KTU NOTES

Module – 1 (Introduction to Data Mining and Data Warehousing)

[Data warehouse-Differences between Operational Database Systems and Data Warehouses, Multidimensional data model- Warehouse schema, OLAP Operations, Data Warehouse Architecture, Data Warehousing to Data Mining, Data Mining Concepts and Applications, Knowledge Discovery in Database Vs Data mining, Architecture of typical data mining system, Data Mining Functionalities, Data Mining Issues.]

Data Warehouse: Data warehousing provides architectures and tools for business executives to systematically organize, understand, and use their data to make strategic decisions. Data warehouse refers to a database that is maintained separately from an organization's operational databases. A data warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data in support of management's decision-making process"

A data warehouse focuses on the modelling and analysis of data for decision makers (not on day-to-day transaction).

Provide a simple and concise view around particular subject issues by excluding data that are not useful in the decision support process.

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

2. Time-variant: Data are stored to provide information from a historical perspective

Every key structure in the data warehouse contains, either implicitly or explicitly, an element of time.

3. Non-volatile: 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.

Data warehousing is the process of constructing 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.

Difference between Operational Database systems and Data Warehouse

Operational Database systems

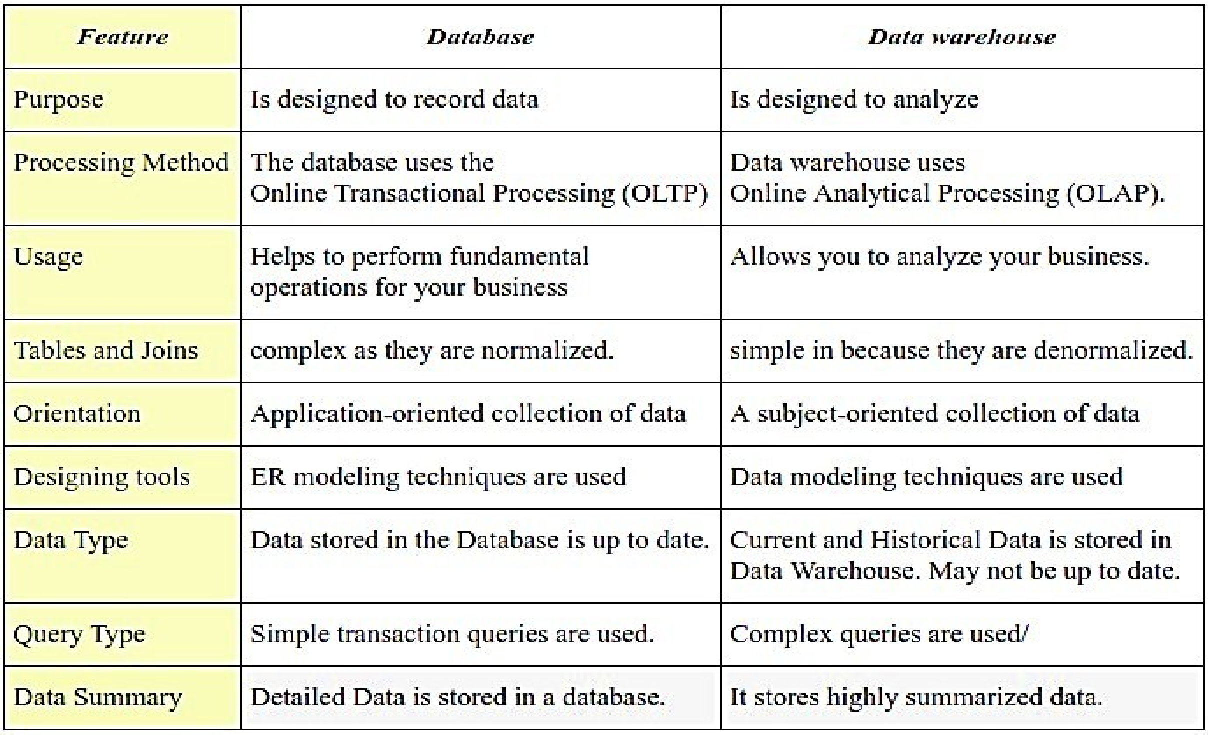
- Main task is to perform on-line transaction and query processing. These systems are called on-line transaction processing (OLTP) systems.

□ They cover most of the day-to-day operations of an organization, such as purchasing, inventory, manufacturing, banking, payroll, registration, and accounting.

Data Warehouse

□ 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 the different users. These systems are known as on-line analytical processing (OLAP) systems.



Feature	*Database*	*Data warehouse*
Purpose	Is designed to record data	Is designed to analyze
Processing Method	The database uses the Online Transactional Processing (OLTP)	Data warehouse uses Online Analytical Processing (OLAP).
Usage	Helps to perform fundamental operations for your business	Allows you to analyze your business.
Tables and Joins	complex as they are normalized.	simple in because they are denormalized.
Orientation	Application-oriented collection of data	A subject-oriented collection of data
Designing tools	ER modeling techniques are used	Data modeling techniques are used
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.
Query Type	Simple transaction queries are used.	Complex queries are used/
Data Summary	Detailed Data is stored in a database.	It stores highly summarized data.

Difference between OLTP and OLAP

1. Users and system orientation:

□ OLTP system is customer-oriented and is used for transaction and query processing by clerks, clients, and information technology professionals.

□ OLAP system is market-oriented and is used for data analysis by knowledge workers, including managers, executives, and analysts.

2. Data contents:

□ OLTP system manages current data

□ OLAP system manages large amounts of historical data, provides facilities for summarization and aggregation, and stores and manages information at different levels of granularity.

3. 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 snowflake model and a subjectoriented database design.

