Data Mining
UNIT-I

Introduction to Data Mining: Data mining, Knowledge Discovery process, Data Mining Functionalities-Kinds of Patterns, Major Issues in Data Mining. Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Dissimilarity of Numeric Data-Distance measures, Data Pre-processing: Major Tasks in Data Pre-processing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Data Mining

Data mining is the process of extracting knowledge or insights from large amounts of data using various statistical and computational techniques. The data can be structured, semi-structured or unstructured, and can be stored in various forms such as databases, data warehouses, and data lakes.

The primary goal of data mining is to discover hidden patterns and relationships in the data that can be used to make informed decisions or predictions. This involves exploring the data using various techniques such as clustering, classification, regression analysis, association rule mining, and anomaly detection.

Data mining has a wide range of applications across various industries, including marketing, finance, healthcare, and telecommunications. For example, in marketing, data mining can be used to identify customer segments and target marketing campaigns, while in healthcare, it can be used to identify risk factors for diseases and develop personalized treatment plans.

However, data mining also raises ethical and privacy concerns, particularly when it involves personal or sensitive data. It's important to ensure that data mining is conducted ethically and with appropriate safeguards in place to protect the privacy of individuals and prevent misuse of their data.

KDD Process in Data Mining

In the context of computer science, "Data Mining" can be referred to as knowledge mining from data, knowledge extraction, data/pattern analysis, data archaeology, and data dredging. Data Mining also known

as Knowledge Discovery in Databases, refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data stored in databases.

The need of data mining is to extract useful information from large datasets and use it to make predictions or better decision-making. Nowadays, data mining is used in almost all places where a large amount of data is stored and processed.

For examples: Banking sector, Market Basket Analysis, Network Intrusion Detection.

KDD Process

KDD (Knowledge Discovery in Databases) is a process that involves the extraction of useful, previously unknown, and potentially valuable information from large datasets. The KDD process is an iterative process and it requires multiple iterations of the above steps to extract accurate knowledge from the data.The following steps are included in KDD process:

Data Cleaning

Data cleaning is defined as removal of noisy and irrelevant data from collection.

1. Cleaning in case of **Missing values**.
2. Cleaning **noisy** data, where noise is a random or variance error.
3. Cleaning with **Data discrepancy detection** and **Data transformation tools**.

Data Integration

Data integration is defined as heterogeneous data from multiple sources combined in a common source(DataWarehouse). Data integration using **Data Migration tools, Data Synchronization tools and ETL** (Extract-Load-Transformation) process.

Data Selection

Data selection is defined as the process where data relevant to the analysis is decided and retrieved from the data collection. For this we can use **Neural network, Decision Trees, Naive bayes, Clustering**, and **Regression** methods.

Data Transformation

Data Transformation is defined as the process of transforming data into appropriate form required by mining procedure. Data Transformation is a two step process:

1. **Data Mapping**: Assigning elements from source base to destination to capture transformations.
2. **Code generation**: Creation of the actual transformation program.

Data Mining

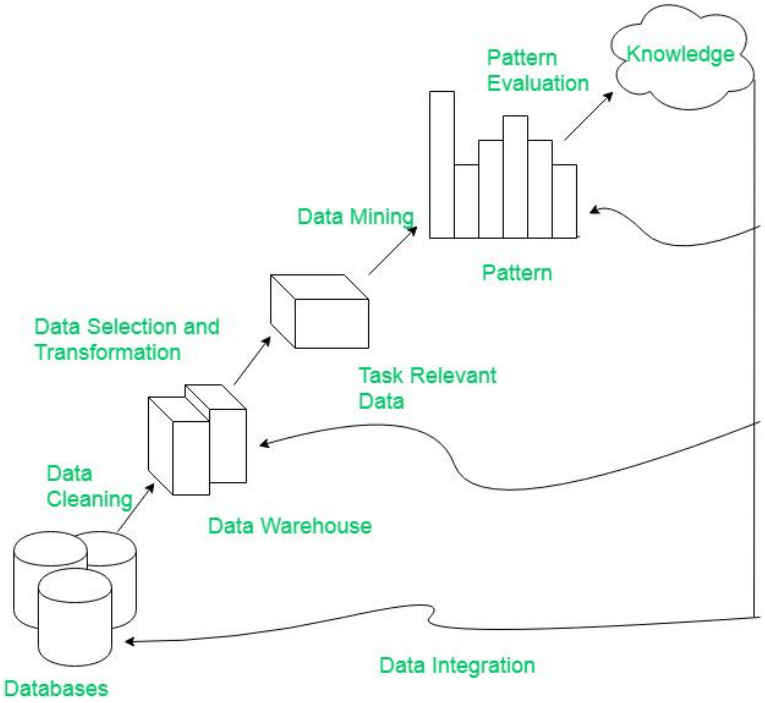
Data mining is defined as techniques that are applied to extract patterns potentially useful. It transforms task relevant data into **patterns, and d**ecides purpose of model using **classification** or **characterization**.

Pattern Evaluation

Pattern Evaluation is defined as identifying strictly increasing patterns representing knowledge based on given measures. It find **interestingness score** of each pattern, and uses **summarization** and **Visualization** to make data understandable by user.

Knowledge Representation

This involves presenting the results in a way that is meaningful and can be used to make decisions.



Note: KDD is an **iterative process** where evaluation measures can be enhanced, mining can be refined, new data can be integrated and transformed in order to get different and more appropriate results.**Preprocessing of databases** consists of **Data cleaning** and **Data Integration**.

Advantages of KDD

1. **Improves decision-making:** KDD provides valuable insights and knowledge that can help organizations make better decisions.
2. **Increased efficiency:** KDD automates repetitive and time-consuming tasks and makes the data ready for analysis, which saves time and money.
3. **Better customer service:** KDD helps organizations gain a better understanding of their customers' needs and preferences, which can help them provide better customer service.
4. **Fraud detection:** KDD can be used to detect fraudulent activities by identifying patterns and anomalies in the data that may indicate fraud.
5. **Predictive modeling:** KDD can be used to build predictive models that can forecast future trends and patterns.

Disadvantages of KDD

1. **Privacy concerns:** KDD can raise privacy concerns as it involves collecting and analyzing large amounts of data, which can include sensitive information about individuals.
2. **Complexity:** KDD can be a complex process that requires specialized skills and knowledge to implement and interpret the results.
3. **Unintended consequences:** KDD can lead to unintended consequences, such as bias or discrimination, if the data or models are not properly understood or used.
4. **Data Quality:** KDD process heavily depends on the quality of data, if data is not accurate or consistent, the results can be misleading
5. **High cost:** KDD can be an expensive process, requiring significant investments in hardware, software, and personnel.
6. **Overfitting:** KDD process can lead to overfitting, which is a common problem in machine learning where a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new unseen data.

Difference between KDD and Data Mining

Parameter	KDD	Data Mining
Definition	KDD refers to a process of identifying valid, novel, potentially useful, and ultimately understandable patterns and relationships in data.	Data Mining refers to a process of extracting useful and valuable information or patterns from large data sets.
Objective	To find useful knowledge from data.	To extract useful information from data.
Techniques Used	Data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation, and knowledge representation and visualization.	Association rules, classification, clustering, regression, decision trees, neural networks, and dimensionality reduction.
Output	Structured information, such as rules and models, that can be used to make decisions or predictions.	Patterns, associations, or insights that can be used to improve decision-making or understanding.
Focus	Focus is on the discovery of useful knowledge, rather than simply finding patterns in data.	Data mining focus is on the discovery of patterns or relationships in data.
Role of domain expertise	Domain expertise is important in KDD, as it helps in defining the goals of the process, choosing appropriate data, and interpreting the results.	Domain expertise is less critical in data mining, as the algorithms are designed to identify patterns without relying on prior knowledge.

Tasks and Functionalities of Data Mining

Data Mining functions are used to define the trends or correlations contained in data mining activities. In comparison, data mining **activities** can be divided into 2 categories:

1]Descriptive Data Mining:

This category of data mining is concerned with finding patterns and relationships in the data that can provide insight into the underlying structure of the data. Descriptive data mining is often used to summarize or explore the data, and it can be used to answer questions such as: What are the most common patterns or relationships in the data? Are there any clusters or groups of data points that share common characteristics? What are the outliers in the data, and what do they represent?
Some common techniques used in descriptive data mining include:

Cluster analysis:

This technique is used to identify groups of data points that share similar characteristics. Clustering can be used for segmentation, anomaly detection, and summarization.

Association rule mining:

This technique is used to identify relationships between variables in the data. It can be used to discover co-occurring events or to identify patterns in transaction data.

Visualization:

This technique is used to represent the data in a visual format that can help users to identify patterns or trends that may not be apparent in the raw data.

2]Predictive Data Mining: This category of data mining is concerned with developing models that can predict future behavior or outcomes based on historical data. Predictive data mining is often used for classification or regression tasks, and it can be used to answer questions such as: What is the likelihood that a customer will churn? What is the expected revenue for a new product launch? What is the probability of a loan defaulting?
Some common techniques used in predictive data mining include:

Decision trees: This technique is used to create a model that can predict the value of a target variable based on the values of several input variables. Decision trees are often used for classification tasks.

Neural networks: This technique is used to create a model that can learn to recognize patterns in the data. Neural networks are often used for image recognition, speech recognition, and natural language processing.

Regression analysis: This technique is used to create a model that can predict the value of a target variable based on the values of several input variables. Regression analysis is often used for prediction tasks.

Both descriptive and predictive data mining techniques are important for gaining insights and making better decisions. Descriptive data mining can be used to explore the data and identify patterns, while predictive data mining can be used to make predictions based on those patterns. Together, these techniques can help organizations to understand their data and make informed decisions based on that understanding.

Data Mining Functionality

1. Class/Concept Descriptions: Classes or definitions can be correlated with results. In simplified, descriptive and yet accurate ways, it can be helpful to define individual groups and concepts. These class or concept definitions are referred to as class/concept descriptions.

- **Data Characterization:** This refers to the summary of general characteristics or features of the class that is under the study. The output of the data characterization can be presented in various forms include pie charts, bar charts, curves, multidimensional data cubes.

  Example: To study the characteristics of software products with sales increased by 10% in the previous years. To summarize the characteristics of the customer who spend more than $5000 a year at AllElectronics, the result is general profile of those customers such as that they are 40-50 years old, employee and have excellent credit rating.

- **Data Discrimination:** It compares common features of class which is under study. It 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.

  Example: we may want to compare two groups of customers those who shop for computer products regular and those who rarely shop for such products(less than 3 times a year), the resulting description provides a general comparative profile of those customers, such as 80% of the customers who frequently purchased computer products are between 20 and 40 years old and have a university degree, and 60% of the customers who infrequently buys such products are either seniors or youth, and have no university degree.

2.Mining Frequent Patterns, Associations, and Correlations:

Frequent patterns are nothing but things that are found to be most common in the data. There are different kinds of frequencies that can be observed in the dataset.

- **Frequent item set:** This applies to a number of items that can be seen together regularly for eg: milk and sugar.
- **Frequent Subsequence:** This refers to the pattern series that often occurs regularly such as purchasing a phone followed by a back cover.
- **Frequent Substructure:** It refers to the different kinds of data structures such as trees and graphs that may be combined with the itemset or subsequence.

Association Analysis: The process involves uncovering the relationship between data and deciding the rules of the association. It is a way of discovering the relationship between various items.

Example: Suppose we want to know which items are frequently purchased together. An example for such a rule mined from a transactional database is,

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

where X is a variable representing a customer. 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. This association rule involves a single attribute or predicate (i.e., buys) that repeats. Association rules that contain a single predicate are referred to as single-dimensional association rules.

age (X, "20…29") ∧ income (X, "40K..49K") ⇒ buys (X, "laptop")

[support = 2%, confidence = 60%].

The **rule says that** 2% are 20 to 29 years old with an income of $40,000 to $49,000 and have purchased a laptop. There is a 60% probability that a customer in this age and income group will purchase a laptop. The association involving more than one attribute or predicate can be referred to as a multidimensional association rule.

Typically, association rules are discarded as uninteresting if they do not satisfy both a minimum support threshold and a minimum confidence threshold. Additional analysis can be performed to uncover interesting statistical correlations between associated attribute–value pairs.

Correlation Analysis: Correlation is a mathematical technique that can show whether and how strongly the pairs of attributes are related to each other.

