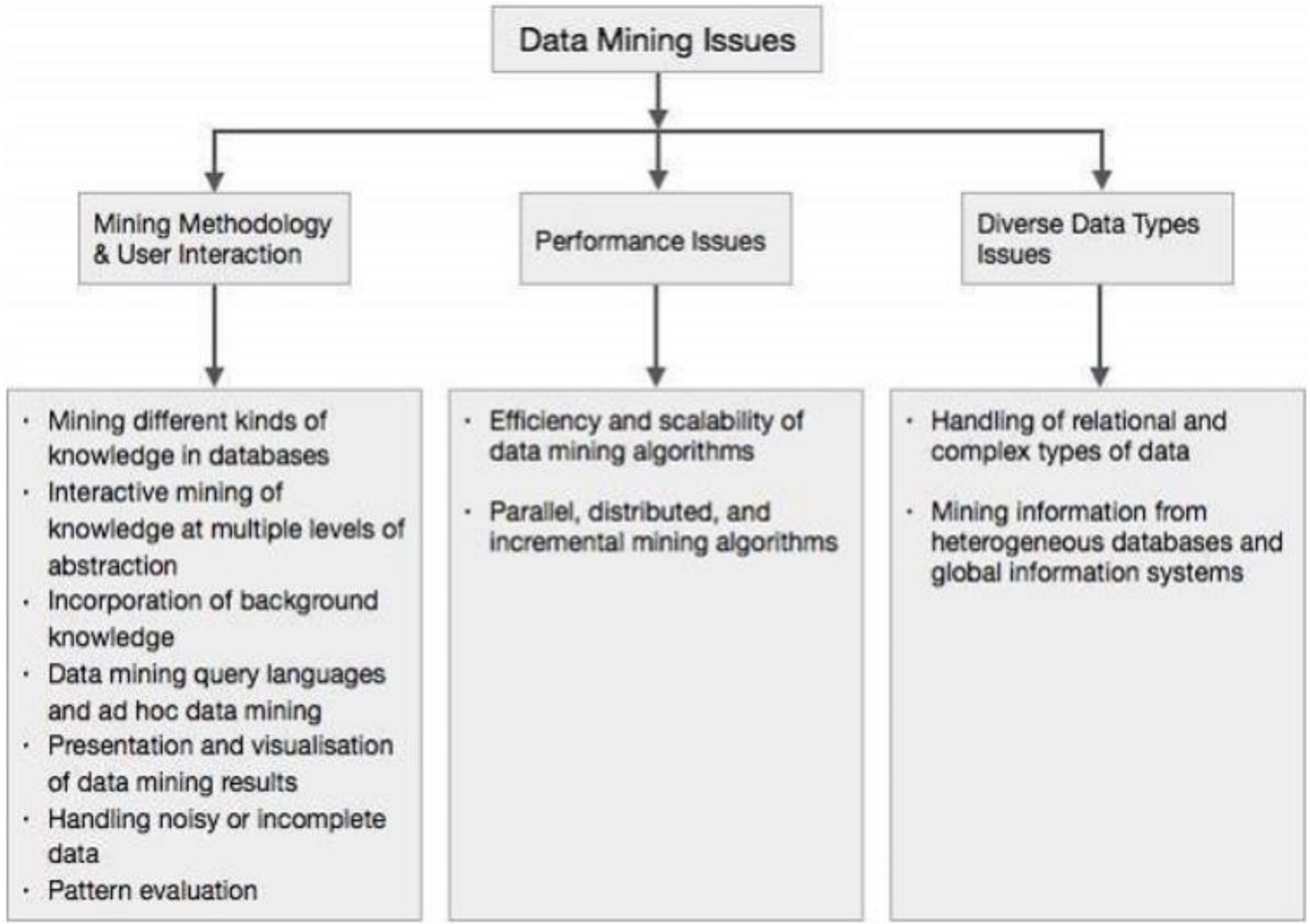
Diverse Data Types Issues

- Handling of relational and complex types of data
- Mining information from heterogeneous databases and global information systems

