MCS-221: Data Warehousing and Data Mining
Guess Paper-1

Q. What is data Warehouse?

Ans. The term "Data Warehouse" was first coined by Bill Inmon in 1990. According to Inmon, a data warehouse is a subject oriented, integrated, time-variant, and non-volatile collection of data. This data helps analysts to take informed decisions in an organization.

An operational database undergoes frequent changes on a daily basis on account of the transactions that take place. Suppose a business executive wants to analyze previous feedback on any data such as a product, a supplier, or any consumer data, then the executive will have no data available to analyze because the previous data has been updated due to transactions.

A data warehouses provides us generalized and consolidated data in multidimensional view. Along with generalized and consolidated view of data, a data warehouses also provides us Online Analytical Processing (OLAP) tools. These tools help us in interactive and effective analysis of data in a multidimensional space. This analysis results in data generalization and data mining.

Data mining functions such as association, clustering, classification, prediction can be integrated with OLAP operations to enhance the interactive mining of knowledge at multiple level of abstraction. That's why data warehouse has now become an important platform for data analysis and online analytical processing.

Understanding a Data Warehouse:

- A data warehouse is a database, which is kept separate from the organization's operational database.
- There is no frequent updating done in a data warehouse.
- It possesses consolidated historical data, which helps the organization to analyze its business.
- A data warehouse helps executives to organize, understand, and use their data to take strategic decisions.
- Data warehouse systems help in the integration of diversity of application systems.
- A data warehouse system helps in consolidated historical data analysis.

Q. Explain the concepts of data Warehouse?

Ans. Data warehousing is the process of constructing and using a data warehouse. A data warehouse is constructed by integrating data from multiple heterogeneous sources that support analytical reporting, structured and/or ad hoc queries, and decision making. Data warehousing involves data cleaning, data integration, and data consolidations.

Using Data Warehouse Information: There are decision support technologies that help utilize the data available in a data warehouse. These technologies help executives to use the warehouse quickly and effectively. They can gather data, analyze it, and take decisions based on the information present in the warehouse. The information gathered in a warehouse can be used in any of the following domains –

Tuning Production Strategies: The product strategies can be well tuned by repositioning the products and managing the product portfolios by comparing the sales quarterly or yearly.

Customer Analysis: Customer analysis is done by analyzing the customer's buying preferences, buying time, budget cycles, etc.

Operations Analysis: Data warehousing also helps in customer relationship management, and making environmental corrections. The information also allows us to analyze business operations.

Integrating Heterogeneous Databases: To integrate heterogeneous databases, we have two approaches

1. **Query-Driven Approach:** This is the traditional approach to integrate heterogeneous databases. This approach was used to build wrappers and integrators on top of multiple heterogeneous databases. These integrators are also known as mediators.

2. **Process of Query-Driven Approach:** When a query is issued to a client side, a metadata dictionary translates the query into an appropriate form for individual heterogeneous sites involved.

Now these queries are mapped and sent to the local query processor.
The results from heterogeneous sites are integrated into a global answer set.

Disadvantages:

- Query-driven approach needs complex integration and filtering processes.
- This approach is very inefficient.
- It is very expensive for frequent queries.
- This approach is also very expensive for queries that require aggregations.

(2) **Update-Driven Approach:** This is an alternative to the traditional approach. Today's data warehouse systems follow update-driven approach rather than the traditional approach discussed earlier. In update-driven approach, the information from multiple heterogeneous sources are integrated in advance and are stored in a warehouse. This information is available for direct querying and analysis.

Advantages: This approach has the following advantages –

- This approach provide high performance.
- The data is copied, processed, integrated, annotated, summarized and restructured in semantic data store in advance.
- Query processing does not require an interface to process data at local sources.

Functions of Data Warehouse Tools and Utilities: The following are the functions of data warehouse tools and utilities

- **Data Extraction:** Involves gathering data from multiple heterogeneous sources.
- **Data Cleaning:** Involves finding and correcting the errors in data.
- **Data Transformation:** Involves converting the data from legacy format to warehouse format.
- **Data Loading:** Involves sorting, summarizing, consolidating, checking integrity, and building indices and partitions.
- **Refreshing:** Involves updating from data sources to warehouse.