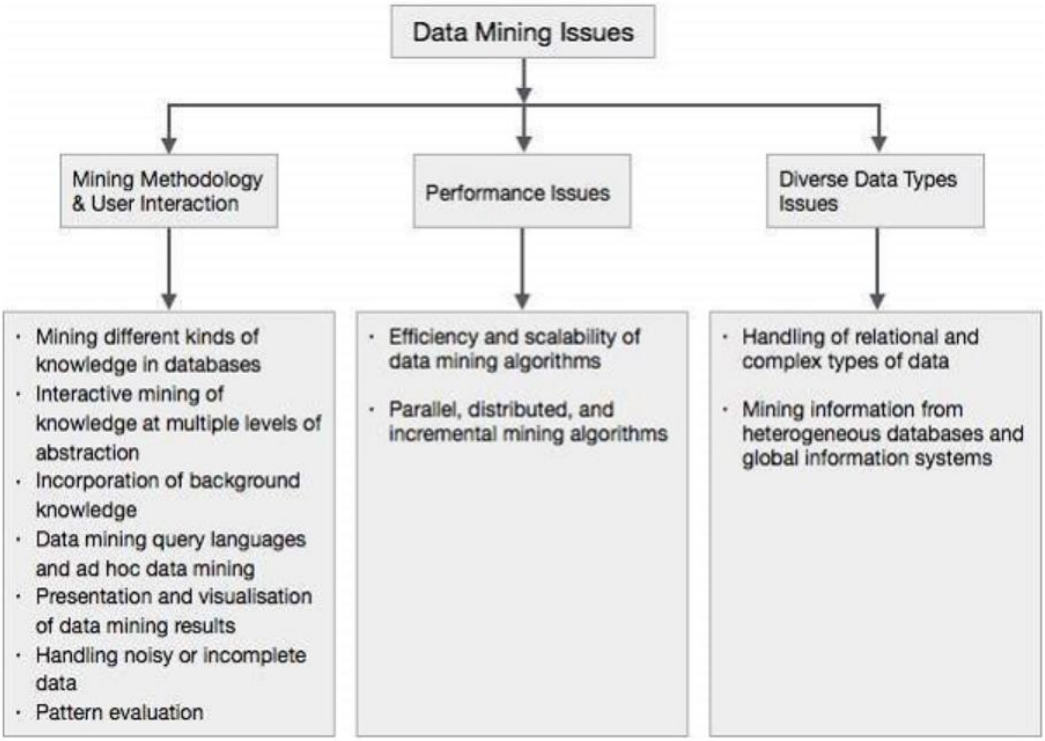
For example, Highted people tend to have more weight.

Data Mining - Issues

Data mining is not an easy task, as the algorithms used can get very complex and data is not always available at one place. It needs to be integrated from various heterogeneous data sources. These factors also create some issues. Here in this tutorial, we will discuss the major issues regarding −

- Mining Methodology and User Interaction
- Performance Issues
- Diverse Data Types Issues

The following diagram describes the major issues.



Mining Methodology and User Interaction Issues

It refers to the following kinds of 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

There can be performance-related issues such as follows −

- **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.

Data Objects and Data Attribute Types

When we talk about data mining , we usually discuss knowledge discovery from data. To learn about the data, it is necessary to discuss data objects, data attributes, and types of data attributes. Mining data includes knowing about data, finding relations between data. And for this, we need to discuss data objects and attributes.

Data objects are the essential part of a database. A data object represents the entity. Data Objects are like a group of attributes of an entity. For example, a sales data object may represent customers, sales, or purchases. When a data object is listed in a database they are called data tuples.

What are Data Attributes?

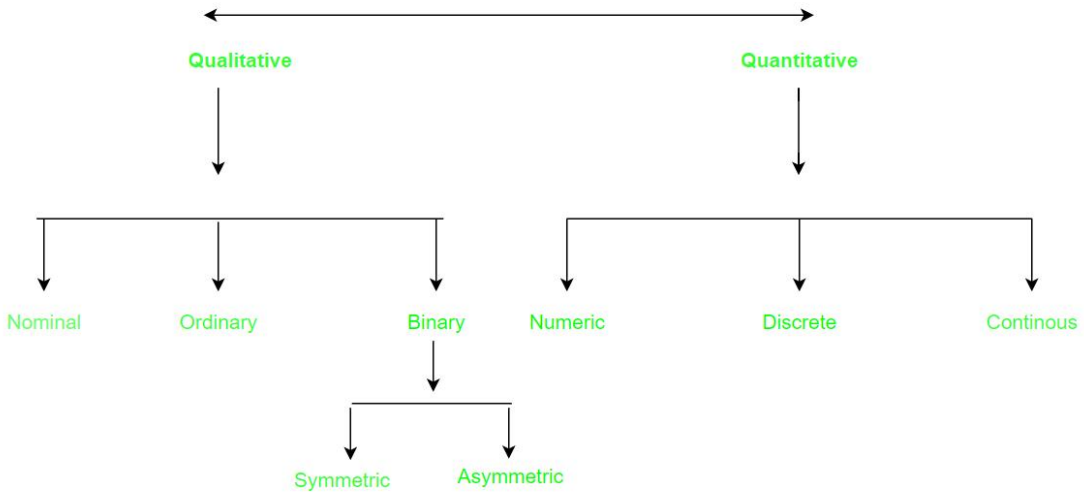
- Data attributes refer to the specific characteristics or properties that describe individual data objects within a dataset.
- These attributes provide meaningful information about the objects and are used to analyze, classify, or manipulate the data.
- Understanding and analyzing data attributes is fundamental in various fields such as statistics , machine learning , and data analysis, as they form the basis for deriving insights and making informed decisions from the data.
- Within predictive models, attributes serve as the predictors influencing an outcome. In descriptive models, attributes constitute the pieces of information under examination for inherent patterns or correlations.

We can say that a **set of attributes used to describe a given object are known as attribute vector or feature vector.**
Examples of data attributes include numerical values (e.g., age, height), categorical labels (e.g., color, type), textual descriptions (e.g., name, description), or any other measurable or qualitative aspect of the data objects.

Types of attributes:

This is the initial phase of data preprocessing involves categorizing attributes into different types, which serves as a foundation for subsequent data processing steps. Attributes can be broadly classified into two main types:

1. Qualitative (Nominal (N), Ordinal (O), Binary(B)).
2. Quantitative (Numeric, Discrete, Continuous)



Qualitative Attributes:

1. Nominal Attributes :

Nominal attributes, as related to names, refer to categorical data where the values represent different categories or labels without any inherent order or ranking. These attributes are often used to represent names or labels associated with objects, entities, or concepts.

Example :

Attribute	Values
Colours	Black, Brown, White
Categorical Data	Lecturer, Professor, Assistant Professor

2. Binary Attributes: Binary attributes are a type of qualitative attribute where the data can take on only two distinct values or states. These attributes are often used to represent yes/no, presence/absence, or true/false conditions within a dataset. They are particularly useful for representing categorical data where there are only two possible outcomes. For instance, in a medical study, a binary attribute could represent whether a patient is affected or unaffected by a particular condition.

- **Symmetric:** In a symmetric attribute, both values or states are considered equally important or interchangeable. For example, in the attribute "Gender" with values "Male" and "Female," neither value holds precedence over the other, and they are considered equally significant for analysis purposes.

Attribute	Values
Gender	Male , Female

- **Asymmetric:** An asymmetric attribute indicates that the two values or states are not equally important or interchangeable. For instance, in the attribute "Result" with values "Pass" and "Fail," the states are not of equal importance; passing may hold greater significance than failing in certain contexts, such as academic grading or certification exams

Attribute	Values
Cancer detected	Yes, No
result	Pass , Fail

3. Ordinal Attributes : Ordinal attributes are a type of qualitative attribute where the values possess a meaningful order or ranking, but the magnitude between values is not precisely quantified. In other words, while the order of values indicates their relative importance or precedence, the numerical difference between them is not standardized or known.

Example:

Attribute	Value
Grade	A,B,C,D,E,F
Basic pay scale	16,17,18

Quantitative Attributes:

1. Numeric: A numeric attribute is quantitative because, it is a measurable quantity, represented in integer or real values. Numerical attributes are of 2 types: **interval** , and **ratio-scaled.**

- An **interval-scaled** attribute has values, whose differences are interpretable, but the numerical attributes do not have the correct reference point, or we can call zero points. Data can be added and subtracted at an interval scale but can not be multiplied or divided. Consider an example of temperature in degrees Centigrade. If a day's temperature of one day is twice of the other day we cannot say that one day is twice as hot as another day.
- A **ratio-scaled** attribute is a numeric attribute with a fix zero-point. If a measurement is ratio-scaled, we can say of a value as being a multiple (or ratio) of another value. The values are ordered, and we can also compute the difference between values, and the mean, median, mode, Quantile-range, and Five number summary can be given.

2. Discrete : Discrete data refer to information that can take on specific, separate values rather than a continuous range. These values are often distinct and separate from one another, and they can be either numerical or categorical in nature.

Example:

Attribute	Value
Profession	Teacher, Business man, Peon
ZIP Code	301701, 110040

3. Continuous : Continuous data, unlike discrete data, can take on an infinite number of possible values within a given range. It is characterized by being able to assume any value within a specified interval, often including fractional or decimal values.

Example :

Attribute	Value
Height	5.4, 6.2 ...etc
weight	50.33 .....etc

What is a target attribute?

A target attribute, also known as a target variable or response variable, is a specific attribute or column in a dataset that represents the outcome or prediction target in a supervised learning problem. In supervised learning, the goal is typically to predict or model the value of the target attribute based on the values of other attributes, known as predictor variables or features.

For example, in a dataset of housing prices, the target attribute might be the sale price of houses, while the predictor variables could include attributes such as the number of bedrooms, the square footage, and the location. The target attribute is what the model aims to predict or estimate based on the input features.

Basic Statistical Descriptions of Data in Data Mining

When it comes to successful planning, having a complete grasp of your data is essential, including a statistical description of data. Some fundamental statistical reports may be used to determine the data's characteristics and the numbers that should be ignored since they are either noise or outliers.

Here we will discuss the fundamentals of describing three different statistical categories in data science. First, we use measures based on the **central tendency** to examine the middle of the data distribution. Where do most possible values for a specific attribute appear to fall intuitively? Discussion often centers on a number's mode, mode range, median, average, and other similar measures.

Descriptive Statistics

Descriptive statistics are like a translator for data that helps provide the patterns and trends of data. Descriptive statistics use various tools, such as measures of central tendency and variability and graphical representations, to present data in a meaningful and accessible way. **Central tendency** measures the typical value of a dataset, while **variability** measures show how well the spread-out data is. Graphical representations add a visual element, making it easier to spot patterns and trends. Descriptive statistics are essential for analyzing and understanding data, providing a foundation for complex statistical analysis in various fields, including business, healthcare, and social sciences.

Bar graphs, pie charts, and line graphs may be found in most statistical or graphical data presentation software programs. Quantile plots, quantile-

quantile plots, histograms, and scatter plots are more common methods for displaying data summaries and distributions.

Types of Descriptive Statistics

Descriptive Statistics can be divided in terms of data types, patterns, or characteristics- Distribution(or frequency distribution), Centra Tendency, and Variability(or Dispersion)

Frequency Distribution

The number of times a specific value appears in the data is its frequency (f). A variable's distribution is its frequency pattern or the collection of all conceivable values and the frequencies corresponding to those values. **Frequency tables** or **charts** are used to represent frequency distributions.Both categorical and numerical variables can be employed using frequency distribution tables. Only class intervals, which will be detailed momentarily, should be used with continuous variables. **Bar graphs, pie charts, and line graphs** may be found in most statistical or graphical data presentation software programs. **Quantile plots, quantile-quantile plots, histograms, and scatter plots are more common methods for displ**aying data summaries and distributions.

Central Tendency Indicators: Mean, Median, and Mode

Mean

It describes how to calculate the average of a set of data using various methods. Let's imagine that we have a database that stores object with values for attributes such as salary, and let's also pretend that we have access to this database.

Consequently, the N observations that we have of X are represented as follows: x1, x2,... xN. Within the scope of this discussion, the compilation of numerical information is typically referred to as the "data set" (for X). On what part of the scatter plot would the majority of the salary data points be located? Because of this, we are able to gain a sense of the general pattern of the data.

Indicators of central tendency include the mean, median, mode, and even the data's midpoint.

The (arithmetic) mean is the most common and trustworthy numerical measure of the "center" of a data set. It is also one of the most often used measures of central tendency. Let's assume that X is a numeric property and that x1, x2,..., and xN are the N values or observations. The sum total of these numbers is equal to

$$\text{MEAN} = \sum N \div n$$

where N is the sum of all integers
n represents the total number of integers.

DATA: { 16, 17, 10, 13, 20, 18, 13, 14, 18 }

$$\bar{x} = \frac{16 + 17 + 10 + 13 + 20 + 18 + 13 + 14 + 18}{9}$$

$$\bar{x} = \frac{139}{9}$$

$\bar{x} = 15.444$ (rounded to three decimal places)

Median

While there is no more descriptive statistic than the mean, it is not necessarily the most accurate approach to locating the data center. The mean's extreme value (or outlier) sensitivity is a serious flaw. Even a handful of outliers can skew the median.

For instance, some companies may have one or two exceptionally well-paid managers who drive up the average income for everyone else. The same holds true for test results: a small number of students with extremely poor marks can significantly lower the class average. The trimmed mean, which is the mean found by cutting off the values at the high and low ends, can be used to reduce the effect of a tiny number of extreme values. In order to find the average income, for instance, we may filter the data and discard the highest and lowest numbers. Avoiding a 20% snip at each end might save us from losing some crucial details.

The median, the middle value in a set of ordered data values, is a more appropriate measure of the center of data when the data are skewed (asymmetric). It's the threshold that divides the top half from the bottom half of a dataset.