4. View:

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

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

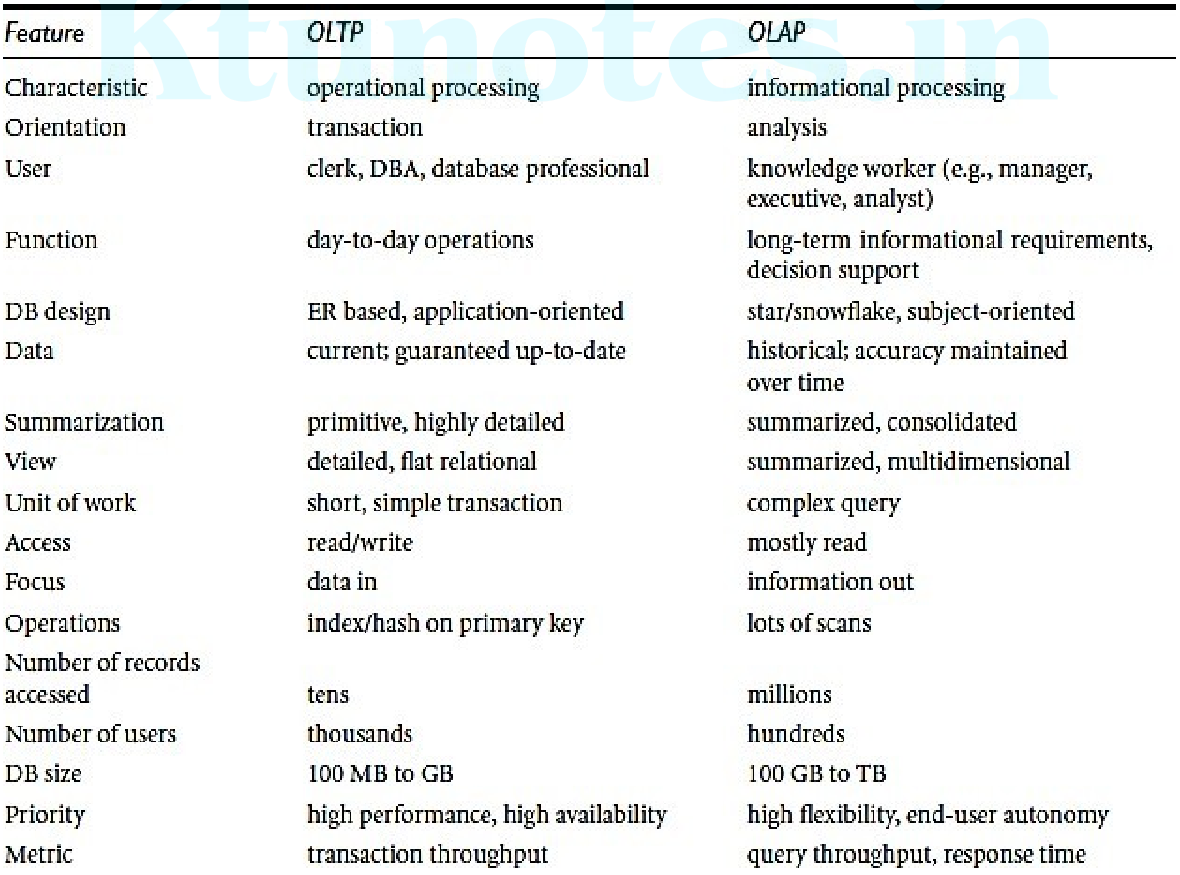
□ OLAP systems also deal with information that originates from different organizations.

□ OLAP data are stored on multiple storage media.

5. Access patterns:

□ The access patterns of an OLTP system consist mainly of short, atomic transactions. Such a system requires concurrency control and recovery mechanisms.

Accesses to OLAP systems are mostly read-only operations although many could be complex queries.



Feature	*OLTP*	*OLAP*
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA, database professional	knowledge worker (e.g., manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements, decision support
DB design	ER based, application-oriented	star/snowflake, subject-oriented
Data	current; guaranteed up-to-date	historical; accuracy maintained over time
Summarization	primitive, highly detailed	summarized, consolidated
View	detailed, flat relational	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read
Focus	data in	information out
Operations	index/hash on primary key	lots of scans
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	100 MB to GB	100 GB to TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

Multidimensional data model

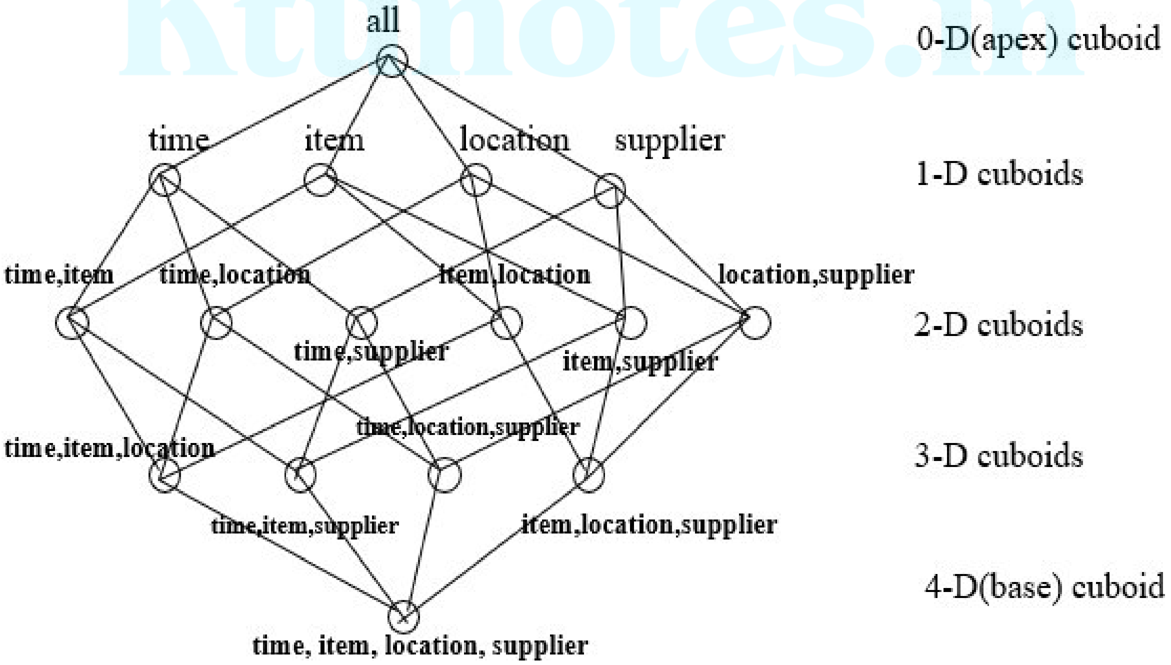
Data warehouses and OLAP tools are based on a multidimensional data model. This model views data in the form of a data cube.

