CST466
DATA MINING

MODULE-1

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.

TRACE KTU

DATA MINING

- Data mining refers to **extracting or mining knowledge from large amounts of data**.
- Alternatively, we can define data mining as a **technique to find hidden patterns in a huge history database** to help top level managers in **decision making**.
- Also known as;
  - ★ Knowledge extraction.
  - ★ Data/pattern analysis.
  - ★ Data archaeology.
  - ★ Data dredging.
  - ★ Exploratory data analysis etc.,

TRACE KTU

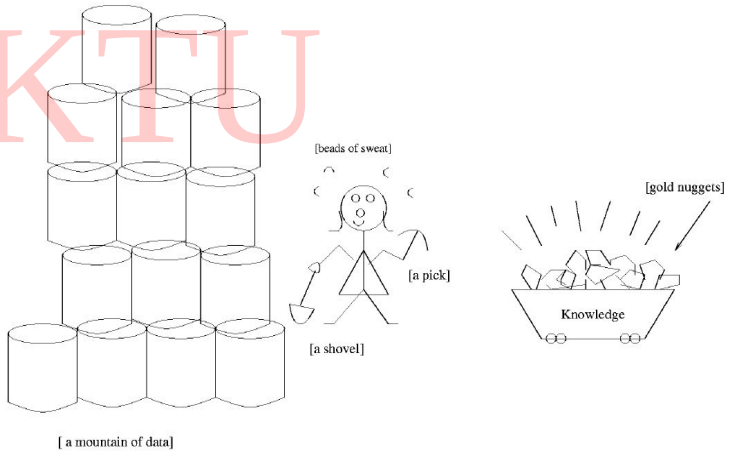


Fig: Data mining - searching for knowledge in data.

KNOWLEDGE DISCOVERY IN DATABASES v/s DATA MINING

- The terms knowledge discovery in databases (KDD) and data mining are often used interchangeably.
  - **KDD** - Knowledge discovery in databases (KDD) is the process of finding useful information and patterns in data.
  - **Data mining** - The use of algorithms to extract the information and patterns derived by the KDD process.

TRACE KTU

- 2 views for the term data mining;
  1. Many people treat **data mining as a synonym for KDD.**

     Knowledge discovery in databases (KDD) is the process of finding useful information and patterns in data.

  2. Some other group of people treats **data mining as one of the many steps in the process of KDD.**

- **Knowledge discovery in databases (KDD)** is the process of finding **useful information and patterns in data.**

- The following figure depicts data mining as a step in KDD.
- It consists of an iterative sequence of the following steps;

1. Data cleaning
2. Data integration
3. Data selection
4. Data transformation
5. Data Mining
6. Pattern Evaluation
7. Knowledge presentation

TRACE KTU

- Steps 1 - 4 : Different forms of data preprocessing. ie, data are prepared for mining.
- Step 5 : Data Mining.
    - This step may interact with the user or a knowledge base.
    - The interesting patterns are presented to the user and may be stored as a new knowledge in the knowledge base.

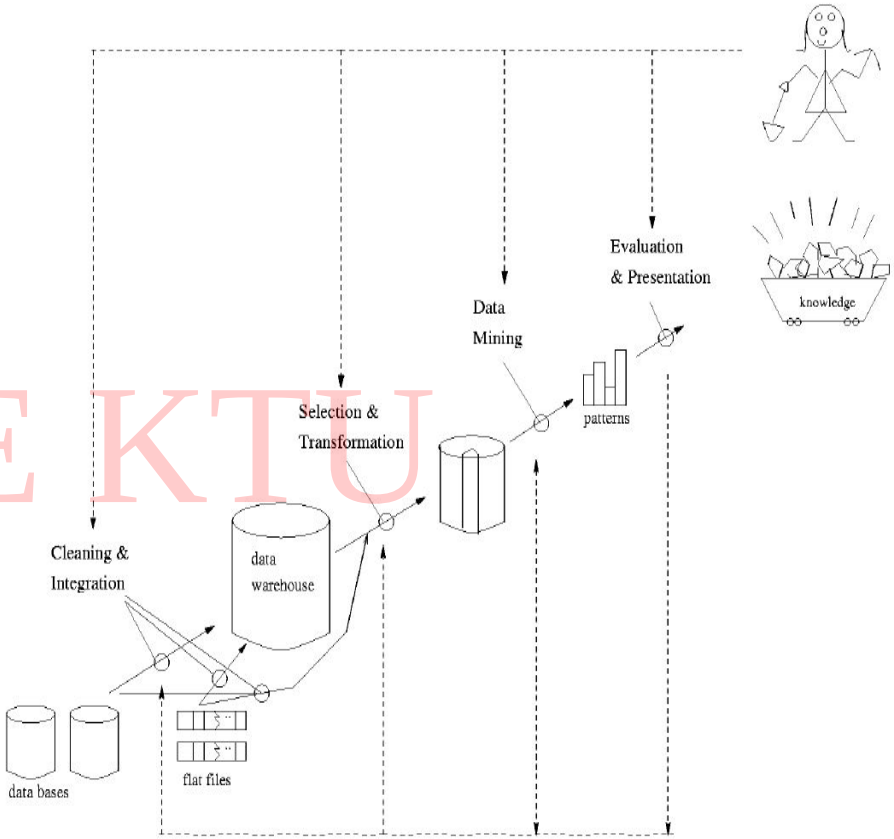


Fig: Data mining as a step in KDD

1. Data cleaning:

- To remove noise and inconsistent data.

2. Data integration:

- Multiple data sources may be combined.

3. Data selection:

- Data relevant to the analysis task are retrieved from the database.

4. Data transformation

- Data are transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations.

TRACE KTU

5. Data mining:

- An essential process where intelligent methods are applied to extract data patterns.

6. Pattern evaluation:

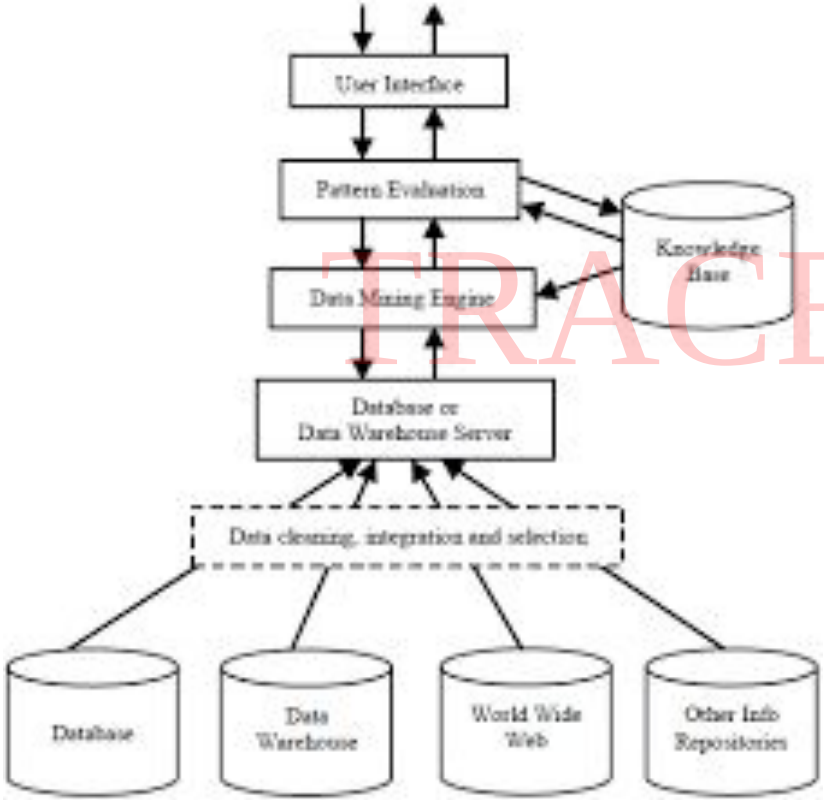
- To identify the truly interesting patterns representing knowledge based on interestingness measures.