While the median is typically used with numerical data in probability and statistics, applying the notion to ordinal data is also possible. For the sake of argument, let's say that N values for attribute X have been sorted ascendingly in a given data collection. The median of an ordered collection is the middle value if N is odd. In the case where N is an even number, the median is not a discrete value but rather the midpoint between the two extremes.

The median is calculated as the mean of the middle two values of a numerical attribute X.

Mode

The mode, in addition to the mean, can be utilized in calculating the average. The mode is the statistic representing the value that occurs the most frequently in a set of numbers.

As a consequence of this, it is possible to compute both its qualitative and quantitative features of it. It is possible for there to be a variety of modes since the frequency with the highest value might have a number of different values. Data sets can be classified as unimodal, bimodal, or trimodal, depending on the number of modes they include. When talking about multimodal data sets, it is usual practice to refer to them as having two or more modes. On the other hand, if each data value only appears once, there will be no mode.

$$\text{Mean} = \frac{\text{Sum of Observations}}{\text{Total Number of Observations}}$$

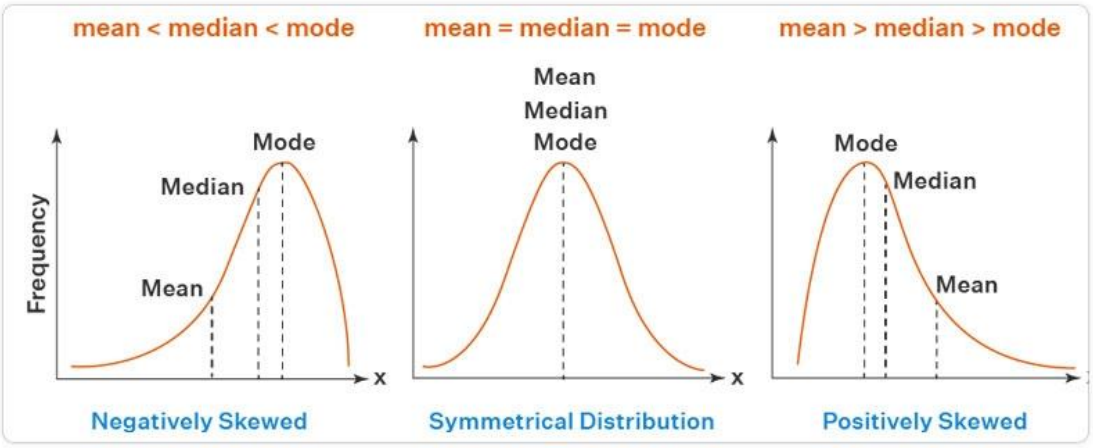
$$\text{If 'n' is odd:} \quad \text{Median} = \left(\frac{n+1}{2}\right)^{th} \text{term}$$

$$\text{If 'n' is even:} \quad \text{Median} = \frac{\left(\frac{n}{2}\right)^{th}\text{term} + \left(\frac{n}{2}+1\right)^{th}\text{term}}{2}$$

$$\text{Mode} = L + h\,\frac{(f_m - f_1)}{(f_m - f_1) + (f_m - f_2)}$$

The following empirical connection holds for unimodal frequency distributions that are somewhat asymmetrical:

Mean−Mode = 3×(Mean−Median)



As a result, when the mean and median are available, calculating the mode of a unimodal frequency distribution that is only slightly skewed is a piece of cake.

As shown in the Figure above, the mean, median, and mode are all centered on the same value in a unimodal frequency curve exhibiting perfect symmetry in the data. Unfortunately, though, the facts in most practical contexts are asymmetric. It's also possible for them to be

positively skewed, with the mode below the median, or negatively skewed, with the mode above the median.

The central tendency of data collection may also be evaluated using the midrange. It is calculated by averaging the greatest and lowest numbers in the set. SQL's max() and min() aggregate methods make it simple to calculate this algebraic metric ().

The Dispersion of Data

Before diving into further measures of data dispersion, let's first examine the range, quantiles, quartiles, percentiles, and interquartile range.

Range

Let X be a numerical attribute, and let x1, x2,..., xN be a series of observations for X. The set's range is calculated by taking the absolute value of the difference between the greatest and smallest values in the set (max()-min()).

$$\text{Range} = X_{max} - X_{min}$$

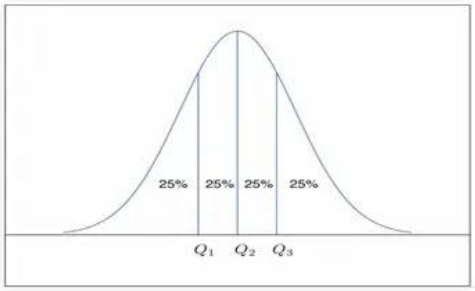
Quartile

Let's pretend that we have a set of numbers ordered from largest to smallest for attribute X. Imagine we could select particular data points to partition the data distribution equally. Quartiles are a statistical description of data measures based on these values. The quartiles of distribution are discrete numbers chosen at predetermined intervals to create groups of data that are roughly equivalent in size. (We say "basically" because there might not be values of X that divide the data into precisely equal-sized sections. For the sake of clarity, let's assume they're on par. For a given distribution of data, the kth q-quartile is the number x such that no more than k/q of the data values are fewer than x, and no more than (q k)/q of the data values are higher than x, where k is an integer such that 0 k q. Each of the q-quartiles has a probability of q-1.

The 2-quartile is the midpoint between a data set's minimum and maximum values. You may think of it as the middle number. The three data points known as the "4-quartiles" that divide the distribution into four equal parts, with each portion representing one-fourth of the whole, provide the basis for this analysis. Most people will refer to them as "quartiles" instead. The median, the quartiles, and the percentiles are the most often used quartiles, whereas the 100-quartiles are more usually known as percentiles; they divide the data distribution into 100 equal-sized sequential groupings.

The quartiles can be used to determine a distribution's shape, size, and central tendency. Q1 represents the 25th percentile or the first quartile. It discards the bottom 25% of the records.

In this case, Q3 represents the 75th percentile, which eliminates the bottom 75% (or top 25%) of the data. The 50th percentile is the middle quartile. The median represents the midpoint of a set of data.



A straightforward measure of dispersion, the gap between the first and third quartiles reveals the span of values occupied by the middle 50% of the data. The IQR formula is IQR = Q3 Q1, where Q3 and Q1 are the third and first quartiles.

$$Quartile\ Deviation = \frac{Third\ Quartile - First\ Quartile}{2}$$

$$Quartile\ Deviation\ (Q.D) = \frac{Q_3 - Q_1}{2}$$

Variance and Standard Deviation

Distributive metrics of data include variance and standard deviation. They show how dispersed a data set is. When the standard deviation is small, the data points cluster tightly around the mean, but when it's big, the data points are dispersed over a wide range of values.

The variance of N observations, x1,x2,...,xN, for a numeric attribute X

The **variance** of a population of n is $\sigma^2 = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n} = \frac{\sum_{i=1}^{n} x_i^2}{n} - \mu^2.$

The **standard deviation** of a population of n is $\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}}.$

Examples on Variance and Standard Deviation

Example 1: Find the variance and standard deviation of all the possibilities of rolling a die.

All possible outcomes of rolling a die are {1; 2; 3; 4; 5; 6}.

This data set has six values (n) = 6

Before finding the variance, we need to find the mean of the data set.

Mean, $\bar{x} = (1+2+3+4+5+6)/6 = 3.5$

We can put the value of data and mean in the formula to get;

$\sigma^2 = \Sigma (x_i - \bar{x})^2/n$

$\Rightarrow \sigma^2 = [(1-3.5)^2 + (2-3.5)^2 + (3-3.5)^2 + (4-3.5)^2 + (5-3.5)^2 + (6-3.5)^2]/6$

$\Rightarrow \sigma^2 = (6.25+2.25+0.25+0.25+2.25+6.25)/6$

Variance (σ^2) = 2.917

Now,

Standard Deviation (σ) = $\sqrt{}$ (σ^2)

$\Rightarrow$ Standard Deviation (σ) = $\sqrt{}$(2.917)

$\Rightarrow$ Standard Deviation (σ) = 1.708

Example 2: Find the variance and standard deviation of all the even numbers less than 10.

Solution:

Even Numbers less than 10 are {0, 2, 4, 6, 8}

This data set has five values (n) = 5

Before finding the variance, we need to find the mean of the data set.

Mean, $\overline{x} = (0+2+4+6+8)/5 = 4$

We can put the value of data and mean in the formula to get;

$\sigma^2 = \Sigma (x_i - \overline{x})^2/n$

$\Rightarrow \sigma^2 = [(0-4)^2 + (2-4)^2 + (4-4)^2 + (6-4)^2 + (8-4)^2]/5$

$\Rightarrow \sigma^2 = (16 + 4 + 0 + 4 + 16)/5 = 40/5$

Variance (σ^2) = 8

Now, Standard Deviation (σ) = $\sqrt{}$ (σ^2)

$\Rightarrow$ Standard Deviation (σ) = $\sqrt{}$(8)

$\Rightarrow$ Standard Deviation (σ) = 2.828

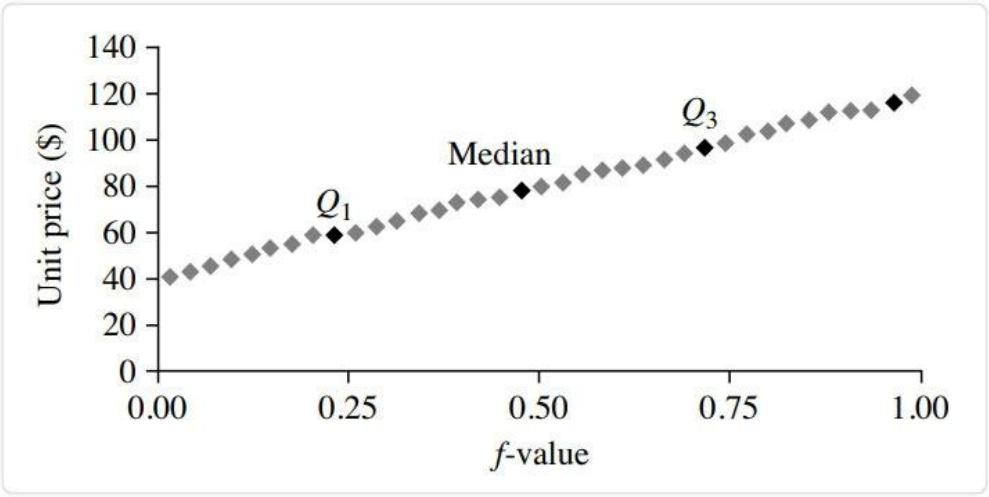
A Quantile-Quantile Plot

The quantiles of one univariate distribution are shown against the quantiles of another distribution in a q-q graphic. It's an effective mental imagery method since it permits the user to see whether there is a change while switching distributions.

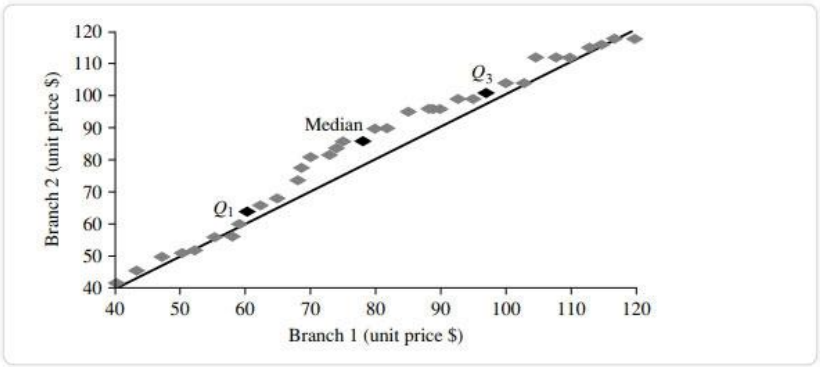
Let's pretend we have data from two branches with respect to the attribute or variable unit price. For simplicity, we will refer to the first branch's data as x1,...,xN and the second branch's data as y1,...,yM, with both sets of data being arranged from most recent to least recent. In the case where M = N (i.e., the number of points in both sets is equal), we may

plot yi versus xi, where yi and xi are the I 0.5)/N quantiles of the two sets of data.

Only M points will fit on the q-q plot if M N (the second branch contains fewer data than the first). The quantile of the y distribution at index I, expressed as yi, is given by I 0.5)/M.



Unit price ($)	Count of items sold
40	275
43	300
47	250
–	–
74	360
75	515
78	540
–	–
115	320
117	270
120	350



Distribution of the quantiles vs the quantiles. Plots unit pricing data on a quantile-quantile retail sales chart at two All Electronics locations during a specific time frame. For each data set, each dot represents a quantile,

and the unit prices at branch 1 are shown against those at branch 2 at that quantile. (The straight line is meant to be compared to when the unit price is the same at all branches for a particular quantile.The darker data points represent information from Quarter 1, the median, and Quarter 3.For example, we can observe that in the first quarter of this year, the unit price of things sold at branch 1 was somewhat lower than that at branch 2. So, 25% of sales at Store 1 were for products that cost less than equivalent to $60, but at Branch 2, only 25% of things sold were $60 or less. Half of the things sold at Branch 1 were less than $78 (the median, see Q2), and half sold at Branch 2 were less than $85 (the 25th percentile). Generally, we see that the unit prices of goods sold at branch 1 are lower than those sold at branch 2, indicating a change in distribution.