A data cube allows data to be modelled and viewed in multiple dimensions. It is defined by dimensions and facts.

- Dimension tables, such as item (item_name, brand, type), or time(day, week, month, quarter, year)
- Fact table contains measures (such as dollars_sold) and keys to each of the related dimension tables

In data warehousing literature, an n-D base cube is called a base cuboid. The top most 0-D cuboid, which holds the highest-level of summarization, is called the apex cuboid. The lattice of cuboids forms a data cube.

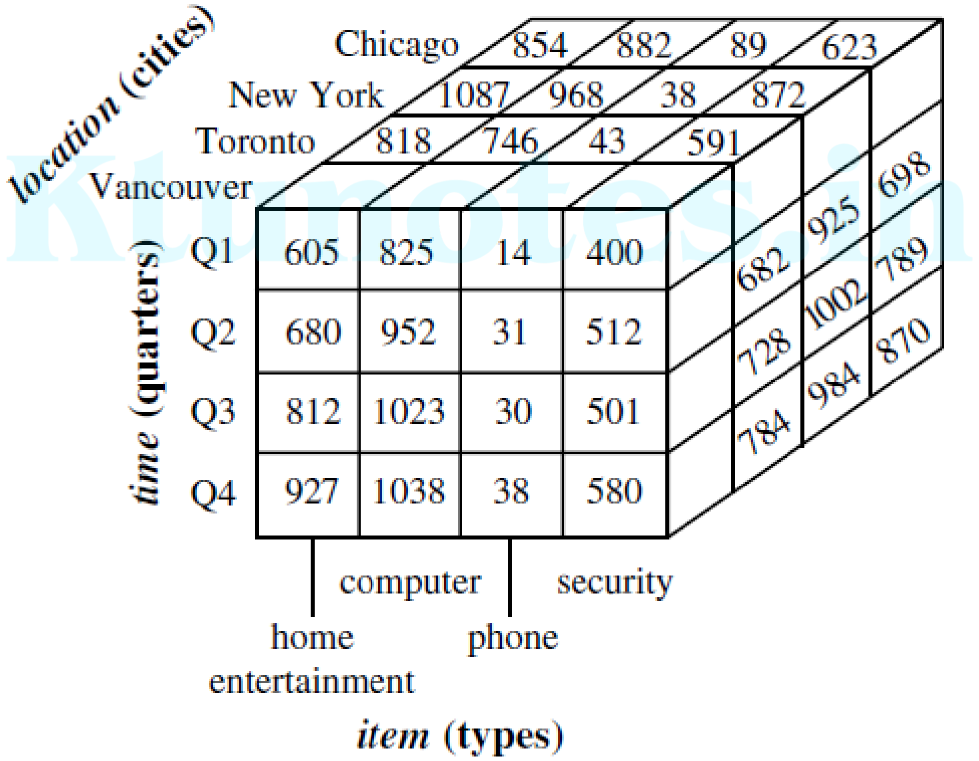
Cube: A Lattice of Cuboids



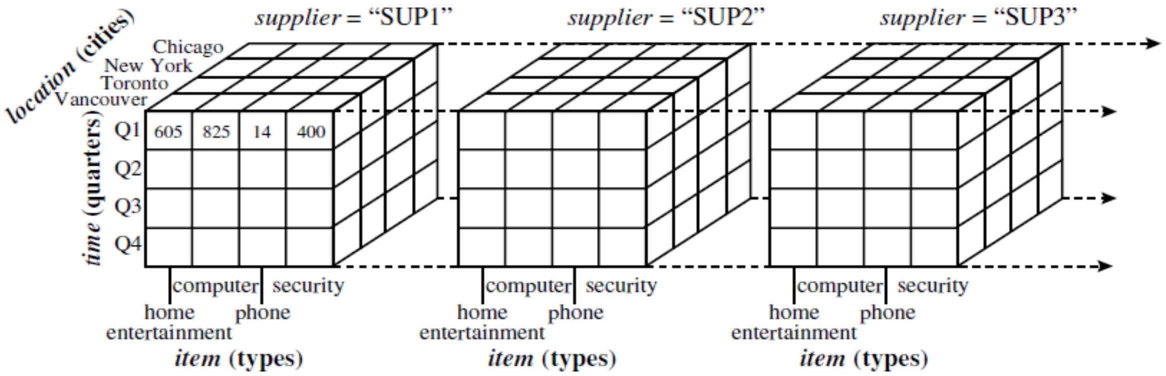
Example

Table 3.3 A 3-D view of sales data for *AllElectronics*, according to the dimensions *time*, *item*, and *location*. The measure displayed is *dollars_sold* (in thousands).

	location = "Chicago"				*location* = "New York"				*location* = "Toronto"				*location* = "Vancouver"			
	item				*item*				*item*				*item*			
time	*home ent.*	*comp.*	*phone*	*sec.*	*home ent.*	*comp.*	*phone*	*sec.*	*home ent.*	*comp.*	*phone*	*sec.*	*home 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