Q. Differentiate between OLTP and Data Warehouse?

Ans. Difference between Data Warehousing and Online-Transaction processing (OLTP) is as follows:

Data Warehousing DWH	Online Transaction
It is technique that gathers or collect data from different sources into central repository.	It is technique that is used for detailed day to day transaction data which keep chaining on everyday.
It is designed for decision making process.	It is designed for business transaction process.
It stores large amount of data or historical data.	It holds current data.
It used for analyzing the business.	It used for running the business.
In Data warehousing, the size of database is around 100GB-2TB .	In Online transaction processing, the size of data base is around 10MB-100GB.
In Data warehousing, denormalized data is present.	In Online transaction processing, normalized data is present.
It uses Query processing.	It uses transaction processing
It is subject-oriented.	It is application-oriented.
In Data warehousing, data redundancy is present.	In Online transaction processing, there is no data redundancy.

Q. Briefly describe the need and advantages of data Warehouse?

Ans. Data Warehouse is needed for the following reasons:

(1) **Business User:** Business users require a data warehouse to view summarized data from the past. Since these people are non-technical, the data may be presented to them in an elementary form.

(2) **Store historical data:** Data Warehouse is required to store the time variable data from the past. This input is made to be used for various purposes.

(3) **Make strategic decisions:** Some strategies may be depending upon the data in the data warehouse. So, data warehouse contributes to making strategic decisions.

(4) **For data consistency and quality:** Bringing the data from different sources at a commonplace, the user can effectively undertake to bring the uniformity and consistency in data.

(5) **High response time:** Data warehouse has to be ready for somewhat unexpected loads and types of queries, which demands a significant degree of flexibility and quick response time.

Benefits of Data Warehouse

(1) Understand business trends and make better forecasting decisions.

(2) Data Warehouses are designed to perform well enormous amounts of data.

(3) The structure of data warehouses is more accessible for end-users to navigate, understand, and query.

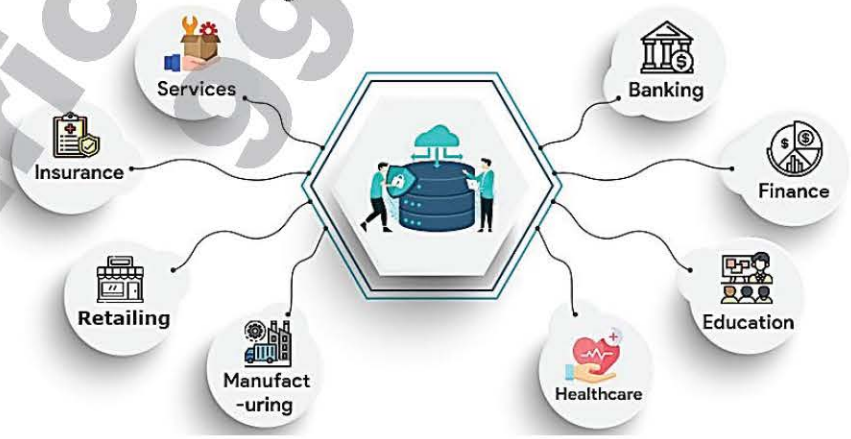
(4) Queries that would be complex in many normalized databases could be easier to build and maintain in data warehouses.

(5) Data warehousing is an efficient method to manage demand for lots of information from lots of users.

(6) Data warehousing provide the capabilities to analyze a large amount of historical data.

Q. List the Applications of data Warehouse?

Ans. Applications of Data Warehousing:



Every organization, no matter in what industry it works in or how big or small it is, requires a data warehouse to connect its disparate sources for anticipating, analysis, reporting, business intelligence, and facilitating robust decision-making. Here, we are listing down the best applications of data warehousing across different industries.

- **Banking:** With the perfect Data Warehousing solution, bankers can manage all their available resources more effectively. They can better analyze their consumer data, government regulations, and market trends to facilitate better decision-making.
- **Finance:** The application of data warehousing in the financial industry is the same as in the banking sector. The right solution helps the financing industry analyze customer expenses that enable them to outline better strategies to maximize profits at both ends.
- **Education:** The educational sector requires data warehousing to have a comprehensive view of their students' and faculty data. It provides educational institutions access to real-time data feeds to make valued and informed decisions.
- **Healthcare:** Another critical use of data warehouses is in the Healthcare sector. All the clinical, financial, and employee data are stored in the warehouse, and analysis is run to derive valuable insights to strategize resources in the best way possible.
- **Manufacturing & Distribution:** With an effective data warehousing solution, organizations in the manufacturing & distribution sector can organize all their data under one roof and predict market changes, analyze the latest trends, view development areas, and finally can make result-driven decisions.
- **Retailing:** Retailers are the mediators between wholesalers and end customers, and that's why it is necessary for them to maintain the records of both parties. For helping them store data in an organized manner, the application of data warehousing comes into the frame.
- **Insurance:** In the Insurance sector, data warehousing is required to maintain existing customers' records and analyze the same to up see client trends to bring more footsteps towards the business.
- **Services:** In the services sector, data warehousing is used for maintaining customer details, financial records, and resources to analyze patterns and boost decision-making for positive outcomes.