Data Visualization

Data visualization is the graphical representation of information. It translates complex data sets into visual formats that are easier for the human brain to understand. This can include a variety of visual tools such as:

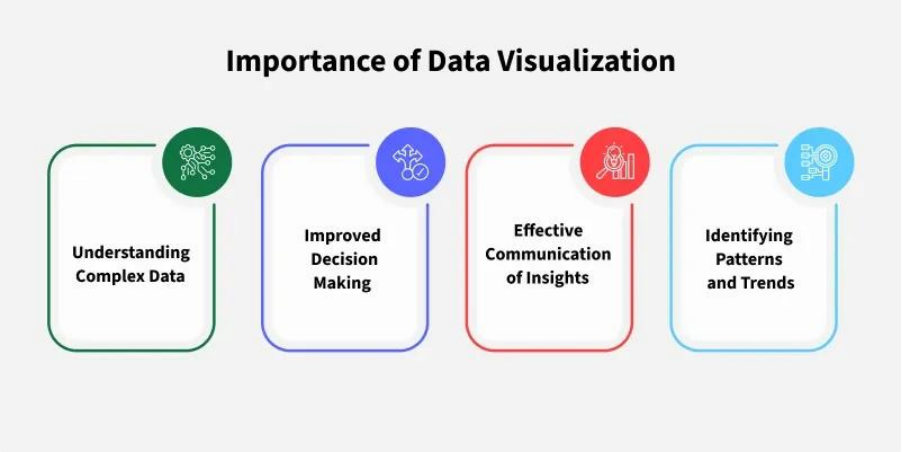
- **Charts**: Bar charts, line charts, pie charts, etc.
- **Graphs**: Scatter plots, histograms, etc.
- **Maps**: Geographic maps, heat maps, etc.
- **Dashboards**: Interactive platforms that combine multiple visualizations.

The primary goal of data visualization is to make data more accessible and easier to interpret allow users to identify patterns, trends, and outliers quickly. This is particularly important in big data where the large volume of information can be confusing without effective visualization techniques.

Why is Data Visualization Important?

Let’s take an example. Suppose you compile data of the company’s profits from 2013 *to 2023* and create a line chart. It would be very easy to see the line going constantly up with a drop in just 2018. So you can observe in a second that the company has had continuous profits in all the years except a loss in 2018.

It would not be that easy to get this information so fast from a data table. This is just one demonstration of the usefulness of data visualization. Let's see some more reasons why visualization of data is so important.



Importance of Data Visualization

1. Data Visualization Simplifies the Complex Data

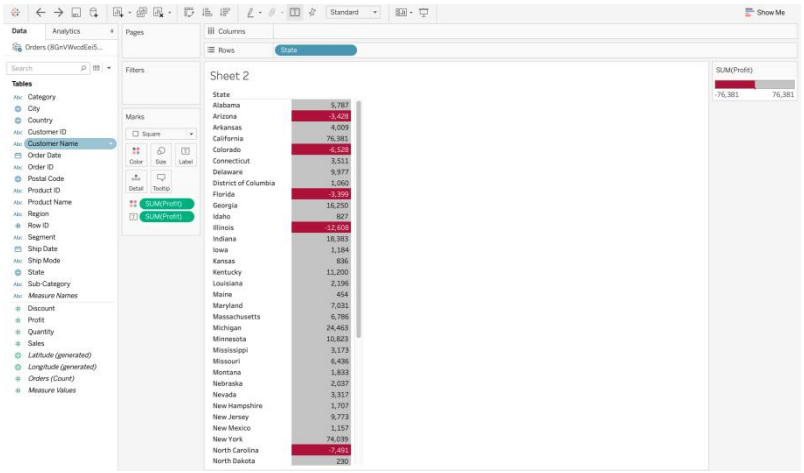
Large and complex data sets can be challenging to understand. Data visualization helps break down complex information into simpler, visual formats making it easier for the audience to grasp. For example in a scenario where sales data is visualized using a heat map on Tableau states that have suffered a net loss are colored red. This visual makes it instantly obvious which states are under-performing.

2. Enhances Data Interpretation

Visualization highlights patterns, trends, and correlations in data that might be missed in raw data form. This enhanced interpretation helps in making informed decisions. Consider another Tableau visualization that demonstrates the relationship between sales and profit. It might show that higher sales do not necessarily equate to higher profits this trend that could be difficult to find from raw data alone. This perspective helps businesses adjust strategies to focus on profitability rather than just sales volume.

3. Data Visualization Saves Time

It is definitely faster to gather some insights from the data using data visualization rather than just studying a chart. In the screenshot below on Tableau it is very easy to identify the states that have suffered a net loss rather than a profit. This is because all the cells with a loss are coloured red using a heat map, so it is obvious states have suffered a loss. Compare this to a normal table where you would need to check each cell to see if it has a negative value to determine a loss. Visualizing Data can save a lot of time in this situation.



4. Improves Communication

Visual representations of data make it easier to share findings with others especially those who may not have a technical background. This is important in business where stakeholders need to understand data-driven insights quickly. Let see the below TreeMap visualization on Tableau showing the number of sales in each region of the United States with the largest rectangle representing California due to its high sales volume. This visual context is much easier to grasp rather than detailed table of numbers.

5. Data Visualization Tells a Data Story

Data visualization is also a medium to tell a data story to the viewers. The visualization can be used to present the data facts in an easy-to-understand form while telling a story and leading the viewers to an inevitable conclusion. This data story should have a good beginning, a basic plot, and an ending that it is leading towards. For example, if a data analyst has to craft a data visualization for company executives detailing the profits of various products then the data story can start with

the profits and losses of multiple products and move on to recommendations on how to tackle the losses.

Data Visualization techniques:

1.Pixel oriented visualization techniques:

- A simple way to visualize the value of a dimension is to use a pixel where the color of the pixel reflects the dimension's value.
- For a data set of m dimensions pixel oriented techniques create m windows on the screen, one for each dimension.
- The m dimension values of a record are mapped to m pixels at the corresponding position in the windows.
- The color of the pixel reflects other corresponding values.
- Inside a window, the data values are arranged in some global order shared by all windows
- Eg: All Electronics maintains a customer information table, which consists of 4 dimensions: income, credit_limit, transaction_volume and age. We analyze the correlation between income and other attributes by visualization.
- We sort all customers in income in ascending order and use this order to layout the customer data in the 4 visualization windows as shown in fig.
- The pixel colors are chosen so that the smaller the value, the lighter the shading.
- Using pixel based visualization we can easily observe that credit_limit increases as income increases customer whose income is in the middle range are more likely to purchase more from All Electronics, these is no clear correlation between income and age.

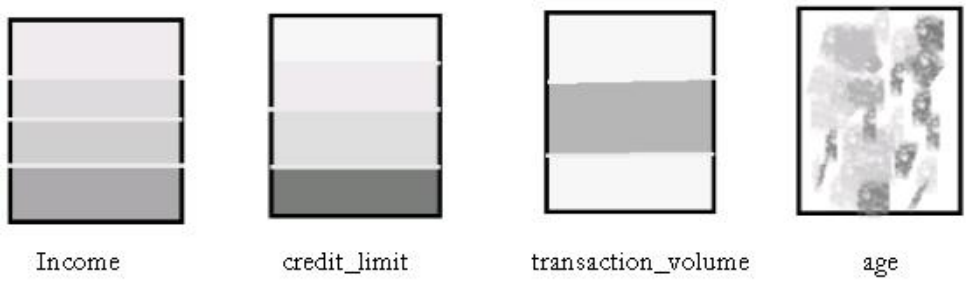


Fig: Pixel oriented visualization of 4 attributes by sorting all customers in income Ascending order.

2.Geometric Projection visualization techniques

- A drawback of pixel-oriented visualization techniques is that they cannot help us much in understanding the distribution of data in a multidimensional space.
- Geometric projection techniques help users find interesting projections of multidimensional data sets.
- A scatter plot displays 2-D data point using Cartesian co-ordinates. A third dimension can be added using different colors of shapes to represent different data points.
- Eg. Where x and y are two spatial attributes and the third dimension is represented by different shapes
- Through this visualization, we can see that points of types "+" &"X" tend to be collocated.

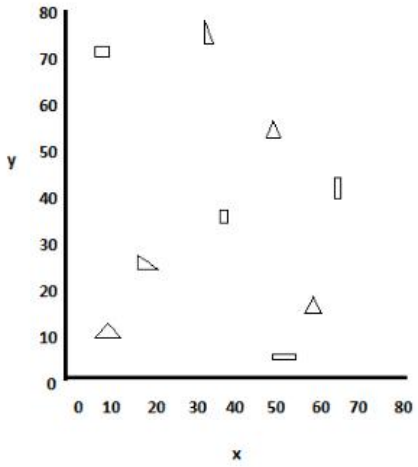


Fig: visualization of 2D data set using scatter plot

3.Icon based visualization techniques:

- It uses small icons to represent multidimensional data values
- 2 popular icon based techniques:-

3.1 Chern off faces: - They display multidimensional data of up to 18 variables as a cartoon human face.

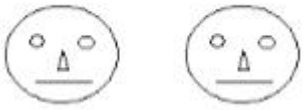


Fig: chern off faces each face represents an 'n' dimensional data points (n<18)

3.2 Stick figures: It maps multidimensional data to five –piece stick figure, where each figure has 4 limbs and a body.

- 2 dimensions are mapped to the display axes and the remaining dimensions are mapped to the angle and/ or length of the limbs.

4. Hierarchical visualization techniques (i.e. subspaces)

The subspaces are visualized in a hierarchical manner.

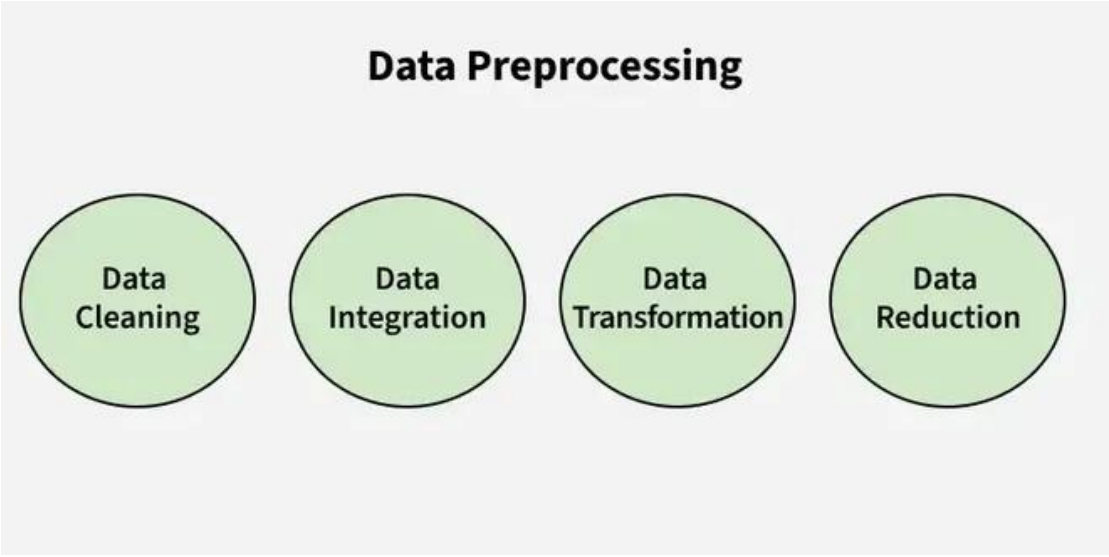
Data Preprocessing in Data Mining

Data preprocessing is the process of preparing raw data for analysis by cleaning and transforming it into a usable format. In data mining it refers to preparing raw data for mining by performing tasks like cleaning, transforming, and organizing it into a format suitable for mining algorithms.

- Goal is to improve the quality of the data.
- Helps in handling missing values, removing duplicates, and normalizing data.
- Ensures the accuracy and consistency of the dataset.

Steps in Data Preprocessing

Some key steps in data preprocessing are Data Cleaning, Data Integration, Data Transformation, and Data Reduction.



1. Data Cleaning: It is the process of identifying and correcting errors or inconsistencies in the dataset. It involves handling missing values, removing duplicates, and correcting incorrect or outlier data to ensure the dataset is accurate and reliable. Clean data is essential for effective analysis, as it improves the quality of results and enhances the performance of data models.