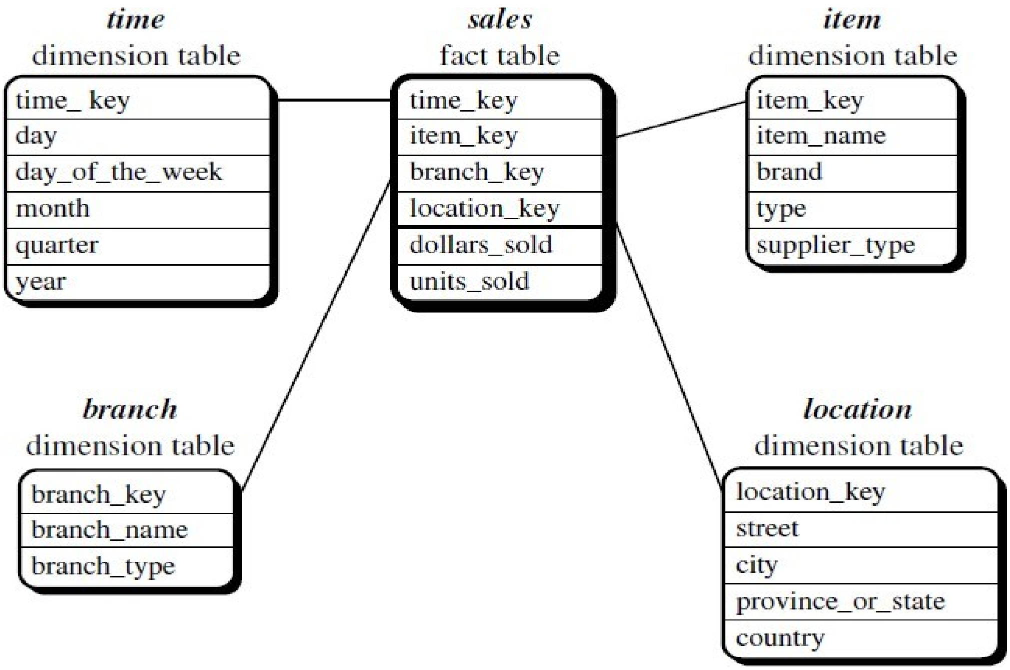
A 4-D data cube representation of sales data, according to the dimensions time, item, location, and supplier. The measure displayed is dollars sold (in thousands).



Warehouse schema

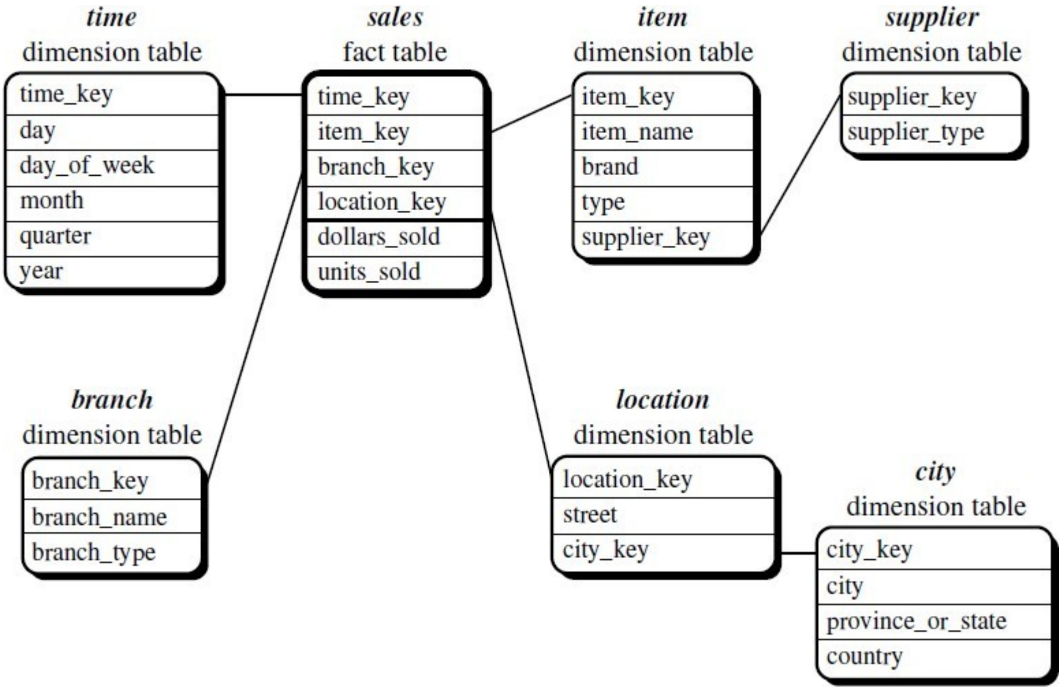
The most popular data model for a data warehouse is a multidimensional model. Such a model can exist in the form of a star schema, a snowflake schema, or a fact constellation schema.

1. 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
2. graph resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.

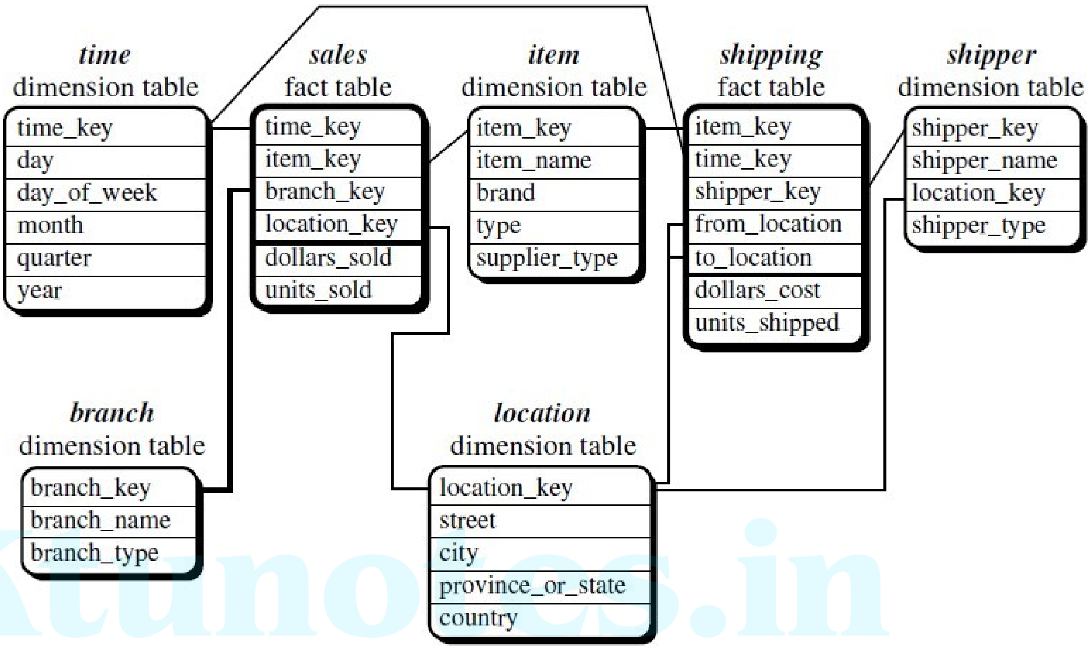


Notice that in the star schema, each dimension is represented by only one table, and each table contains a set of attributes. For example, the location dimension table contains the attribute set *location key, street, city, province or state, country*. This constraint may introduce some redundancy. For example, *"Vancouver"* and "*Victoria"* are both cities in the Canadian province of British Columbia.