7. Knowledge presentation:

- Visualization and knowledge representation techniques are used to present mined knowledge to users.

ARCHITECTURE OF TYPICAL DATA MINING SYSTEM:

- The following figure represents the architecture of a typical data mining system.



MAJOR COMPONENTS:

- Database, data warehouse, World wide web, or other information repository.
- Database or other data warehouse server.
- Knowledge base.
- Data mining engine.
- Pattern evaluation module.
- User interface.

TRACE KTU

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.

TRACE KTU

Data mining engine:

- This is an essential component of a data mining system.
- Consists of a set of functional modules for tasks such as;
    - Characterization.
    - Association and correlation analysis.
    - Classification.
    - Prediction.
    - Cluster analysis.
    - Outlier analysis.
    - 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.
- It may use interestingness thresholds to filter out discovered patterns.
- Alternatively, the pattern evaluation module may be integrated with the mining module, depending on the implementation of the data mining method used.
- For efficient data mining, it is highly recommended to push the evaluation of pattern interestingness as deep as possible into the mining process so as to confine the search to only the interesting patterns.

TRACE KTU

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.
- In addition, this component allows the user to browse database and data warehouse schemas or data structures, evaluate mined patterns, and visualize the patterns in different forms.

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.
- Knowledge such as user beliefs may also be included.
- Other examples of domain knowledge are additional interestingness constraints or thresholds, and metadata.

TRACE KTU

DATA MINING APPLICATIONS:

1. **Market basket Analysis (association rule analysis):**
   - Analyses hidden rules called association rule in a large transactional database.
   - Eg: Consider the rule {pen, pencil -> book}
   - The above rule provides the information that whenever pen and pencil are purchased together, book is also purchased.
   - So, these items can be placed together for sales or supplied as a complementary product with one another to increase the overall sales of each item.

2. **Classification:**

TRACE KTU

   - The goal is to classify a new data record into one of the many possible classes, which are already known.
   - Eg: In a loan database, to classify an applicant as a prospective applicant or defaulter, given his various personal and demographic features along with previous purchase characteristics.

3. **Estimation:**

   - To predict the attribute of a data instance - usually a numeric value rather than categorical class.
   - Eg: To estimate the percentage of marks of a student, whose previous marks are already known.

4. Prediction:

- Predicts a future outcome rather than the current behaviour.
- The output attribute can be categorical or numeric.
- Eg: Predict next week's closing price for the Google share price per unit.

(Note: The border line between prediction, classification and estimation is too narrow)

5. Clustering:

- Unsupervised learning is used, where target classes are unknown.
- Eg: Given 1000 applicants have to be classified based on certain similarity criteria and it is not predefined which are those classes to which the applicants should finally be grouped into.

TRACE KTU

6. Web mining:

- It is a data mining technique to automatically discover and extract information from Web documents and services.
- The main purpose of web mining is discovering useful information from the World-Wide Web and its usage patterns.

7. Text mining:

- Text mining is a component of data mining that deals specifically with unstructured text data.
- It involves the use of natural language processing (NLP) techniques to extract useful information and insights from large amounts of unstructured text data.

8. Social network data analysis:

- The process of investigating social structures through the use of networks and graph theory.
- The data from social networks is aggregated and analyzed to identify patterns and trends.

TRACE KTU

9. Business data analytics

10. Bioinformatics:

- Bioinformatics is a scientific subdiscipline that involves using computer technology to collect, store, analyze and disseminate biological data and information, such as DNA and amino acid sequences or annotations about those sequences.
- Mining biological data helps to extract useful knowledge from massive datasets gathered in biology, and in other related life sciences areas such as medicine and neuroscience.

DATA MINING ISSUES:

1. Mining methodology and user interaction issues:

- *Mining different kinds of knowledge in databases:*
  - Different users may be interested in different kinds of knowledge.So, 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.
  - Domain knowledge related to databases can help focus and speed up a data mining process, or judge the interestingness of discovered patterns.

TRACE KTU

- *Handling noisy or incomplete data:*
  - The data stored in a database may reflect noise, exceptional cases, or incomplete data objects.
  - When mining data regularities, these objects may confuse the process, causing the knowledge model constructed to overfit the data.
  - As a result, the accuracy of the discovered patterns can be poor.
- *Presentation and visualization of data mining results:*
  - Discovered knowledge should be expressed in high-level languages, visual representations, or other expressive forms so that the knowledge can be easily understood and directly usable by humans.
  - This requires the system to adopt expressive knowledge representation techniques, such as trees, tables, rules, graphs, charts, crosstabs, matrices, or curves.
- *Data mining query languages and ad hoc data mining:*
  - Relational query languages (such as SQL) allow users to pose adhoc queries for data retrieval).
  - Similarly, high-level data mining query languages need to be developed to allow users to describe adhoc data mining tasks.
  - Such a language should be integrated with a database or data warehouse query language and optimized for efficient and flexible data mining

TRACE KTU

2. Performance issues:

- These include efficiency, scalability, and parallelization of data mining algorithms.
- *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.
  - In other words, 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.
  - Moreover, the high cost of some data mining processes promotes the need for incremental data mining algorithms that incorporate database updates without having to mine the entire data again "from scratch."
  - Such algorithms perform knowledge modification incrementally to amend and strengthen what was previously discovered.

TRACE KTU

3. Issues relating to the diversity of database types:

- *Handling of relational and complex types of data:*
  - Relational databases and data warehouses are widely used.So, 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.
  - It is unrealistic to expect one system to mine all kinds of data, given the diversity of data types and different goals of data mining.
  - 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, semistructured, or unstructured data with diverse data semantics poses great challenges to data mining.

❖ The above issues are considered major requirements and challenges for the further evolution of data mining technology.

❖ Some of the challenges have been addressed in recent data mining research and development, to a certain extent while others are still at the research stage.