- **Missing Values:** This occur when data is absent from a dataset. You can either ignore the rows with missing data or fill the gaps manually, with the attribute mean, or by using the most probable value. This ensures the dataset remains accurate and complete for analysis.
- **Noisy Data:** It refers to irrelevant or incorrect data that is difficult for machines to interpret, often caused by errors in data collection or entry. It can be handled in several ways:
  - **Binning Method:** The data is sorted into equal segments, and each segment is smoothed by replacing values with the mean or boundary values.
  - **Regression:** Data can be smoothed by fitting it to a regression function, either linear or multiple, to predict values.
  - **Clustering:** This method groups similar data points together, with outliers either being undetected or falling outside the clusters. These techniques help remove noise and improve data quality.
- **Removing Duplicates:** It involves identifying and eliminating repeated data entries to ensure accuracy and consistency in the dataset. This process prevents errors and ensures reliable analysis by keeping only unique records.

2. Data Integration: It involves merging data from various sources into a single, unified dataset. It can be challenging due to differences in data formats, structures, and meanings. Techniques like record linkage and data fusion help in combining data efficiently, ensuring consistency and accuracy.

- **Record Linkage** is the process of identifying and matching records from different datasets that refer to the same entity, even if they are represented differently. It helps in combining data from various sources by finding corresponding records based on common identifiers or attributes.
- **Data Fusion** involves combining data from multiple sources to create a more comprehensive and accurate dataset. It integrates information that may be inconsistent or incomplete from different sources, ensuring a unified and richer dataset for analysis.

3. Data Transformation: It involves converting data into a format suitable for analysis. Common techniques include normalization, which scales data to a common range; standardization, which adjusts data to have zero mean and unit variance; and discretization, which converts continuous data into discrete categories. These techniques help prepare the data for more accurate analysis.

- **Data Normalization**: The process of scaling data to a common range to ensure consistency across variables.
- **Discretization**: Converting continuous data into discrete categories for easier analysis.
- **Data Aggregation**: Combining multiple data points into a summary form, such as averages or totals, to simplify analysis.
- **Concept Hierarchy Generation**: Organizing data into a hierarchy of concepts to provide a higher-level view for better understanding and analysis.

4. Data Reduction: It reduces the dataset's size while maintaining key information. This can be done through feature selection, which chooses the most relevant features, and feature extraction, which transforms the data into a lower-dimensional space while preserving important details. It uses various reduction techniques such as,

- **Dimensionality Reduction (e.g., Principal Component Analysis)**: A technique that reduces the number of variables in a dataset while retaining its essential information.
- **Numerosity Reduction**: Reducing the number of data points by methods like sampling to simplify the dataset without losing critical patterns.
- **Data Compression**: Reducing the size of data by encoding it in a more compact form, making it easier to store and process.

Uses of Data Preprocessing

Data preprocessing is utilized across various fields to ensure that raw data is transformed into a usable format for analysis and decision-making. Here are some key areas where data preprocessing is applied:

1. Data Warehousing: In data warehousing, preprocessing is essential for cleaning, integrating, and structuring data before it is stored in a centralized repository. This ensures the data is consistent and reliable for future queries and reporting.

2. Data Mining: Data preprocessing in data mining involves cleaning and transforming raw data to make it suitable for analysis. This step is crucial for identifying patterns and extracting insights from large datasets.

3. Machine Learning: In machine learning, preprocessing prepares raw data for model training. This includes handling missing values, normalizing features, encoding categorical variables, and splitting datasets into training and testing sets to improve model performance and accuracy.

4. Data Science: Data preprocessing is a fundamental step in data science projects, ensuring that the data used for analysis or building predictive models is clean, structured, and relevant. It enhances the overall quality of insights derived from the data.

5. Web Mining: In web mining, preprocessing helps analyze web usage logs to extract meaningful user behavior patterns. This can inform marketing strategies and improve user experience through personalized recommendations.

6. Business Intelligence (BI): Preprocessing supports BI by organizing and cleaning data to create dashboards and reports that provide actionable insights for decision-makers.

7. Deep Learning Purpose: Similar to machine learning, deep learning applications require preprocessing to normalize or enhance features of the input data, optimizing model training processes.

Advantages of Data Preprocessing

- **Improved Data Quality**: Ensures data is clean, consistent, and reliable for analysis.
- **Better Model Performance**: Reduces noise and irrelevant data, leading to more accurate predictions and insights.
- **Efficient Data Analysis**: Streamlines data for faster and easier processing.
- **Enhanced Decision-Making**: Provides clear and well-organized data for better business decisions.

Disadvantages of Data Preprocessing

- **Time-Consuming**: Requires significant time and effort to clean, transform, and organize data.
- **Resource-Intensive**: Demands computational power and skilled personnel for complex preprocessing tasks.
- **Potential Data Loss**: Incorrect handling may result in losing valuable information.
- **Complexity**: Handling large datasets or diverse formats can be challenging.

UNIT-II

Data Warehouse and OLAP: Data Warehouse basic concepts, Differences between Operational Database Systems and Data Warehouses, Multi tiered Architecture, Data Warehouse Models, Schemas for Multidimensional Data Models, Typical OLAP Operations, Data Warehouse Design Process, OLAP Servers.

Data Warehouse

Data Warehouse is a collection of corporate data aggregated from one or several sources. It serves as a business analytical tool, which allows analyzing and comparing data in order to solve working issues and improve business processes.

How does it work?

The concept firstly appeared in the 1980s. It was developed to support operational systems' data flow transferred to decision-making systems. These systems required the analysis of large amounts of heterogeneous data accumulated by companies over time.

Data warehouse works on the following principle:

data is extracted into one area from heterogeneous sources

⇓

converted in accordance with the needs of the decision support system

⇓

stored in the warehouse

Thus, the system provides answers for business decisions analyzing all these heterogeneous data. That is why data warehousing primarily aimed to simplify decision-making processes and helps executives to get required information based on the whole data quickly.

Data Warehouse benefits

There are two major reasons to use this technology:

1. **Quality data**: organizations add data sources in the data warehouse, so they can be sure in their relevancy and constant availability. This provides higher data quality and data integrity for informed decision-making.
2. **Promotes decision-making**: strategy decisions are based on facts and relevant data. They are supported by information that organization has collected over time. Another plus is that leaders are better informed about data requests and can extract information according to their specific requirements.

Key points

To summarize data warehouse overview, let's point out some important facts:

- It is not a database and data mart as data warehouse is much bigger and aimed at huge informational amounts analyzing;
- The system helps to promote decision-making;
- Data warehousing provides capabilities for reporting, analyzing on different aggregate levels.

Differences between Operational Database Systems and Data Warehouse

The Operational Database is the source of data for the information distribution center. It incorporates point by point data utilized to run the day to day operations of the trade. The information as often as possible changes as upgrades are made and reflect the current esteem of the final transactions. Operational Database Administration Frameworks too called as OLTP (Online Transactions Processing Databases), are utilized to oversee energetic information in real-time.

Data Stockroom Frameworks serve clients or information specialists within the reason of information investigation and decision-making. Such frameworks can organize and show data in particular designs to oblige the differing needs of different clients. These frameworks are called as Online-Analytical Processing (OLAP) Frameworks.

Difference between Operational Database and Data Warehouse:

- Operational database systems and data warehouses are two different types of database systems that are used for different purposes in organizations.
- Operational database systems are designed to support day-to-day operations of an organization. These systems are optimized for transaction processing and are used to manage and control the processes that create and deliver the organization's products or services. Examples of operational database systems include customer relationship management systems, inventory management systems, and order processing systems.
- On the other hand, data warehouses are designed to support decision-making and analysis activities within an organization. These systems are used to consolidate data from multiple operational systems and provide a unified view of the organization's data. Data warehouses are optimized for querying and reporting and are used to support business intelligence, data analysis, and decision-making activities.

Some key differences between operational database systems and data warehouses include:

1. **Purpose:** Operational database systems are used to support day-to-day operations of an organization, while data warehouses are used to support decision-making and analysis activities.
2. **Data Structure:** Operational database systems typically have a normalized data structure, which means that the data is organized into many related tables to reduce data redundancy and improve data consistency. Data warehouses, on the other hand, typically have a denormalized data structure, which means that the data is organized into fewer tables optimized for reporting and analysis.
3. **Data Volume:** Operational database systems typically store a smaller volume of data compared to data warehouses, which may store years of historical data.
4. **Performance:** Operational database systems are optimized for transaction processing and are designed to support high-volume,

high-speed transaction processing. Data warehouses, on the other hand, are optimized for querying and reporting and are designed to support complex analytical queries that may involve large volumes of data.

In summary, while operational database systems are optimized for transaction processing and day-to-day operations, data warehouses are optimized for querying and analysis to support decision-making activities.

Operational Database	Data Warehouse
Operational frameworks are outlined to back high-volume exchange preparing.	Data warehousing frameworks are regularly outlined to back high-volume analytical processing (i.e., OLAP).
It is planned for real-time commerce managing and processes.	It is outlined for investigation of commerce measures by subject range, categories, and qualities.
Relational databases are made for on-line value-based Preparing (OLTP)	Data Warehouse planned for on-line Analytical Processing (OLAP)
Operational frameworks are ordinarily optimized to perform quick embeds and overhauls of cooperatively little volumes of data.	Data warehousing frameworks are more often than not optimized to perform quick recoveries of moderately tall volumes of information.
Data In	Data out
Operational database systems are generally application-oriented.	While data warehouses are generally subject-oriented.

Multi-tier Architecture of Data Warehouse

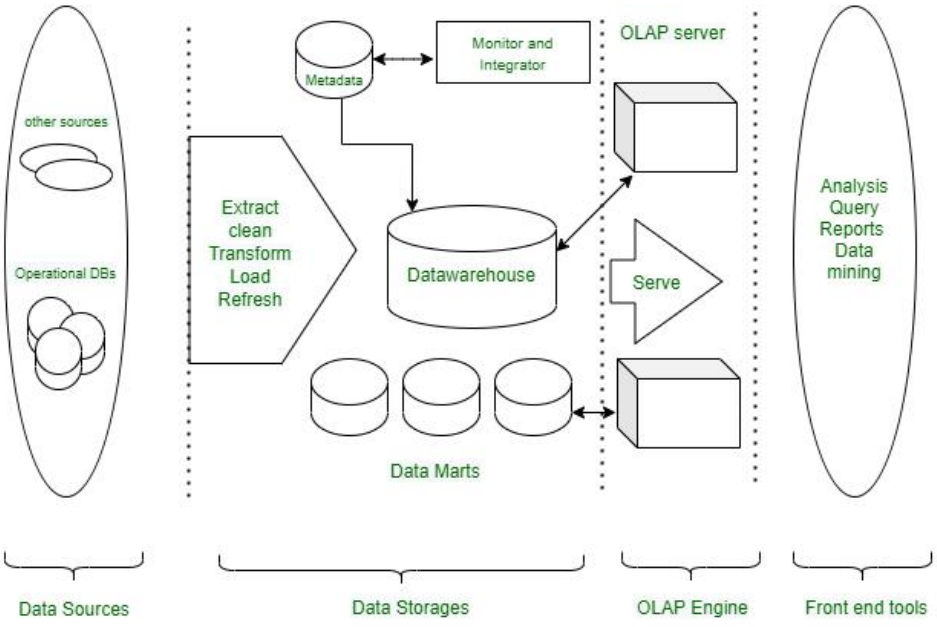
A data warehouse is Representable by data integration from multiple heterogeneous sources. It was defined by **Bill Inmon** in 1990. The data warehouse is an integrated, subject-oriented, time-variant, and non-volatile collection of data. A Data Warehouse is structured by data integration from multiple heterogeneous sources. It is a system used for data analysis and reporting.

A data warehouse is deliberate a core factor of business intelligence. BI technology provides a historical, current, and predictive view of business operations without data mining many businesses may not be able to perform effective market analysis, the strength and weakness of their competitors, profitable decisions, etc.

Data Warehouse is referred to the data repository that is maintained separately from the organization's operational data.

Multi-Tier Data Warehouse Architecture consists of the following components:

1. Bottom Tier
2. Middle Tier
3. Top Tier



Three/Multi-tier Architecture of Data Warehouse

Bottom Tier(Data sources and data storage) :

1. The bottom Tier usually consists of Data Sources and Data Storage.
2. It is a warehouse database server. For Example RDBMS.
3. In Bottom Tier, using the application program interface(called gateways), data is extracted from operational and external sources.
4. Application Program Interface likes ODBC(Open Database Connection), OLE-DB(Open-Linking and Embedding for Database), JDBC(Java Database Connection) is supported.
5. ETL stands for Extract, Transform, and Load.
   Several popular ETL tools include:
   I. IBM Infosphere
   II. Informatica
   III. Confluent
   IV. Microsoft SSIS
   V. Snaplogic
   VI. Alooma

Middle Tier :

The middle tier is an OLAP server that is typically implemented using either :
A relational OLAP (ROLAP) model (i.e., an extended relational DBMS that maps operations from standard data to standard data); **or** A multidimensional OLAP (MOLAP) model (ie, a special purpose server that directly implements multidimensional data and operations).