3. Snowflake schema : 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.
   Here, the sales fact table is identical to that of the star schema in Figure. The main difference between the two schemas is in the definition of dimension tables. The single dimension table for item in the star schema is normalized in the snowflake schema, resulting in new item and supplier tables. For example, the item dimension table now contains the attributes *item key, item name, brand, type, and supplier key*, where supplier key is linked to the supplier dimension table, containing supplier key and supplier type information.



4. 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.



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.
2. Drill-down: Drill-down is the reverse of roll-up. It navigates from less detailed data to more detailed data.
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.
4. Pivot (rotate): Pivot (also called rotate) is a visualization operation that rotates the data axes in view in order to provide an alternative presentation of the data.

5. 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.

Data Warehouse Architecture

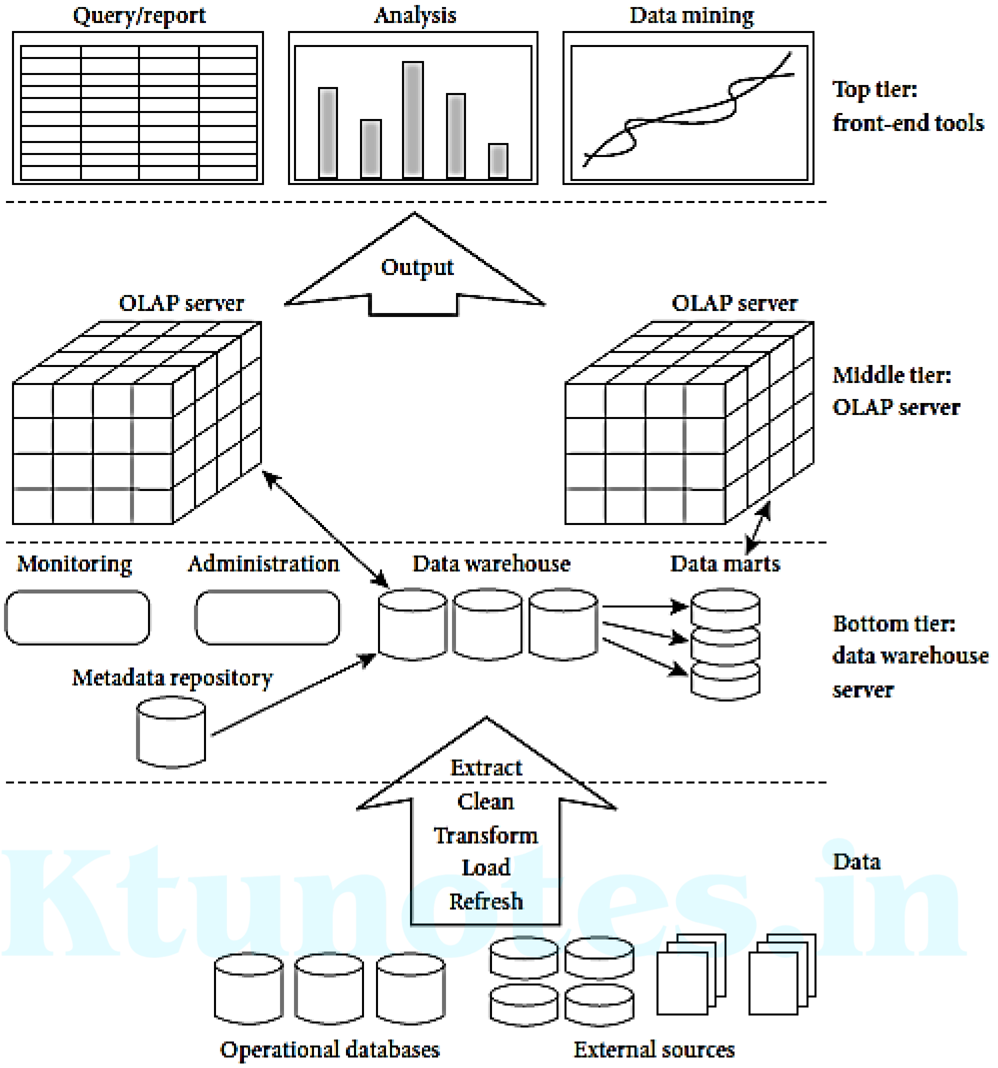
Four views regarding the design of a data warehouse

- Top-down view: allows selection of the relevant information necessary for the data warehouse
- Data source view: exposes the information being captured, stored, and managed by operational systems
- Data warehouse view: consists of fact tables and dimension tables
- Business query view : sees the perspectives of data in the warehouse from the view of end-user

Data Warehouse Design Process

- Top-down, bottom-up approaches or a combination of both
  - Top-down: Starts with overall design and planning (mature)
  - Bottom-up: Starts with experiments and prototypes (rapid)
- From software engineering point of view
  - Waterfall: structured and systematic analysis at each step before proceeding to the next
  - Spiral: rapid generation of increasingly functional systems, short turn around time, quick turn around
- Typical data warehouse design process
  - Choose a business process to model, e.g., orders, invoices, etc.
  - Choose the *grain* (*atomic level of data*) of the business process
  - Choose the dimensions that will apply to each fact table record
  - Choose the measure that will populate each fact table record

A Three-Tier Data Warehouse Architecture



1. The bottom tier is a warehouse database server that is almost always a relational database system. Back-end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (such as customer profile information provided by external consultants). These tools and utilities perform data extraction, cleaning, and transformation (e.g., to merge similar data from different sources into a unified format), as well as load and refresh functions to update the data warehouse. The data are extracted using application program interfaces known as gateways. A gateway is supported by the underlying DBMS and allows client programs to generate SQL code to be executed at a server. Examples of gateways include ODBC (Open Database Connection) and OLEDB (Open Linking and Embedding for

Databases) by Microsoft and JDBC (Java Database Connection). This tier also contains a metadata repository, which stores information about the data warehouse and its contents.