TRACE KTU

DATA MINING FUNCTIONALITIES:

- Data mining functionalities are **used to specify the kind of patterns to be found in data mining tasks.**
- Data mining tasks can be classified into two categories:
  - Descriptive mining tasks.
    - Descriptive mining tasks characterize the general properties of the data in the database.
  - Predictive mining tasks.
    - Predictive mining tasks perform inference on the current data in order to make predictions.
- Following are some of the data mining functionalities , and the kinds of patterns they can discover;

TRACE KTU

1. Concept/Class Description: Characterization and Discrimination.

2. Mining Frequent Patterns, Associations, and Correlations.

3. Classification and Prediction.

4. Cluster Analysis.

5. Outlier Analysis.

6. Evolution Analysis.

Concept/Class Description: Characterization and Discrimination:

- Data entries can be associated with classes or concepts.
- **Eg:**
  - Consider an Electronics store.
  - Classes of items for sale include computers and printers

    &

  - Concepts of customers include bigSpenders and budgetSpenders.
  - It can be useful to describe individual classes and concepts in summarized, concise, and yet precise terms.
  - Such **descriptions of a class or a concept are called class/concept descriptions.**
  - These descriptions can be derived using;
    - **Data characterization**
    - **Data discrimination**
    - **Both data characterization and discrimination**

TRACE KTU

Data characterization:

- 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.
- The output of data characterization can be presented in various forms such as;
  - Pie charts
  - Bar chart
  - Curves
  - Multidimensional data cubes
  - Multidimensional tables
- Eg:
  - 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.

TRACE KTU

Data discrimination:

- 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.**
  - The target and contrasting classes can be specified by a user.
  - The corresponding data objects can be retrieved through database queries.
- The output of data discrimination are similar to those of characteristic descriptions, although discrimination descriptions should include comparative measures that help to distinguish between the target and contrasting classes.
- Discrimination descriptions expressed in the form of rules are called discriminant rules.
- Eg:
  - 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.

TRACE KTU

Mining Frequent Patterns, Associations, and Correlations:

- Frequent patterns are the patterns that occur frequently in data.
- There are many kinds of frequent patterns such as;
  - **Frequent itemsets**
  - **Frequent subsequences**
  - **Frequent substructures**
- A frequent itemset typically refers to a set of items that frequently appear together in a transactional data set.
  - **Eg:** Milk and Bread are frequently purchased together in grocery stores by many customers.
- A frequently occurring subsequence, such as the pattern that customers tend to purchase first a PC, followed by a digital camera, and then a memory card, is a (frequent) sequential pattern.
- A substructure can refer to different structural forms, such as graphs, trees, or lattices, which may be combined with itemsets or subsequences.
- If a substructure occurs frequently, it is called a (frequent) structured pattern.
- Mining frequent patterns leads to the discovery of interesting associations and correlations within data.

TRACE KTU

- **Association analysis:**

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

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

Classification and Prediction:

TRACE KTU

- Classification is the process of **finding a model** (or function) **that describes and distinguishes data classes**.
- This model can be used **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 (i.e., data objects whose class label is known).
- The derived model may be represented in various forms, such as;
  - **IF-THEN rules**
  - **Decision trees**
  - **Neural networks**

$age(X, \text{“youth”}) \ AND \ income(X, \text{“high”}) \longrightarrow class(X, \text{“A”})$

$age(X, \text{“youth”}) \ AND \ income(X, \text{“low”}) \longrightarrow class(X, \text{“B”})$

$age(X, \text{“middle\_aged”}) \longrightarrow class(X, \text{“C”})$

Fig:A classification model in the form of **IF-THEN** rules.

- A **decision tree** is a flow-chart-like tree structure.
  - Each node denotes a test on an attribute value.
  - Each branch represents an outcome of the test.
  - Each tree leaves represent classes.

TRACE KTU

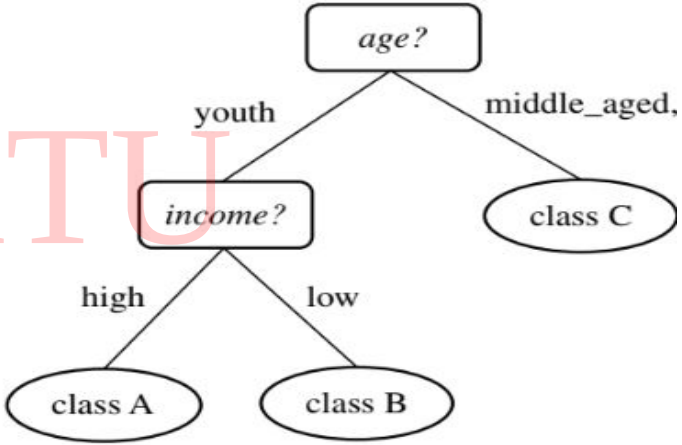


Fig:A classification model in the form of Decision tree.

- A **neural network**, when used for classification, is typically a collection of neuron-like processing units with weighted connections between the units.

Prediction:

- Regression **predicts**/models **continuous-valued functions.**
  - ie, regression is used to predict missing or unavailable numerical data values rather than (discrete) class labels.
- The term **prediction refers to both numeric prediction and class label prediction.**

Note:

- Classification and regression may need to be preceded by **relevance analysis,** which attempts to identify attributes that are significantly relevant to the classification and regression process.
- Such attributes will be selected for the classification and regression process.
- Other attributes, which are irrelevant, can then be excluded from consideration.

TRACE KTU

Cluster Analysis:

- Clustering **groups similar data objects together** (into classes) without consulting a known class label.
  - ie, **class labels are unknown.**
- The class labels are not present in the training data simply because they are not known to begin with.

- Clustering can be used to generate such labels.
- The objects are clustered or grouped based on the principle of **maximizing the intraclass similarity and minimizing the inter-class similarity.**
    - ie, clusters of objects are formed so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters.

Outlier Analysis

- A database may contain data objects that do not comply with the general behavior or model of the data.
- These data objects are **outliers**.
- Most data mining methods discard outliers as noise or exceptions.
- However, in some applications such as 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 mining.
- **Eg:**
    - Outlier analysis may uncover fraudulent usage of credit cards by detecting purchases of extremely large amounts for a given account number in comparison to regular charges incurred by the same account.
    - Outlier values may also be detected with respect to the location and type of purchase, or the purchase frequency.

TRACE KTU

Evolution Analysis:

- Evolution analysis **describes and models regularities or trends for objects whose behavior changes over time.**

OR

- Evolution Analysis pertains to the **study of data sets that change over time.**
- Distinct features of such an analysis includes;
  - Time-series data analysis.
  - Sequence or periodicity pattern matching.
  - Similarity-based data analysis.
- **Eg:**
  - Suppose that you have the major stock market (time-series) data of the last several years.
  - You would like to invest in shares of high-tech industrial companies.
  - A data mining study of stock exchange data may identify stock evolution regularities for overall stocks and for the stocks of particular companies.
  - Such regularities may help predict future trends in stock market prices, contributing to your decision making regarding stock investments.