OLAP server models come in three different categories, including:

1. **ROLAP:** A relational database is not converted into a multidimensional database; rather, a relational database is actively broken down into several dimensions as part of relational online analytical processing(ROLAP). This is used when everything that is contained in the repository is a relational database system.
2. **MOLAP:** A different type of online analytical processing called multidimensional online analytical processing(MOLAP) includes directories and catalogs that are immediately integrated into its multidimensional database system. This is used when all that is contained in the repository is the multidimensional database system.
3. **HOLAP:** A combination of relational and multidimensional online analytical processing paradigms is hybrid online analytical processing(HOLAP). HOLAP is the ideal option for a seamless functional flow across the database systems when the repository

houses both the relational database management system and the multidimensional database management system.

Top Tier :

The top tier is a front-end client layer, which includes query and reporting tools, analysis tools, and/or data mining tools (eg, trend analysis, prediction, etc.).

Here are a few Top Tier tools that are often used:

I. SAP BW
II. SAS Business Intelligence
III. IBM Cognos
IV. Crystal Reports
V. Microsoft BI Platform

Data Warehouse Models

From the architecture point of view, there are three warehouse models-

Enterprise Warehouse:-

- An enterprise warehouse collects all information topics spread throughout the organization.
- It provides corporate-wide data integration, typically from one or several operational systems or external information providers, and is cross-functional in scope.
- It usually contains detailed data as well as summarized data and can range in size from a few gigabytes to hundreds of gigabytes, terabytes, or beyond. Can be an enterprise data warehouse.
- The traditional mainframe, computer super server, or parallel architecture has been implemented on platforms. This requires extensive commercial modeling and may take years to design and manufacture.

Data Mart:-

- A data mart contains a subset of corporate-wide data that is important to a specific group of users.
- The scope is limited to specific selected subjects.
- For example, a marketing data mart may limit its topics to customers, goods, and sales.
- The data contained in the data marts are summarized. Data marts are typically applied to low-cost departmental servers that are Unix/Linux or Windows-based.
- The implementation cycle of a data mart is more likely to be measured in weeks rather than months or years. However, it can be

in the long run, complex integration is involved in its design and planning were not enterprise-wide.

Virtual Warehouse:-

- A virtual warehouse is a group of views on an operational database.
- For efficient query processing, only a few possible summary views can be physical.
- Creating a virtual warehouse is easy, but requires additional capacity on operational database servers.

Advantages of Multi-Tier Architecture of Data warehouse

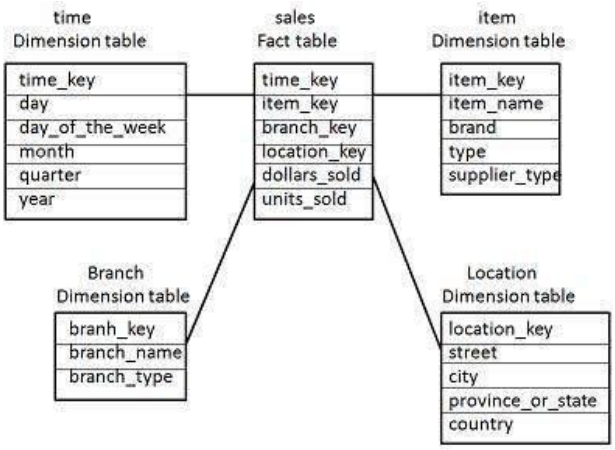
1. **Scalability**: Various components can be added, deleted, or updated in accordance with the data warehouse's shifting needs and specifications.
2. **Better Performance**: The several layers enable parallel and efficient processing, which enhances performance and reaction times.
3. **Modularity**: The architecture supports modular design, which facilitates the creation, testing, and deployment of separate components.
4. **Security**: The data warehouse's overall security can be improved by applying various security measures to various layers.
5. **Improved Resource Management**: Different tiers can be tuned to use the proper hardware resources, cutting expenses overall and increasing effectiveness.
6. **Easier Maintenance**: Maintenance is simpler because individual components can be updated or maintained without affecting the data warehouse as a whole.
7. **Improved Reliability**: Using many tiers can offer redundancy and failover capabilities, enhancing the data warehouse's overall reliability.

Schemas for Multidimensional Data Models

Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema.

Star Schema

- Each dimension in a star schema is represented with only one-dimension table.
- This dimension table contains the set of attributes.
- The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.

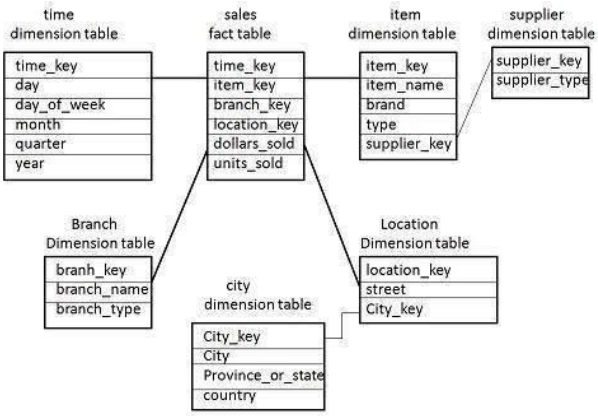


- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.

Note − Each dimension has only one dimension table and each table holds 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 cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province_or_state and country.

Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.
- Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.

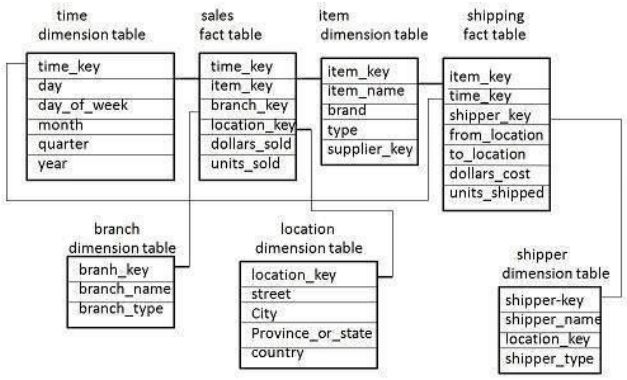


- Now the item dimension table contains the attributes item_key, item_name, type, brand, and supplier-key.
- The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier_key and supplier_type.

Note − Due to normalization in the Snowflake schema, the redundancy is reduced and therefore, it becomes easy to maintain and the save storage space.

Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.
- The following diagram shows two fact tables, namely sales and shipping.



- The sales fact table is same as that in the star schema.
- The shipping fact table has the five dimensions, namely item_key, time_key, shipper_key, from_location, to_location.
- The shipping fact table also contains two measures, namely dollars sold and units sold.
- It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

OLAP servers

Online Analytical Processing(OLAP) refers to a set of software tools used for data analysis in order to make business decisions. OLAP provides a platform for gaining insights from databases retrieved from multiple database systems at the same time. It is based on a multidimensional data model, which enables users to extract and view data from various perspectives. A multidimensional database is used to store OLAP data. Many Business Intelligence (BI) applications rely on OLAP technology.

Type of OLAP servers:

The three major types of OLAP servers are as follows:

- **ROLAP**
- **MOLAP**
- **HOLAP**

Relational OLAP (ROLAP):

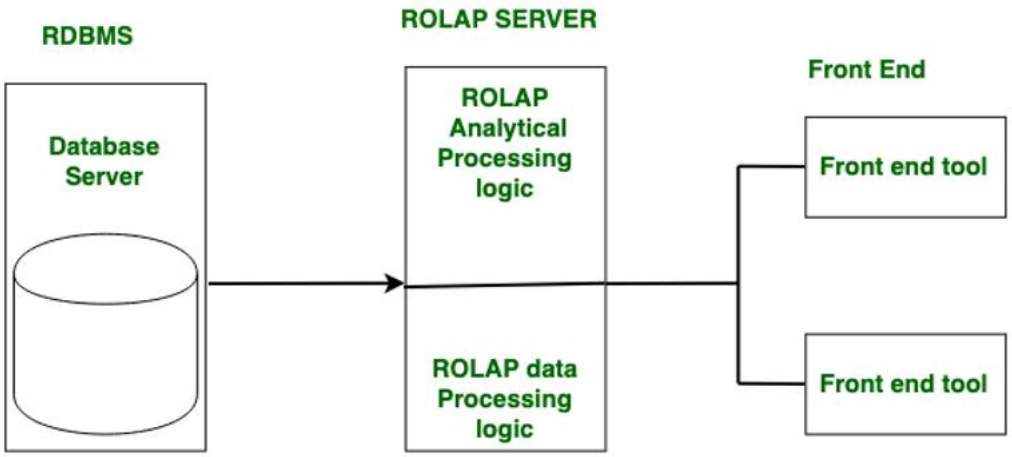
Relational On-Line Analytical Processing (ROLAP) is primarily used for data stored in a relational database, where both the base data and dimension tables are stored as relational tables. ROLAP servers are used to bridge the gap between the relational back-end server and the client's front-end tools. ROLAP servers store and manage warehouse data using RDBMS, and OLAP middleware fills in the gaps.

Benefits:

- It is compatible with data warehouses and OLTP systems.
- The data size limitation of ROLAP technology is determined by the underlying RDBMS. As a result, ROLAP does not limit the amount of data that can be stored.

Limitations:

- SQL functionality is constrained.
- It's difficult to keep aggregate tables up to date.



Multidimensional OLAP (MOLAP):

Through array-based multidimensional storage engines, Multidimensional On-Line Analytical Processing (MOLAP) supports multidimensional views of data. Storage utilization in multidimensional data stores may be low if the data set is sparse.

MOLAP stores data on discs in the form of a specialized multidimensional array structure. It is used for OLAP, which is based on the arrays' random access capability. Dimension instances determine array elements, and the data or measured value associated with each cell is typically stored in the corresponding array element. The multidimensional array is typically stored in MOLAP in a linear

allocation based on nested traversal of the axes in some predetermined order.

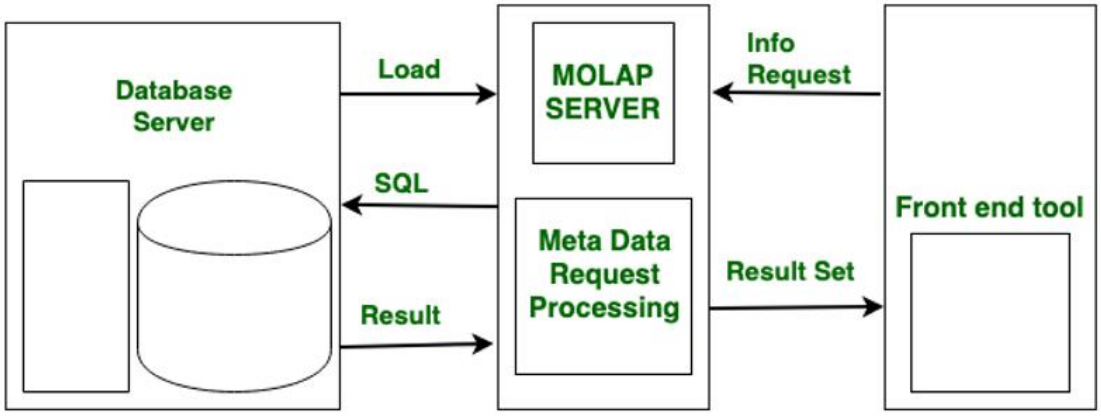
However, unlike ROLAP, which stores only records with non-zero facts, all array elements are defined in MOLAP, and as a result, the arrays tend to be sparse, with empty elements occupying a larger portion of them. MOLAP systems typically include provisions such as advanced indexing and hashing to locate data while performing queries for handling sparse arrays, because both storage and retrieval costs are important when evaluating online performance. MOLAP cubes are ideal for slicing and dicing data and can perform complex calculations. When the cube is created, all calculations are pre-generated.

Benefits:

- Suitable for slicing and dicing operations.
- Outperforms ROLAP when data is dense.
- Capable of performing complex calculations.

Limitations:

- It is difficult to change the dimensions without re-aggregating.
- Since all calculations are performed when the cube is built, a large amount of data cannot be stored in the cube itself.



Hybrid OLAP (HOLAP):

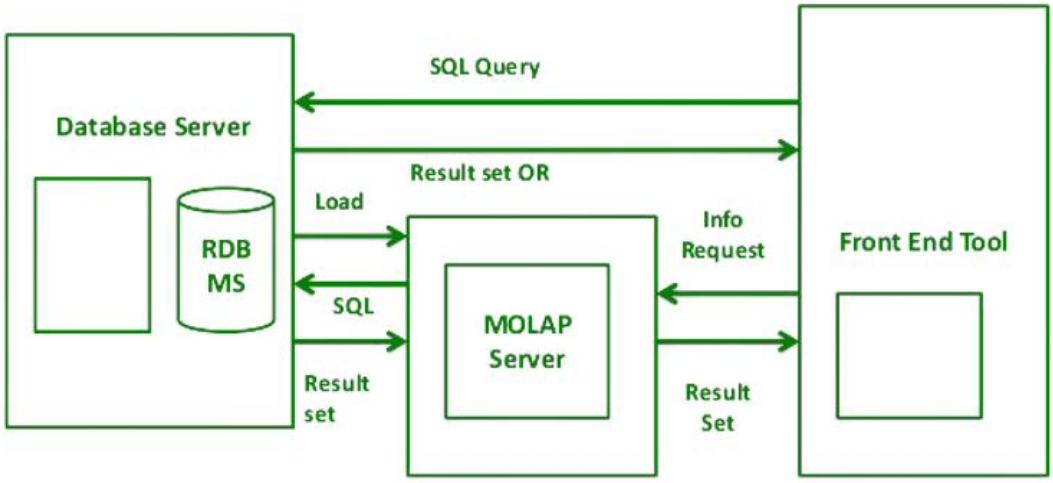
ROLAP and MOLAP are combined in Hybrid On-Line Analytical Processing (HOLAP). HOLAP offers greater scalability than ROLAP and faster computation than MOLAP.HOLAP is a hybrid of ROLAP and MOLAP. HOLAP servers are capable of storing large amounts of detailed data. On the one hand, HOLAP benefits from ROLAP's greater scalability. HOLAP, on the other hand, makes use of cube technology for faster performance and summary-type information. Because detailed data is stored in a relational database, cubes are smaller than MOLAP.