2. The middle tier is an OLAP server that is typically implemented using either (1) a relational OLAP (ROLAP) model, that is, an extended relational DBMS that maps operations on multidimensional data to standard relational operations; or (2) a multidimensional OLAP (MOLAP) model, that is, a special-purpose server that directly implements multidimensional data and operations.
3. The top tier is a front-end client layer, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on.

From the architecture point of view, there are three data warehouse models: the enterprise warehouse, the data mart, and the virtual warehouse.

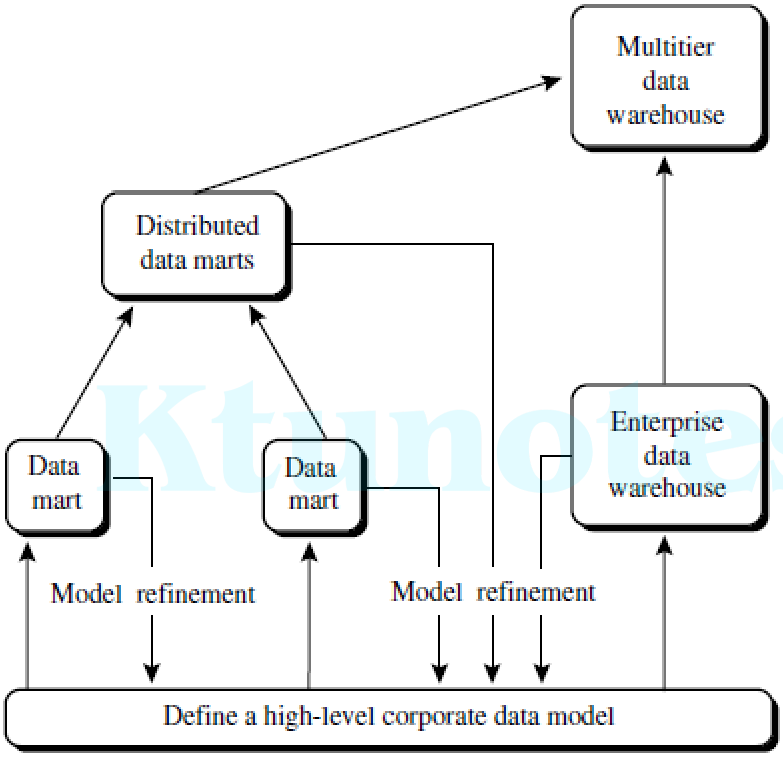
Enterprise warehouse: An enterprise warehouse collects all of the information about subjects spanning the entire organization. It provides corporate-wide data integration, usually from one or more operational systems or external information providers, and is cross-functional in scope. It typically contains detailed data as well as summarized data, and can range in size from a few gigabytes to hundreds of gigabytes, terabytes, or beyond.

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. For example, a marketing data mart may confine its subjects to customer, item, and sales. 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.

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 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.

A recommended approach for data warehouse development



DataWarehouse Back-End Tools and Utilities

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.

Metadata Repository

Metadata are data about data. 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 structure of the data warehouse, 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
- The 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.

OLAP Server Architectures

Relational OLAP (ROLAP) servers: These are the intermediate servers that stand in between a relational back-end server and client front-end tools. They use a relational or extended-relational DBMS to store and manage warehouse data, and OLAP middleware to support missing pieces. ROLAP servers include optimization for each DBMS back end, implementation of aggregation navigation logic, and additional tools and services. ROLAP technology tends to have greater scalability than MOLAP technology. The DSS server of Microstrategy, for example, adopts the ROLAP approach.

Multidimensional OLAP (MOLAP) servers: These servers support multidimensional views of data through array-based multidimensional storage engines. They map multidimensional views directly to data cube array structures. The advantage of using a data cube is that it allows fast indexing to precomputed summarized data.

Hybrid OLAP (HOLAP) servers: The hybrid OLAP approach combines ROLAP and MOLAP technology, benefiting from the greater scalability of ROLAP and

the faster computation of MOLAP. For example, a HOLAP server may allow large volumes of detail data to be stored in a relational database, while aggregations are kept in a separate MOLAP store. The Microsoft SQL Server 2000 supports a hybrid OLAP server.

Specialized SQL servers: To meet the growing demand of OLAP processing in relational databases, some database system vendors implement specialized SQL servers that provide advanced query language and query processing support for SQL queries over star and snowflake schemas in a read-only environment.

Data Mining

Data mining refers to extracting or mining knowledge from large amounts of data. Mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material.

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

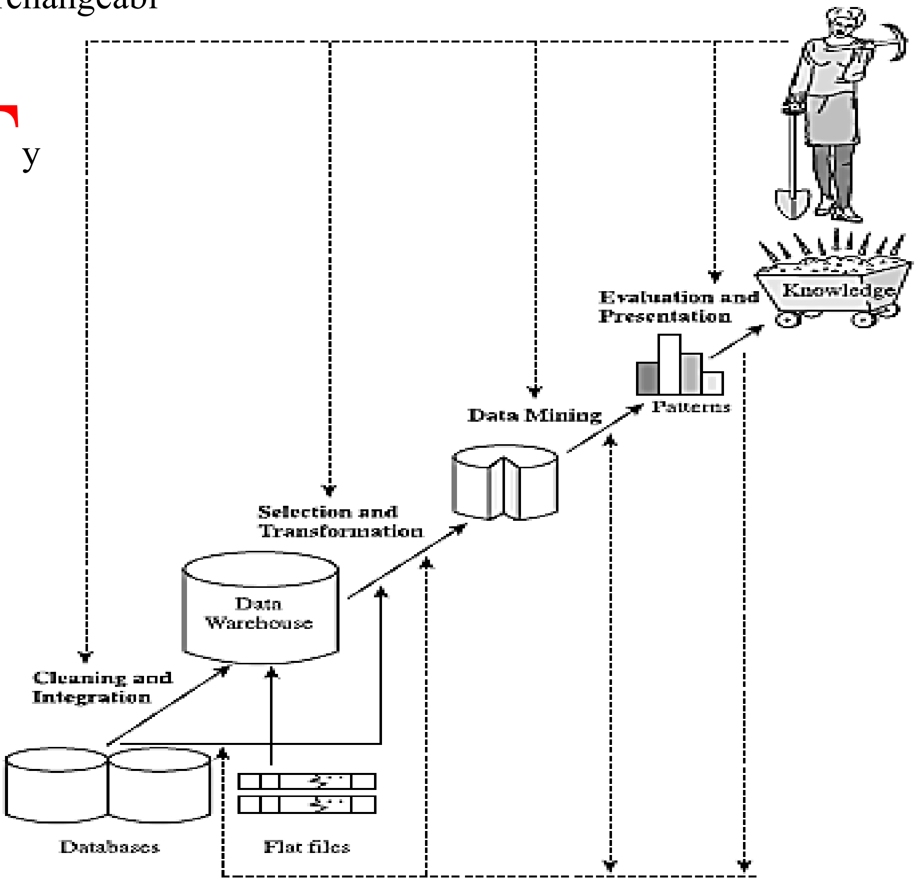
Alternative names: Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.