TRACE KTU

DATA WAREHOUSE:

- A data warehouse refers to a data repository that is maintained separately from an organization's operational databases.
- It can also be described as a centralized data repository which can be queried for **business benefits.**
- **It stores the information an enterprise needs to make strategic decisions.**
- It stores information oriented to satisfy **decision-making** requests.
- Such systems allow for the integration of a variety of application systems.
- They support information processing by providing a solid platform of consolidated **historic data** for analysis.
- It is a group of decision support technologies, targets to enable the knowledge worker (executive, manager, and analyst) to make superior and higher decisions.
- So, Data warehousing support architectures and tool for business executives to systematically organize, understand and use their information to make strategic decisions.

TRACE KTU

- Systematically defining;

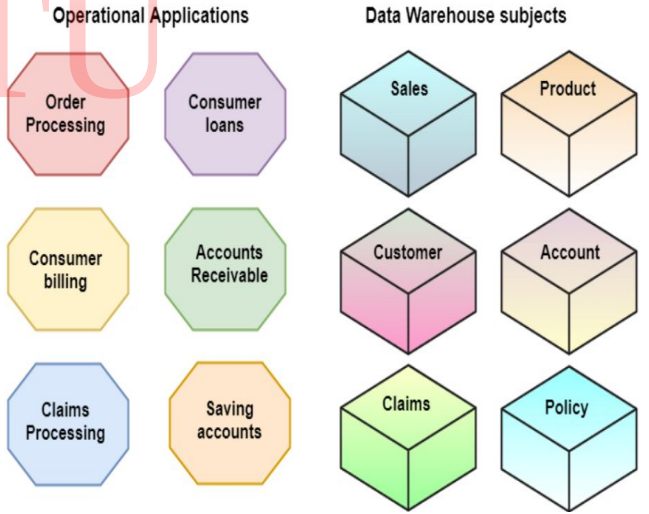
  "A data warehouse is a **subject-oriented, integrated, time-variant**, and **non-volatile** collection of data in support of management's decision making process"

- 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.

Subject-oriented:

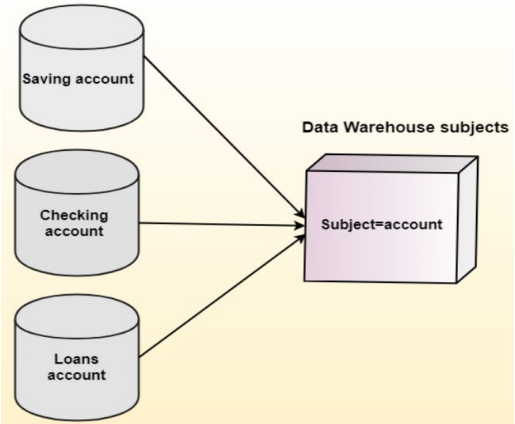
TRACE KTU

- A data warehouse target on the modeling and analysis of data for decision-makers.
- Therefore, data warehouses typically provide a concise and straightforward view around a particular subject, such as customer, product, or sales, instead of the global organization's ongoing operations.
- This is done by excluding data that are not useful concerning the subject and including all data needed by the users to understand the subject.



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.



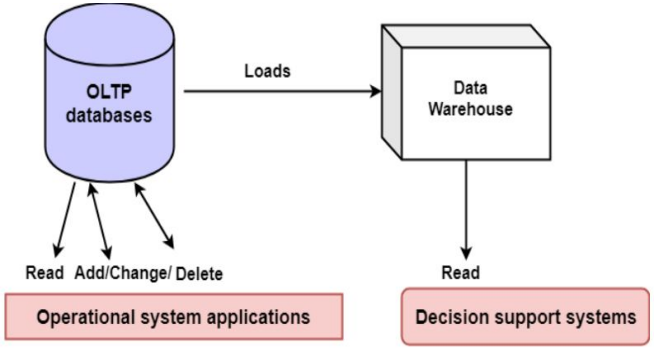
Time-variant:

- Data are stored to provide information from an historic perspective.
- Eg: One can retrieve files from 3 months, 6 months, 12 months, or even previous data from a data warehouse.
- Every key structure in the data warehouse contains, either implicitly or explicitly, a time element.

TRACE KTU

Non-volatile:

- The data warehouse is a physically separate data storage, which is transformed from the source operational RDBMS.
- The operational updates of data do not occur in the data warehouse.
  - i.e., update, insert, and delete operations are not performed.
- It usually requires only two procedures in data accessing:
  - Initial loading of data and access to data.
- Non-volatile defines that once entered into the warehouse, and data should not change.



OPERATIONAL DATABASE SYSTEMS v/s 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.
- **Data warehouse systems** serves 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.

TRACE KTU

- The major distinguishing features of OLTP and OLAP are summarized as follows:
  - **Users and system orientation:**
    - **OLTP system:**
      - Customer-oriented.
      - Used for transaction and query processing by clerks, clients, and IT professionals.
    - **OLAP system:**
      - Market-oriented.
      - Used for data analysis by knowledge workers, including managers, executives, and analysts.
  - **Data contents:**
    - **OLTP system:**
      - Manages current data.
      - This data is typically too detailed to be easily used for decision making.
    - **OLAP system:**
      - Manages large amounts of historic data.
      - This historic data provides facilities for summarization and aggregation, and stores and manages information at different levels of granularity.
      - These features make the data easier to use for informed decision making.

TRACE KTU

- **View:**
  - **OLTP system:**
    - Focuses mainly on the current data within an enterprise or department, without referring to historic data or data in different organizations.
  - **OLAP system:**
    - Spans multiple versions of a database schema, due to the evolutionary process of an organization.
    - Deals with information that originates from different organizations, integrating information from many data stores.
    - Because of their huge volume, OLAP data are stored on multiple storage media.
- **Access patterns:**
  - **OLTP system:**
    - Consists mainly of short, atomic transactions.
    - Such a system requires concurrency control and recovery mechanisms.
  - **OLAP system:**
    - Accesses to OLAP systems are mostly read-only operations (because, most data warehouses store historic rather than up-to-date information)

TRACE KTU

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	historic, 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	GB to high-order GB	$\geq$ TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

TRACE KTU

DATA WAREHOUSE ARCHITECTURE:

- Data warehouses adopt a **three-tier architecture** as shown in figure below.