To Sum it up: A data warehouse improves the decision-making process of a business and boosts organizational performance.

Q. Explain data warehouse Architecture?

Ans. A data warehouse architecture is a method of defining the overall architecture of data communication processing and presentation that exist for end-clients computing within the enterprise. Each data warehouse is different, but all are characterized by standard vital components.

Production applications such as payroll accounts payable product purchasing and inventory control are designed for online transaction processing (OLTP). Such applications gather detailed data from day to day operations.

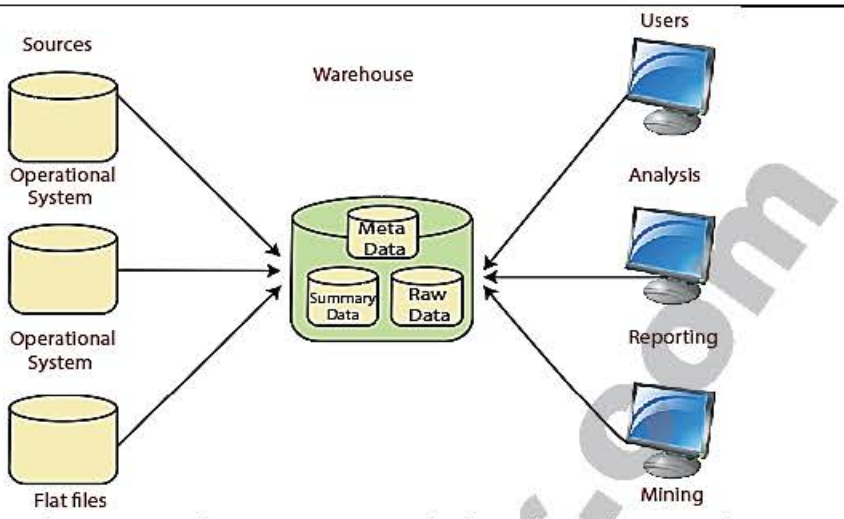
Data Warehouse applications are designed to support the user ad-hoc data requirements, an activity recently dubbed online analytical processing (OLAP). These include applications such as forecasting, profiling, summary reporting, and trend analysis.

Production databases are updated continuously by either by hand or via OLTP applications. In contrast, a warehouse database is updated from operational systems periodically, usually during off-hours. As OLTP data accumulates in production databases, it is regularly extracted, filtered, and then loaded into a dedicated warehouse server that is accessible to users. As the warehouse is populated, it must be restructured tables de-normalized, data cleansed of errors and redundancies and new fields and keys added to reflect the needs to the user for sorting, combining, and summarizing data.

Data warehouses and their architectures very depending upon the elements of an organization's situation.

Three common architectures are:

1. **Data Warehouse Architecture: Basic Architecture of a Data Warehouse**



Operational System: An operational system is a method used in data warehousing to refer to a **system** that is used to process the day-to-day transactions of an organization.

Flat Files: A Flat file system is a system of files in which transactional data is stored, and every file in the system must have a different name.

Meta Data: A set of data that defines and gives information about other data.

- Meta Data used in Data Warehouse for a variety of purpose, including:
- Meta Data summarizes necessary information about data, which can make finding and work with particular instances of data more accessible. For example, author, data build, and data changed, and file size are examples of very basic document metadata.
- Metadata is used to direct a query to the most appropriate data source.

Lightly and highly summarized data: The area of the data warehouse saves all the predefined lightly and highly summarized (aggregated) data generated by the warehouse manager.

The goals of the summarized information are to speed up query performance. The summarized record is updated continuously as new information is loaded into the warehouse.

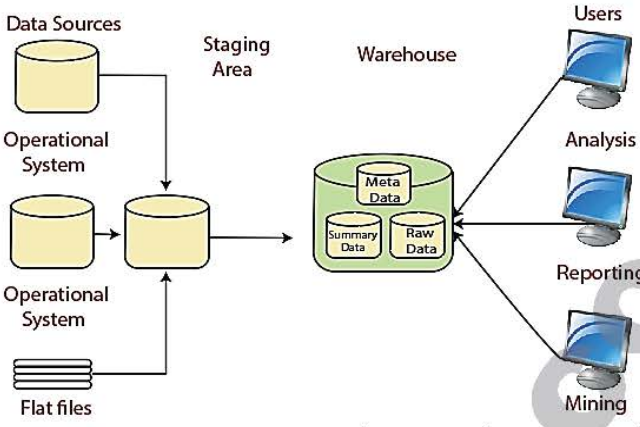
End-User access Tools: The principal purpose of a data warehouse is to provide information to the business managers for strategic decision-making. These customers interact with the warehouse using end-client access tools.