KDD steps – STEPS IN KNOWLEDGE DISCOVERY FROM DATA

Many people treat data mining as a synonym for another popularly used term, knowledge discovery from data, or KDD, while others view data mining as merely an essential step in the process of knowledge discovery. The terms knowledge discovery in databases (KDD) and data mining are often used

interchangeabl

T y



1. Data cleaning (to remove noise and inconsistent data)

2. Data integration (where multiple data sources may be combined)

3. Data selection (where data relevant to the analysis task are retrieved from the database)

4. Data transformation (where data are transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations)

5. Data mining (an essential process where intelligent methods are applied to extract data patterns)

6. Pattern evaluation (to identify the truly interesting patterns representing knowledge based on interestingness measures)

7. Knowledge presentation (where visualization and knowledge representation techniques are used to present mined knowledge to users)

Knowledge discovery in databases (KDD)-is a multistep process of finding useful information and patterns in data while **Data Mining** is one of the steps in KDD of using algorithms for extraction of patterns.

Data mining applications:

1. Classification: Eg: In loan database, to classify an applicant as a prospective or defaulter, given his various personal and demographic features along with previous purchase characteristics.

2. Estimation: Predict the attribute of a data instance. Eg: estimate the percentage of marks of a student, whose previous marks are already known.

3. Prediction: Predictive model predicts a future outcome rather than the current behaviour. Eg: Predict next week's closing price for the Google share price per unit.

4. Market basket analysis(association rule mining)

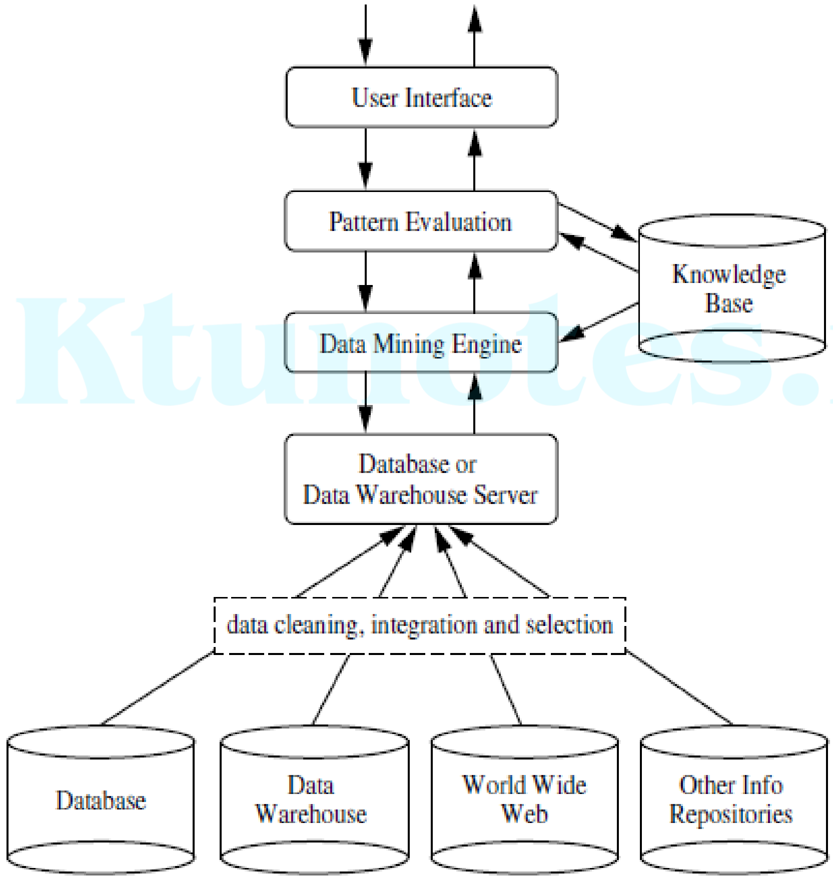
Analyses hidden rules called association rule in a large transactional database.

{pen, pencil-> book} – whenever pen and pencil are purchased together, book is also purchased.

5. Clustering : Classification into different classes based on some similarities but the target classes are unknown.

6. Business intelligence
7. Business data analytics
8. Bioinformatics
9. Web mining
10. Text mining
11. Social network data analysis

Architecture of typical data mining system



Database, data warehouse, World Wide Web, or other information repository: This is one or a set of databases, data warehouses, spreadsheets, or other kinds of information repositories. Data cleaning and data integration techniques may be performed on the data.

Database or data warehouse server: The database or data warehouse server is responsible for fetching the relevant data, based on the user's data mining request.

Knowledge base: This is the domain knowledge that is used to guide the search or evaluate the interestingness of resulting patterns. Such knowledge can include concept hierarchies, used to organize attributes or attribute values into different levels of abstraction.

Data mining engine: This is essential to the data mining system and ideally consists of a set of functional modules for tasks such as characterization, association and correlation analysis, classification, prediction, cluster analysis, outlier analysis, and evolution analysis.

Pattern evaluation module: This component typically employs interestingness measures and interacts with the data mining modules so as to focus the search toward interesting patterns.

User interface: This module communicates between users and the data mining system, allowing the user to interact with the system by specifying a data mining query or task, providing information to help focus the search, and performing exploratory data mining based on the intermediate data mining results.