Benefits:

- HOLAP combines the benefits of MOLAP and ROLAP.
- Provide quick access at all aggregation levels.

Limitations

- Because it supports both MOLAP and ROLAP servers, HOLAP architecture is extremely complex.
- There is a greater likelihood of overlap, particularly in their functionalities.



Typical OLAP Operations

Since OLAP servers are based on multidimensional view of data, we will discuss OLAP operations in multidimensional data.

Here is the list of OLAP operations −

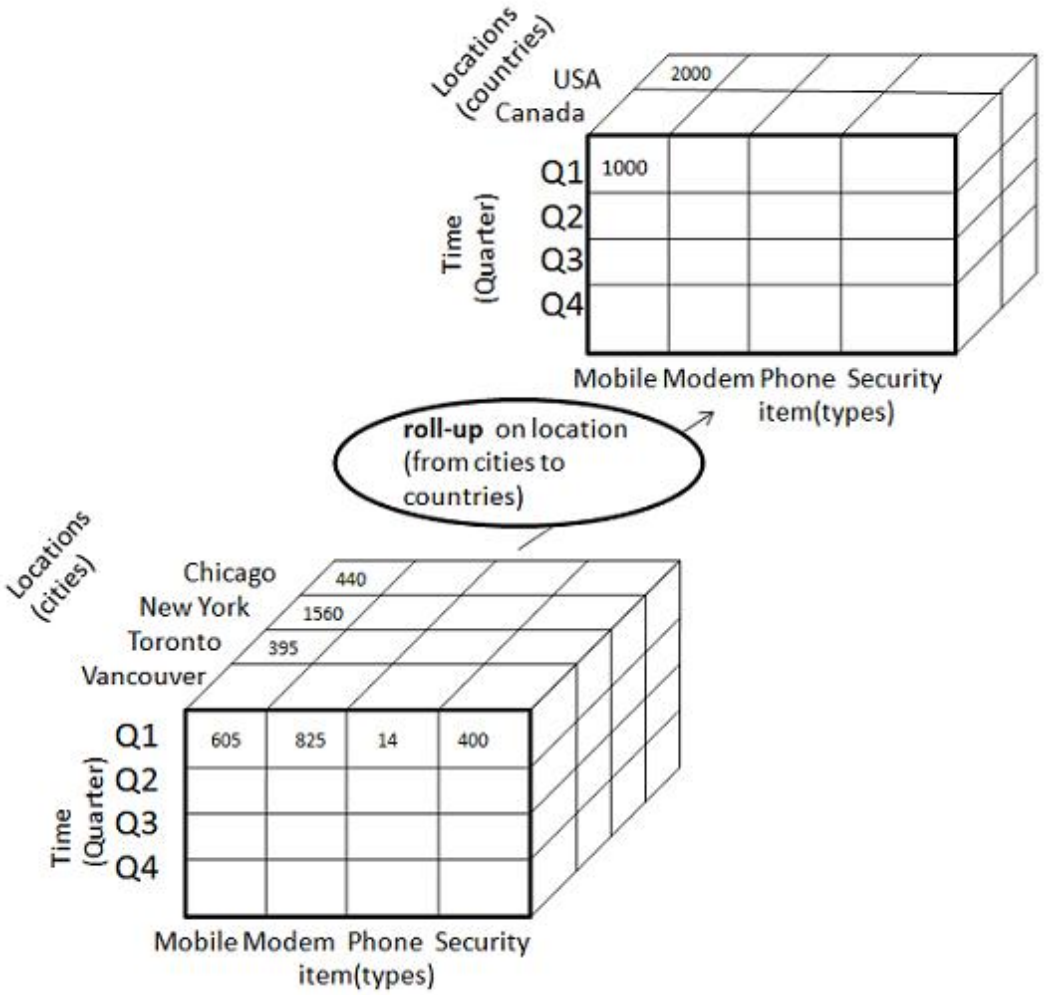
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

Roll-up

Roll-up performs aggregation on a data cube in any of the following ways −

- By climbing up a concept hierarchy for a dimension
- By dimension reduction

The following diagram illustrates how roll-up works.



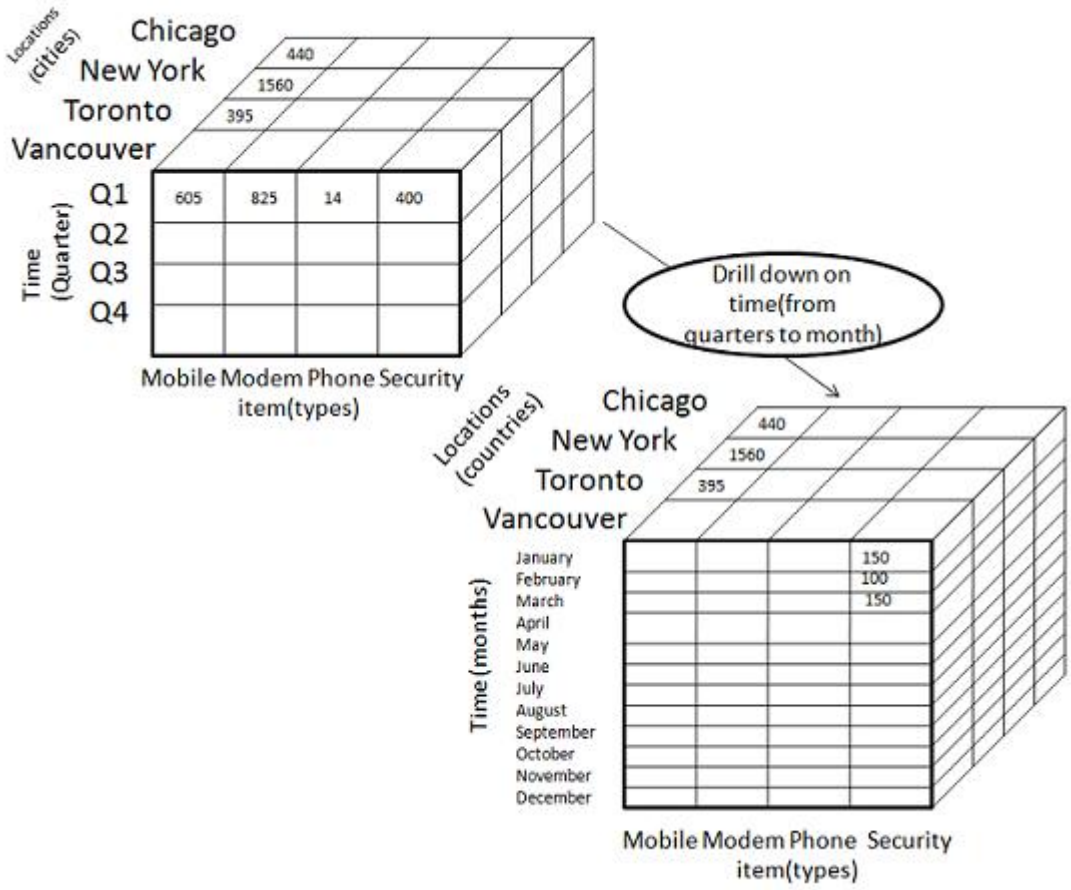
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.
- The data is grouped into cities rather than countries.
- When roll-up is performed, one or more dimensions from the data cube are removed.

Drill-down

Drill-down is the reverse operation of roll-up. It is performed by either of the following ways −

- By stepping down a concept hierarchy for a dimension
- By introducing a new dimension.

The following diagram illustrates how drill-down works −

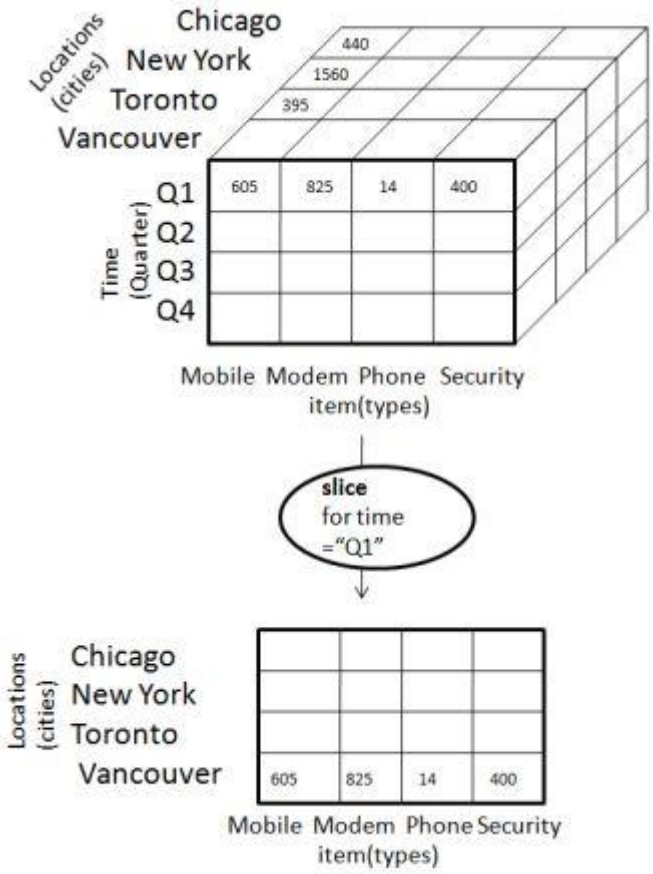


- Drill-down is performed by stepping down a concept hierarchy for the dimension time.
- Initially the concept hierarchy was "day < month < quarter < year."
- On drilling down, the time dimension is descended from the level of quarter to the level of month.
- When drill-down is performed, one or more dimensions from the data cube are added.

- It navigates the data from less detailed data to highly detailed data.

Slice

The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.

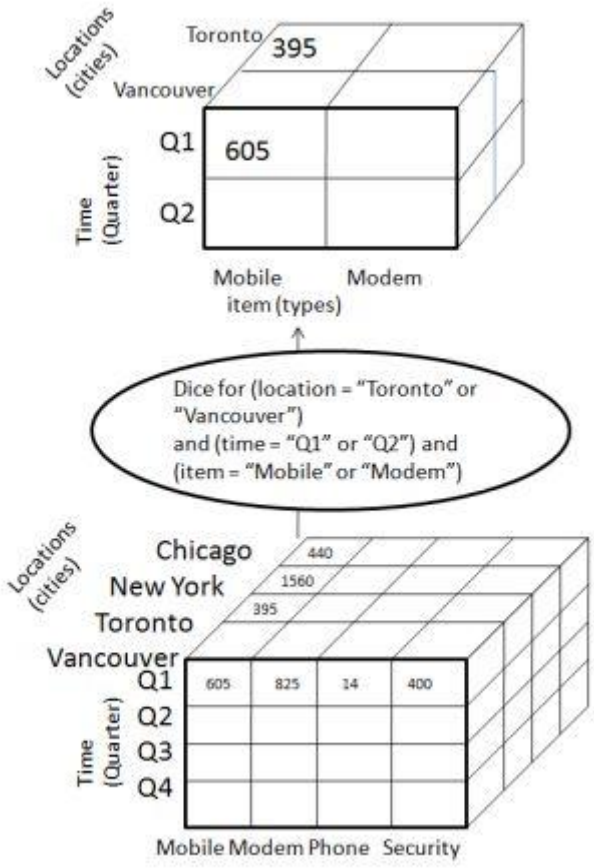


- Here Slice is performed for the dimension "time" using the criterion time = "Q1".

- It will form a new sub-cube by selecting one or more dimensions.

Dice

Dice selects two or more dimensions from a given cube and provides a new sub-cube. Consider the following diagram that shows the dice operation.