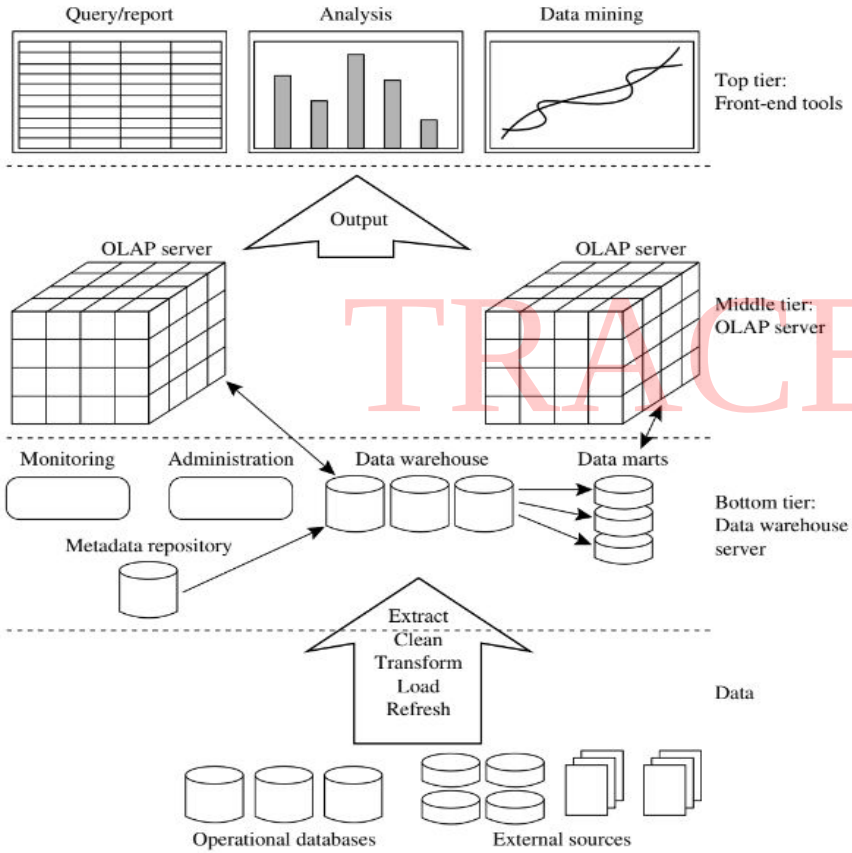


Fig: A three-tier data warehousing architecture.

TRACE KTU

Top-tier:

- The top tier is a **front-end client layer**, which contains;
  - Query and reporting tools
  - Analysis tools
  - Data mining tools

Middle-tier:

TRACE KTU

- The middle tier is an **OLAP server** that is typically implemented using;
  - A relational OLAP (ROLAP) model (i.e., an extended relational DBMS that maps operations on multidimensional data to standard relational operations)

OR

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

Bottom-tier:

- The bottom tier is a **data warehouse 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 (e.g., customer profile information provided by external consultants).
    - These tools and utilities perform data extraction, cleaning, and transformation

      and

    - 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 (Object Linking and Embedding Database) by Microsoft and JDBC (Java Database Connection).
- This tier also contains a metadata repository, which stores information about the data warehouse and its contents.
    - Metadata are data bout the data.
    - Metadata are created for the data names and definitions of the given warehouse.

TRACE KTU

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.

TRACE KTU

Metadata Repository:

- Metadata are data about data.
- When used in a data warehouse, metadata are the data that define warehouse objects.
- The bottom tier of the data warehousing architecture contains a metadata repository.
- Metadata are created for the data names and definitions of the given warehouse.
- Additional metadata are created and captured for timestamping any extracted data, the source of the extracted data, and missing fields that have been added by data cleaning or integration processes.

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

- From the architecture point of view, there are three data warehouse models;
  - The enterprise warehouse
  - The data mart
  - 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.

TRACE KTU

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.

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.

TRACE KTU

MULTIDIMENSIONAL DATA MODEL : DATA CUBE

- A multidimensional model views data in the form of a data-cube.
- A data cube allows data to be modeled and viewed in multiple dimensions.
- In data warehousing, the data cube is n-dimensional.
    - It is defined by dimensions and facts.

Dimensions:

- 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.
- Eg:
    - Consider an electronics shop.
    - The shop 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.**
        - These dimensions allow the store to keep track of things like monthly sales of items and the branches and locations at which the items were sold.
    - Each dimension may have a table associated with it, called a **dimension table**, which further describes the dimension.
        - For example, a dimension table for item may contain the attributes item name, brand, and type.
        - Dimension tables can be specified by users or experts, or automatically generated and adjusted based on data distributions.

TRACE KTU

Facts:

- Facts are numeric measures.
- Facts are the quantities by which we want to analyze relationships between dimensions.
- A multidimensional data model is typically organized around a central theme.
    - This theme is represented by a fact table.
- The fact table contains the names of the facts, or measures, as well as keys to each of the related dimension tables.
- **Eg:**
    - Facts for a sales data warehouse include;
        - dollars sold (sales amount in dollars)
        - units sold (number of units sold)
        - amount budgeted.
    - The fact table contains the names of the facts, or measures, as well as keys to each of the related dimension tables.

TRACE KTU

Example:

- Consider the Electronics sales data for items sold per quarter in the city of Vancouver.
- These data are shown in the table below.

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

TRACE KTU

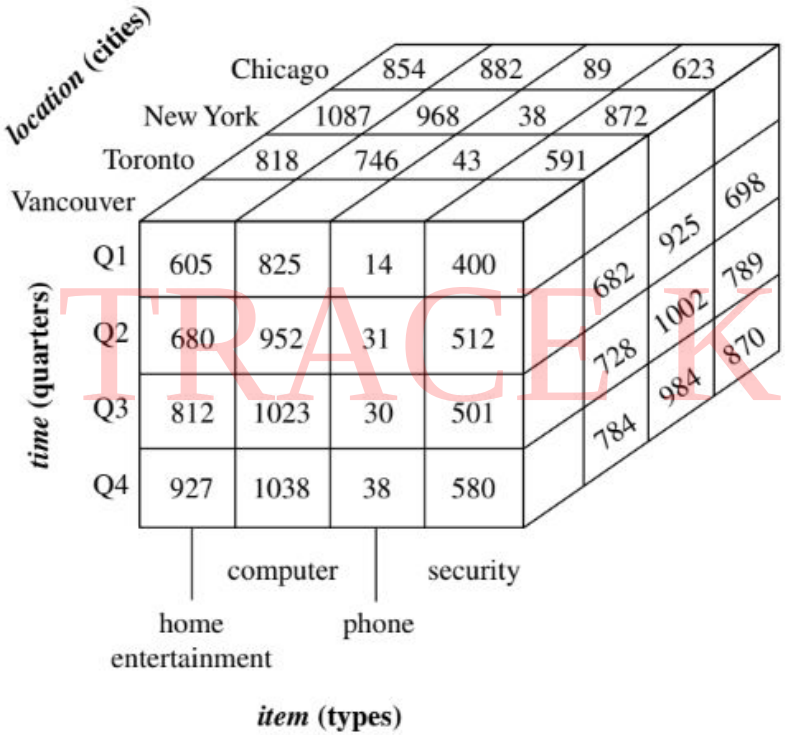
- In this 2-D representation, the sales for Vancouver are shown with respect to the;
  - **Time dimension** (organized in quarters)
  - **Item dimension** (organized according to the types of items sold).
- The fact or measure displayed is **dollars sold** (in thousands).