Data mining Functionalities

1. **Class/Concept Description: Characterization and Discrimination:** Data characterization is a summarization of the general characteristics or features of a target class of data. The data corresponding to the user-specified class are typically collected by a query.

For example, to study the characteristics of software products with sales that increased by 10% in the previous year, the data related to such products can be collected by executing an SQL query on the sales database. Data discrimination is a comparison of the general features of the target class data objects against the general features of objects from one or multiple contrasting classes. For example, a user may want to compare the general features of software products with sales that increased by 10% last year against those with sales that decreased by at least 30% during the same period.

2. Mining Frequent Patterns, Associations, and Correlations: Frequent patterns, as the name suggests, are patterns that occur frequently in data.

A frequently occurring subsequence, such as the pattern that customers, tend to purchase first a laptop, followed by a digital camera, and then a memory card, is a (frequent) sequential pattern..

Association analysis: A confidence, or certainty, of 50% means that if a customer buys a computer, there is a 50% chance that she will buy software as well. A 1% support means that 1% of all the transactions under analysis show that computer and software are purchased together.

3. Classification and Regression for Predictive Analysis:

Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts. The model is derived based on the analysis of a set of training data (i.e., data objects for which the class labels are known).

The model is used to predict the class label of objects for which the class label is unknown.

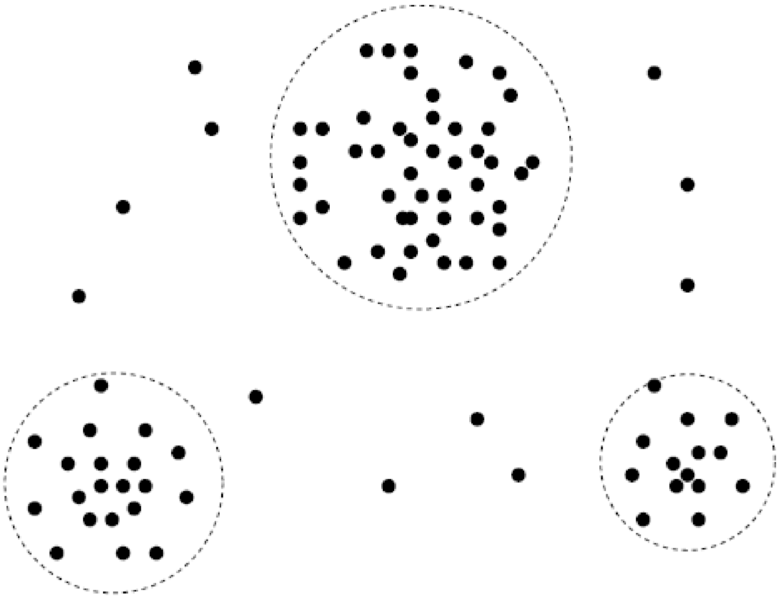
A decision tree is a flowchart-like tree structure, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions.

Regression analysis is a statistical methodology that is most often used for numeric prediction, although other methods exist as well. Regression also encompasses the identification of distribution trends based on the available data.

Unlike classification and regression, which analyze class-labeled (training) data sets,

4. **Clustering** analyzes data objects without consulting class labels. In many cases, class labeled data may simply not exist at the beginning. Clustering can be used to generate class labels for a group of data. The objects are clustered or grouped based on the principle of maximizing the intraclass similarity and minimizing the interclass similarity. That is, clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are

rather dissimilar to objects in other clusters. Each cluster so formed can be viewed as a class of objects, from which rules can be derived. Clustering can also facilitate taxonomy formation, that is, the organization of observations into a hierarchy of classes that group similar events together.



6. Outlier Analysis

A data set may contain objects that do not comply with the general behavior or model of the data. These data objects are outliers. Many data mining methods discard outliers as noise or exceptions. However, in some applications (e.g., fraud detection) the rare events can be more interesting than the more regularly occurring ones. The analysis of outlier data is referred to as outlier analysis or anomaly mining.

Data Mining Issues

Major issues in data mining

1. **Mining methodology and user interaction issues**

- Mining different kinds of knowledge in databases:

Because different users can be interested in different kinds of knowledge, data mining should cover a wide spectrum of data analysis and knowledge discovery task. These tasks may use the same database in different ways and require the development of numerous data mining techniques.

- Interactive mining of knowledge at multiple levels of abstraction:

Interactive mining allows users to focus the search for patterns, providing and refining data mining requests based on returned results. The user can interact with the data mining system to view data and discovered patterns at multiple granularities and from different angles.

- Incorporation of background knowledge:

Background knowledge, or information regarding the domain under study, may be used to guide the discovery process and allow discovered patterns to be expressed in concise terms and at different levels of abstraction

- Pattern evaluation—the interestingness problem:

A data mining system can uncover thousands of patterns. Several challenges remain regarding the development of techniques to assess the interestingness of discovered patterns, particularly with regard to subjective measures that estimate the value of patterns with respect to a given user class, based on user beliefs or expectations. The use of interestingness measures or user- specified constraints to guide the discovery process and reduce the search space is another active area of research.

2. **Performance issues**

- Efficiency and scalability of data mining algorithms:

To effectively extract information from a huge amount of data in databases, data mining algorithms must be efficient and scalable. The running time of a data mining algorithm must be predictable and acceptable in large databases.

- Parallel, distributed, and incremental mining algorithms:

The huge size of many databases, the wide distribution of data, and the computational complexity of some data mining methods are factors motivating the development of parallel and distributed data mining algorithms. Such algorithms divide the data into partitions, which are processed in parallel. The results from the partitions are then merged.

3. **Issues relating to the diversity of database types:**

□ Handling of relational and complex types of data:

Because relational databases and data warehouses are widely used, the development of efficient and effective data mining systems for such data is important. However, other databases may contain complex data objects, hypertext and multimedia data, spatial data, temporal data, or transaction data. Specific data mining systems should be constructed for mining specific kinds of data.

□ Mining information from heterogeneous databases and global information systems:

Local- and wide-area computer networks (such as the Internet) connect many sources of data, forming huge, distributed, and heterogeneous databases. The discovery of knowledge from different sources of structured, semi structured, or unstructured data with diverse data semantics poses great challenges to data mining.

****