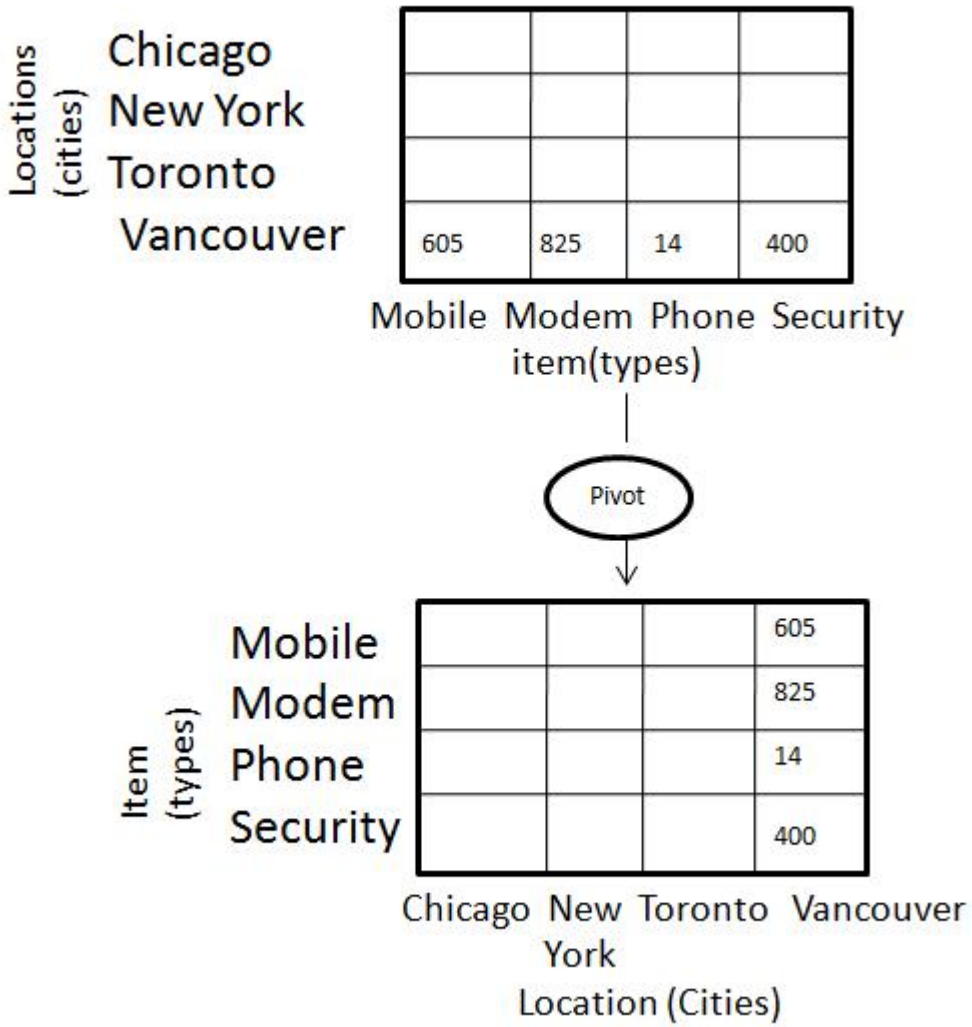


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (location = "Toronto" or "Vancouver")
- (time = "Q1" or "Q2")
- (item =" Mobile" or "Modem")

Pivot

The pivot operation is also known as rotation. It rotates the data axes in view in order to provide an alternative presentation of data. Consider the following diagram that shows the pivot operation.



Data Warehouse Design Process

The demand to store and analyze vast data for business growth necessitates careful design of data warehouse architecture. Various approaches exist for utilizing its components effectively. Guidelines ensure adherence to standards in designing segments of the warehouse. Here will explores different frameworks and approaches, emphasizing the importance of aligning architecture with business needs.

The Two Approaches to Designing the Architecture of a Data Warehouse

When we are designed to build an architecture of a data warehouse, it must always be taken care of for the data model that needs to be integrated. With that case coming up, a **data model** provides a framework and a set of best practices to follow when designing the architecture or troubleshooting issues. As a data warehouse is a

heterogeneous collection of different data sources that are organized under a unified schema, we can broadly follow two approaches for constructing a data warehouse which are explained below :

1. **The Top-down approach**
2. **The Bottom-up approach**

1. The Top-down approach:

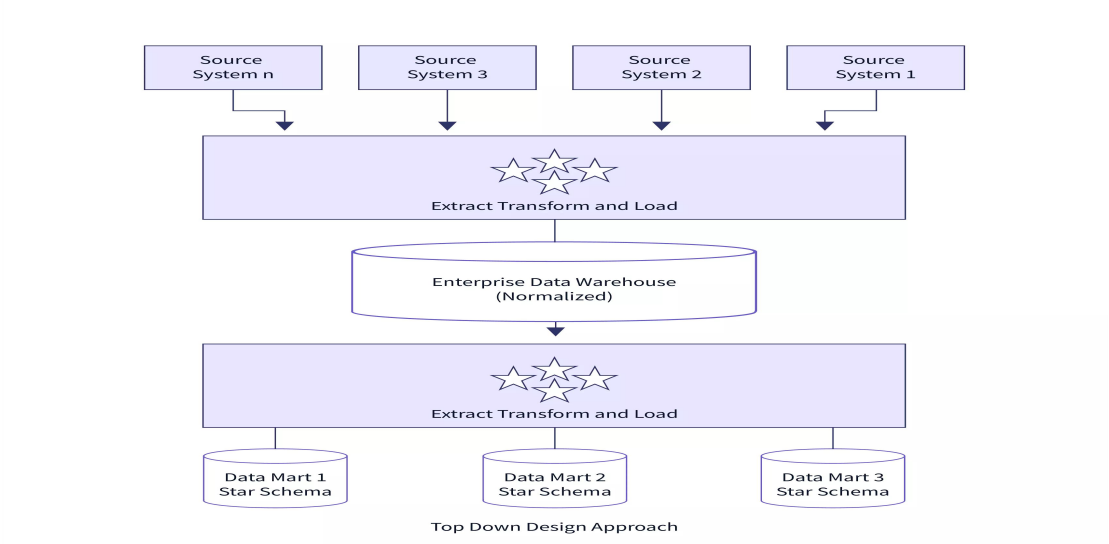
This approach was coined by Inmon, and it can be defined as the data warehouse in this approach acts as a **central information repository** for the complete enterprise, and then the data marts are created from it after the complete data warehouse has been set up.

Elaborating on the "**Top-Down**" design approach, where we pick data from different sources which are validated, reformatted, and saved in a **normalized (up to 3NF)** database as the data warehouse as here we consider the data warehouse as a subject-oriented, time-variant, **non-volatile and integrated data repository** for the entire organization.

The **Top-Down** approach is considered a data-driven approach as the information is collected and integrated first, and then the business requirements by **subjects(or themes)** are subjected to building the data marts. The data marts that we built from this approach while designing the architecture of the data warehouse will still have **consistency when the data overlap** as this approach supports a single integrated data source.

It is widely observed that from the top-down approach to the architecture of the data warehouse, the data marts are built from the data which are selected for specific business subjects or **particular departments** as the data collected here stores the atomic information, that is, the lowest level of granularity.

Below is the pictorial representation of the **TOP_DOWN APPROACH** of the Data Warehouse Architecture:



Components of the Top-Down Approach

A. **The External Sources**: The data or the raw data is collected from the external source, that is, the source of truth, which the organization needs to decide as a best practice for designing the architecture of the data warehouse. The external source is a source from where the data is collected, irrespective of the type of data. The Data is mainly of three different forms, which are :

- **Structured (CSV, excel sheets, relational database, etc)**

- **Semi-structured (HTML, JSON, XML)**

- **Unstructured (audio, video, pdf, etc)**

B. The Stage Area: After the extraction of data is done from the external sources, we see that the data does not follow a particular format that is, some are logical values, numerical values, etc., so to make the standardized data format we need to validate this source data before we load into the data warehouse.

And to solve this we have the **ETL tool**. The ETL tool or the Extract-Transform-Load tool helps in cleansing and transforming the data to serve the **business needs**.

E(Extracted): Here, the raw data is extracted from the external data source.

T(Transform): Here, the raw data that we received is transformed into the standard format, which is universally acceptable and reliable, along with serving the business needs. We make use of the Query Tools at this **stage of data transformation**.

L(Load): Here, the refined-transformed data is loaded into the data warehouse for further analysis to **gain insights** that can help a business grow.

C. The Data Marts: The third component in the Top-down approach while designing the architecture of the data warehouse is the **Datamart**, which can also be suggested as part of the storage component. The data mart stores the information or the transformed data, which is of a particular function/theme of an enterprise mostly handled by a single authority.

We can have as many data marts in an organization that majorly depends upon the subjects/theme. This is widely known that the data mart is also called the subset of the data warehouse as the data stored in the data mart is the same as in the data warehouse **compartmentalized** under different themes.

D. The Data Mining:

The **fourth major component** in the Top-down approach while designing the architecture of the data warehouse is data mining. The raw data was gathered at the External source after the transformation reached the data warehouse. Now this cleansed and transformed data will be of no use until the analyst makes the best use of

the same. So data mining can be explained as the ability to **analyze the transformed data** to find out the hidden patterns that are present in the database or the data warehouse with the help of an algorithm of data mining.

As we know, every architecture has its ability to be modified, and so we have certain **advantages and disadvantages associated**:

Advantages of the Top-Down Approach

There is a consistency that is maintained with the data marts as it is a subset of the data warehouse where the data was cleansed, transformed, and stored. It helps create a **consistent dimensional** view of the data marts even when they overlap. This also goes for the easier and quicker approach towards creating the **data marts**, which can then be used by the analyst.

This is a **widely used approach** as this is considered the strongest business model for designing the **architecture of the data warehouse**.

The Disadvantages of the Top-Down Approach

The **major disadvantage** of the top-down approach is the cost, the time taken in designing the architecture of the data warehouse and its maintenance is also **very costly**.

2. The Bottom-up Approach

This approach was coined by Kimball as it can be defined as data after extraction from the source is cleansed and transformed by the staging area, after which the data is sent to the **data marts** of each theme/subject, and then it is loaded up in the data warehouse.

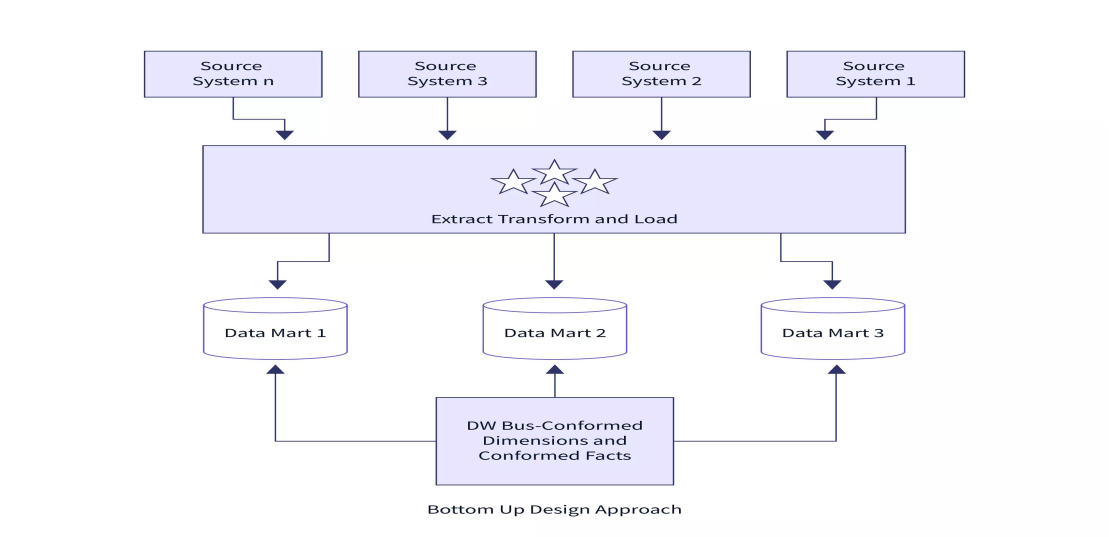
Elaborating on the Bottom-Up design approach, here, similar to the **Top-Up approach** we pick data from different sources, which are validated, cleansed, reformatted, and saved in a data mart instead of the data warehouse of the particular themes/ subjects.

The data marts which are created and store the transformed data start to **provide the reporting capability** to the analyst. As the data marts are based on particular subjects, these address only a single business area.

After the data is saved in data marts, these are then loaded and integrated into the data warehouse.

It can also be described as a copy of the transaction data specifical architecture for **query and analysis**, which is the same for the star schema.

Below is the pictorial representation of the **BOTTOM-UP APPROACH** of the Data Warehouse Architecture:



Advantages of the Bottom-Up Approach

1. The analyst can start generating the reports early as the data marts are readily available after the transformation process in the staging area, which means that **working towards the decision** can be started early, therefore, enabling better decision-making.
2. As the number of data marts that are created can be more which means we can extend the data warehouse
3. The **cost and time taken in designing** the architecture of the data warehouse by the bottom-up approach are low comparatively.
4. The documents can be generated quickly as the data marts are saved with data early.
5. This approach helps in developing the new data marts and then **integrating them** with the other data marts.
6. This approach allows the analysts of the team to learn and grow

The Disadvantage of the Bottom-Up Approach

1. As the **dimensional view of data marts** is not consistent, as was seen in the Top-Down approach, we can say that the model is not strong enough comparatively.
2. The cost of **implementation of the project** is high.
3. As the architecture of the data warehouse needs to be **flexible to accustom** to any changes concerning the needs generated within the organization, we can say that the Bottom-up approach is inflexible to it.