- Suppose that we would like to view the sales data with a third dimension.
- For instance, suppose we would like to view the data according to time, item and location, for the cities Chicago, New York, Toronto, and Vancouver.
- These 3-D data are shown in table below.
- The 3-D data in the table are represented as a series of 2-D tables.

	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

TRACE KTU

- Conceptually, we may also represent the same data in the form of a 3-D data cube as below;



TRACE KTU

WAREHOUSE SCHEMA:

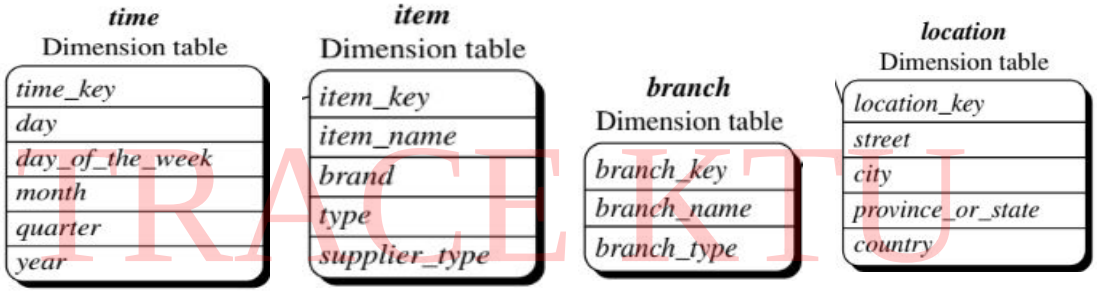
- Schema is a logical description of the entire database.
  - It includes the name and description of records of all record types.
- A data warehouse requires a concise, subject-oriented schema that facilitates online data analysis.
- The most popular data model for a data warehouse is a **multidimensional data model**.
- A data warehouse uses the following schema's;
  - **Star schema**
  - **Snowflake schema**
  - **Fact constellation schema**

TRACE KTU

Note:

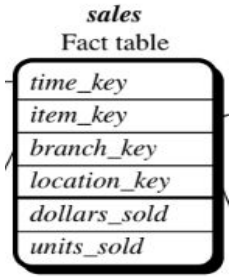
- *In the design of relational databases, an entity-relationship (ER) data model is commonly used.*
- *In ER-model, database schema consists of a set of entities and the relationships between them.*
- *Such a data model is appropriate for online transaction processing (OLTP).*

- **Eg:** Consider an electronics shop.
- The shop may create a **sales data warehouse** in order to keep records of the store's sales with respect to the **dimensions :**
  - **Time**
  - **Item**
  - **Branch**
  - **Location**
- Each dimension may have a table associated with it, called a **dimension table**, which further describes the dimension.



TRACE KTU

- **Facts** are numeric measures by which we can analyze relationships between dimensions.
- Each fact may have a table associate with it, called a **fact table**, which further describes the facts.



Star Schema:

- This the most popular schema used for data warehouse design.
- In star schema, the data warehouse contains;
  - A large **central table (fact table)** containing the bulk of the data, with no redundancy.
  - 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.**

Drawback:

TRACE KTU

- *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}.*
    - *For example, "Trivandrum" and "Kochi" are both cities in the state of Kerala,India.*
    - *Entries for such cities in the location dimension table will create redundancy among the attributes province or state and country;*
    - *ie, , (..., Trivandrum, Kerala,India) and (..., Kochi, Kerala, India).*
    - *This constraint may introduce some redundancy.*

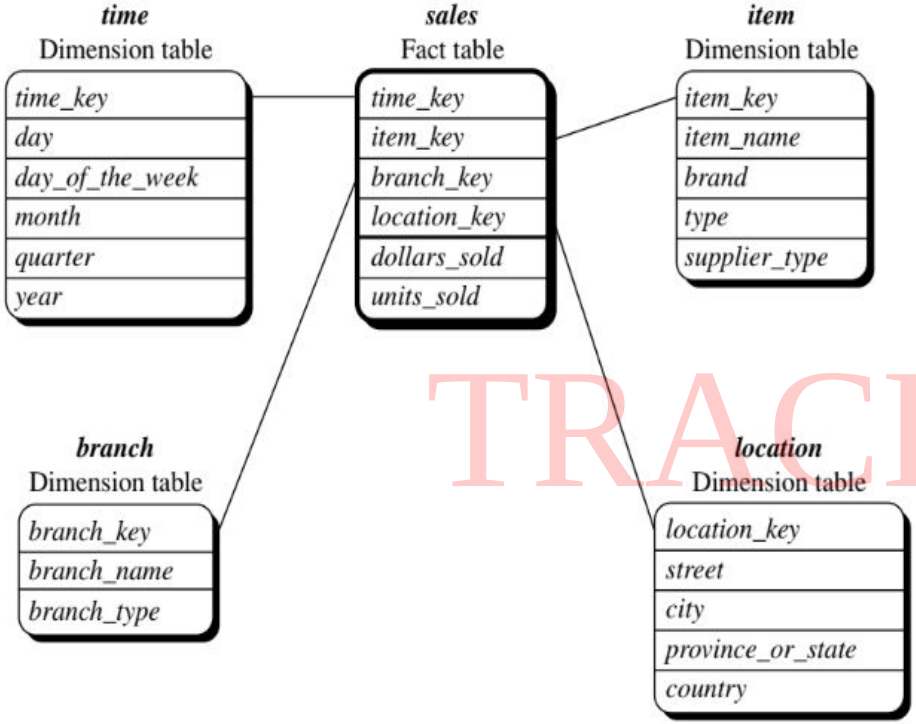


Fig: Star schema of sales data warehouse.

- A star schema for the sales datawarehouse is shown in the 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.

TRACE KTU

Snowflake Schema:

- The snowflake schema is **a variant of the star schema model.**
- In snowflake schema, some **dimension tables are normalized**.
    - ie, data is splitted into additional tables.
- The resulting schema graph forms a shape similar to a snowflake.
- 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.