- Mining methodology and User interaction
  - Mining different kinds of knowledge
    - DM should cover a wide spectrum of data analysis and knowledge discovery tasks
    - Enable to use the database in different ways
    - Require the development of numerous data mining techniques
  - Interactive mining of knowledge at multiple levels of abstraction
    - Difficult to know exactly what will be discovered
    - Allow users to focus the search, refine data mining requests
  - Incorporation of background knowledge
    - Guide the discovery process
    - Allow discovered patterns to be expressed in concise terms and different levels of abstraction
  - Data mining query languages and ad hoc data mining
    - High-level query languages need to be developed
    - Should be integrated with a DB/DW query language

- Presentation and visualization of results
  - Knowledge should be easily understood and directly usable
  - High level languages, visual representations or other expressive forms
  - Require the DM system to adopt the above techniques
- Handling noisy or incomplete data
  - Require data cleaning methods and data analysis methods that can handle noise
- Pattern evaluation – the interestingness problem
  - How to develop techniques to access the interestingness of discovered patterns, especially with subjective measures bases on user beliefs or expectations

- Performance Issues
  - Efficiency and scalability
    - Huge amount of data
    - Running time must be predictable and acceptable
  - Parallel, distributed and incremental mining algorithms
    - Divide the data into partitions and processed in parallel
    - Incorporate database updates without having to mine the entire data again from scratch
- Diversity of Database Types
  - Other database that contain complex data objects, multimedia data, spatial data, etc.
  - Expect to have different DM systems for different kinds of data
  - Heterogeneous databases and global information systems
    - Web mining becomes a very challenging and fast-evolving field in data mining



Data Mining Issues

Mining Methodology & User Interaction

- Mining different kinds of knowledge in databases
- Interactive mining of knowledge at multiple levels of abstraction
- Incorporation of background knowledge
- Data mining query languages and ad hoc data mining
- Presentation and visualisation of data mining results
- Handling noisy or incomplete data
- Pattern evaluation

Performance Issues

- Efficiency and scalability of data mining algorithms
- Parallel, distributed, and incremental mining algorithms

Diverse Data Types Issues

- Handling of relational and complex types of data
- Mining information from heterogeneous databases and global information systems

Mining Methodology and User Interaction Issues

•**Mining different kinds of knowledge in databases** − Different users may be interested in different kinds of knowledge. Therefore it is necessary for data mining to cover a broad range of knowledge discovery task.

•**Interactive mining of knowledge at multiple levels of abstraction** − The data mining process needs to be interactive because it allows users to focus the search for patterns, providing and refining data mining requests based on the returned results.

•**Incorporation of background knowledge** − To guide discovery process and to express the discovered patterns, the background knowledge can be used. Background knowledge may be used to express the discovered patterns not only in concise terms but at multiple levels of abstraction.

- **Data mining query languages and ad hoc data mining** − Data Mining Query language that allows the user to describe ad hoc mining tasks, should be integrated with a data warehouse query language and optimized for efficient and flexible data mining.
- **Presentation and visualization of data mining results** − Once the patterns are discovered it needs to be expressed in high level languages, and visual representations. These representations should be easily understandable.
- **Handling noisy or incomplete data** − The data cleaning methods are required to handle the noise and incomplete objects while mining the data regularities. If the data cleaning methods are not there then the accuracy of the discovered patterns will be poor.
- **Pattern evaluation** − The patterns discovered should be interesting because either they represent common knowledge or lack novelty.

Performance Issues

- **Efficiency and scalability of data mining algorithms** − In order to effectively extract the information from huge amount of data in databases, data mining algorithm must be efficient and scalable.
- **Parallel, distributed, and incremental mining algorithms** − The factors such as huge size of databases, wide distribution of data, and complexity of data mining methods motivate the development of parallel and distributed data mining algorithms. These algorithms divide the data into partitions which is further processed in a parallel fashion. Then the results from the partitions is merged. The incremental algorithms, update databases without mining the data again from scratch.

Diverse **Data Types Issues**

•**Handling of relational and complex types of data** − The database may contain complex data objects, multimedia data objects, spatial data, temporal data etc. It is not possible for one system to mine all these kind of data.

•**Mining information from heterogeneous databases and global information systems** − The data is available at different data sources on LAN or WAN. These data source may be structured, semi structured or unstructured. Therefore mining the knowledge from them adds challenges to data mining.