Advantage:

TRACE KTU

- *Since the dimension tables of snowflake schema is normalized, such a **table is easy to maintain and saves storage space.***
    - *However, this space savings is negligible in comparison to the typical magnitude of the fact table.*

Drawbacks:

- *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**.*

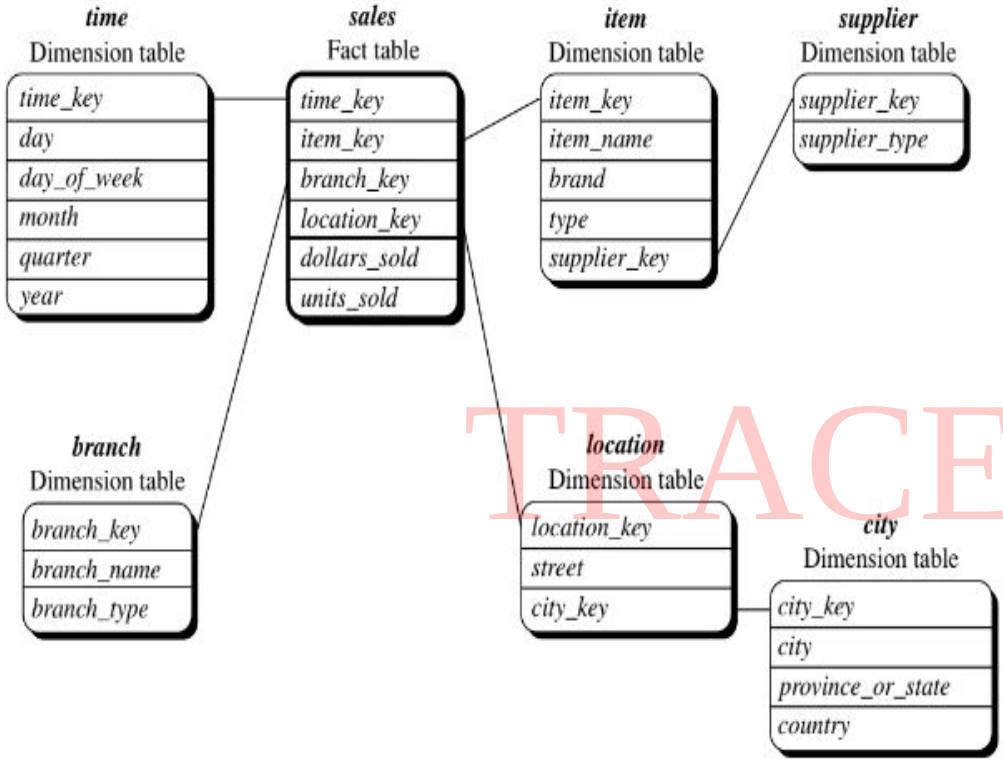


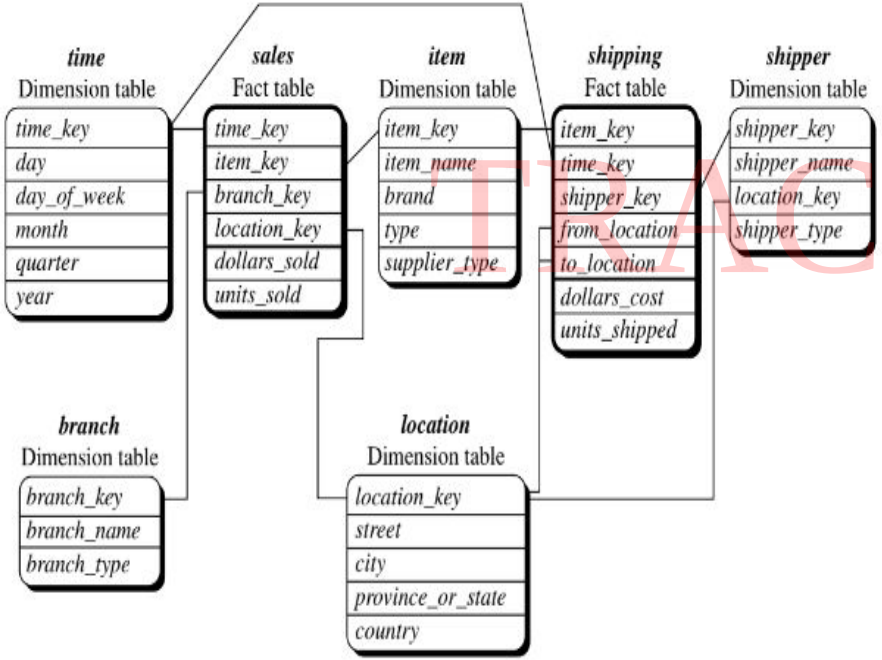
Fig: Snowflake schema of a sales data warehouse.

TRACE KTU

- The *sales fact table* is identical to that of the star schema.
- 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 into two new tables: *item* and *supplier.*
  - 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.*
- Similarly, the single dimension table for location in the star schema can be normalized into two new tables: *location* and *city.*
  - The *city_key* in the new *location* table links to the *city* dimension table.

Fact Constellation Schema:

- 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 is shown in the figure.
- This schema specifies **two fact tables**:
  - *sales* and *shipping*.
- The *sales* table definition is identical to that of the star schema.
- The *shipping* table has five dimensions: item_key,time_key,shipper_key,from_location and to_location — 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, item,* and *location* are shared between the *sales* and *shipping* fact tables.

TRACE KTU

Note:

Data warehouse v/s Data mart:

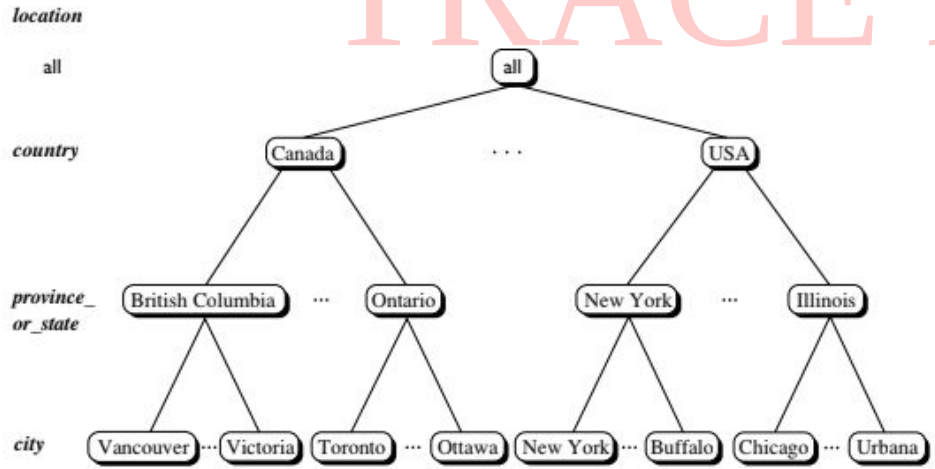
- In data warehousing, there is a distinction between a data warehouse and a data mart.
- A data warehouse collects information about subjects that span the entire organization, such as customers, items, sales, assets, and personnel, and thus its scope is enterprise-wide.
    - For data warehouses, the fact constellation schema is commonly used, since it can model multiple, interrelated subjects.
- A data mart is a department subset of the data warehouse.
- It 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.

TRACE KTU

CONCEPT HIERARCHY:

- A concept hierarchy defines a sequence of **mappings from a set of low-level concepts to higher-level,** more general **concepts.**
  - Consider a concept hierarchy for the dimension *location*.
  - City values for *location* includes: Vancouver, Toronto, New York, and Chicago.
  - Each city 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 United States) to which they belong.
  - These mappings form a concept hierarchy for the dimension *location,* mapping a set of low-level concepts (i.e., cities) to higher-level, more general concepts (i.e., countries).
  - This concept hierarchy is illustrated in the figure.

TRACE KTU



street < city < province or state < country

OLAP OPERATIONS:

- In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies.
- This 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.
- Following are the common OLAP operations for multidimensional data;
    - **Roll-up**
    - **Drill-down**
    - **Slice**
    - **Dice**
    - **Pivot**

TRACE KTU

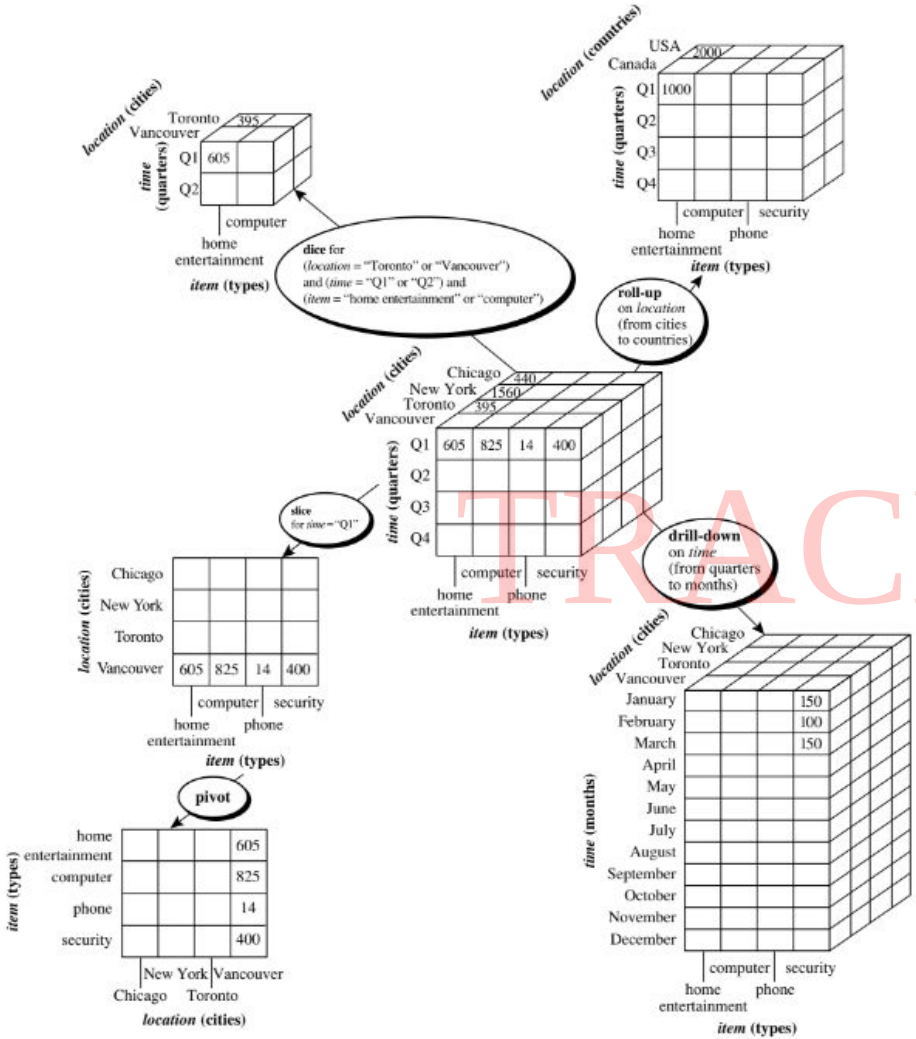


Fig: Examples of OLAP operations on multidimensional data.

- At the center of the figure is a data cube for *sales* data warehouse.
- The cube contains the dimensions;
  - *location, time,* and item
  - *location* is aggregated with respect to *city* values.
  - *time* is aggregated with respect to *quarters.*
  - *item* is aggregated with respect to *item types.*
- The measure displayed is *dollars sold*.
- The data examined are for the cities *Chicago, New York, Toronto,* and *Vancouver.*

TRACE KTU

Roll-up: (Drill-up)

- Performs **aggregation** on a data cube, either by **climbing up a concept hierarchy for a dimension** or by **dimension reduction.**
- It **navigates from more detailed data to less detailed data**.
- Figure shows the result of a roll-up operation performed on the central cube by climbing up the concept hierarchy for the dimension *location*.
- **The roll-up operation shown aggregates the data by ascending the *location* hierarchy from the level of *city* to the level of *country.***
    - In other words, rather than grouping the data by city, the resulting cube groups the data by country.
- When roll-up is performed by dimension reduction, one or more dimensions are removed from the given cube.
    - For example, consider a *sales* data cube containing only the *location* and *time* dimensions.
    - Roll-up may be performed by removing the *time* dimension, resulting in an aggregation of the total sales by *location*, rather than by *location* and by *time.*

TRACE KTU

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.**
- Figure shows the result of a drill-down operation performed on the central cube by stepping down a concept hierarchy for time defined as “day < month < quarter < year.”
- 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.
- Drill-down can also be performed by adding new dimensions to a cube.
    - For example, a drill-down on the central cube can occur by introducing an additional dimension, such as customer group.

TRACE KTU

Slice:

- The slice operation performs a **selection on one dimension of the given cube, resulting in a subcube.**
- Figure shows a slice operation where the sales data are selected from the central cube for the dimension *time* using the criterion *time* = *"Q1."*

Dice:

- The dice operation **defines a subcube by performing a selection on two or more dimensions.**
- Figure shows a dice operation on the central cube based on the following selection criteria that involve three dimensions:

(location = "Toronto" or "Vancouver") and (time = "Q1" or "Q2") and (item = "home entertainment" or "computer").

TRACE KTU

Pivot:(Rotate)

- Pivot is a **visualization operation** that **rotates the data axes** in view to **provide an alternative data presentation.**
- Figure shows a pivot operation where the item and location axes in a 2-D slice are rotated.
- Other examples include rotating the axes in a 3-D cube, or transforming a 3-D cube into a series of 2-D planes.

TYPES OF OLAP SERVERS : ROLAP v/s MOLAP v/s HOLAP

- OLAP servers provides multidimensional data from data warehouses or data marts to business users, without concerns regarding how or where the data are stored.
- Implementations of a warehouse server for OLAP processing include the following:
  - **Relational OLAP (ROLAP) servers**
  - **Multidimensional OLAP (MOLAP) servers**
  - **Hybrid OLAP (HOLAP) servers**

Relational OLAP (ROLAP) servers:

TRACE KTU

- 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.
- **Eg:** The **DSS server** of Microstrategy adopts the ROLAP approach.

Multidimensional OLAP (MOLAP) servers:

- These servers **support multidimensional data views** 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.
- Many MOLAP servers adopt a two-level storage representation to handle dense and sparse data sets: Denser subcubes are identified and stored as array structures, whereas sparse subcubes employ compression technology for efficient storage utilization.

TRACE KTU

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 detailed data to be stored in a relational database, while aggregations are kept in a separate MOLAP store.
- **Eg:** The **Microsoft SQL Server 2000** supports a hybrid OLAP server.