The examples of some of the end-user access tools can be:

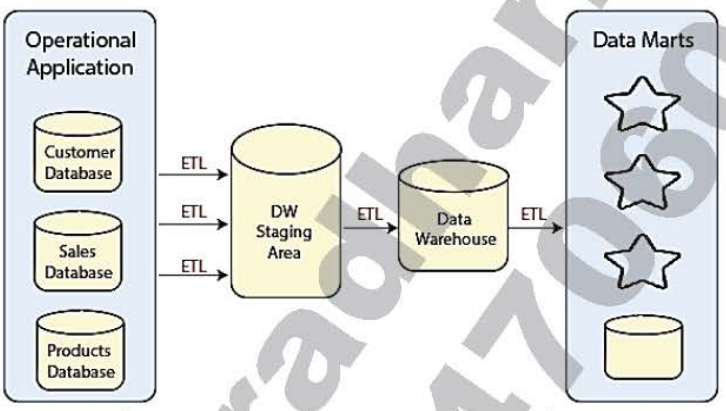
- Reporting and Query Tools
- Application Development Tools
- Executive Information Systems Tools
- Online Analytical Processing Tools
- Data Mining Tools

2 **Data Warehouse Architecture: With Staging Area:** We must clean and process your operational information before put it into the warehouse.We can do this programmatically, although data warehouses uses a staging area (A place where data is processed before entering the warehouse). A staging area simplifies data cleansing and consolidation for operational method coming from multiple source systems, especially for enterprise data warehouses where all relevant data of an enterprise is consolidated.

Architecture of a Data Warehouse with a Staging Area

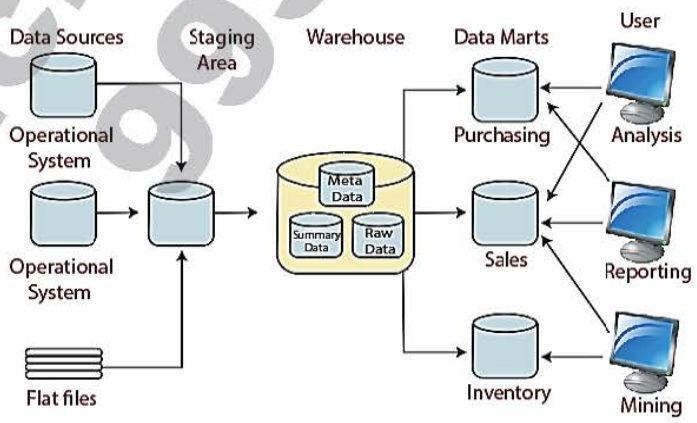


3. **Data Warehouse Staging Area** is a temporary location where a record from source systems is copied.



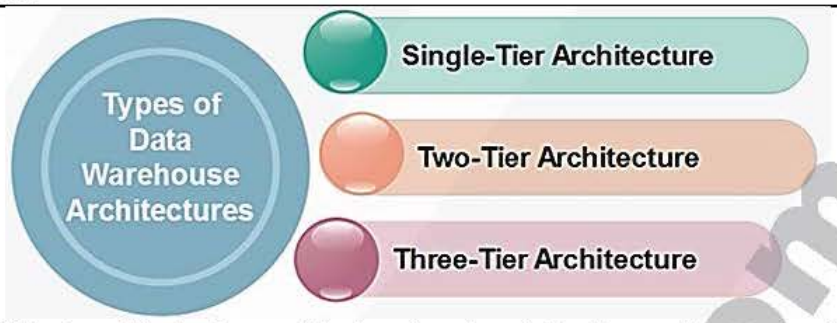
4. **Data Warehouse Architecture: With Staging Area and Data Marts:** We may want to customize our warehouse's architecture for multiple groups within our organization. We can do this by adding data marts. A data mart is a segment of a data warehouses that can provided information for reporting and analysis on a section, unit, department or operation in the company, e.g., sales, payroll, production, etc.
The figure illustrates an example where purchasing, sales, and stocks are separated. In this example, a financial analyst wants to analyze historical data for purchases and sales or mine historical information to make predictions about customer behavior.

Architecture of a Data Warehouse with a Staging Area and Data Marts



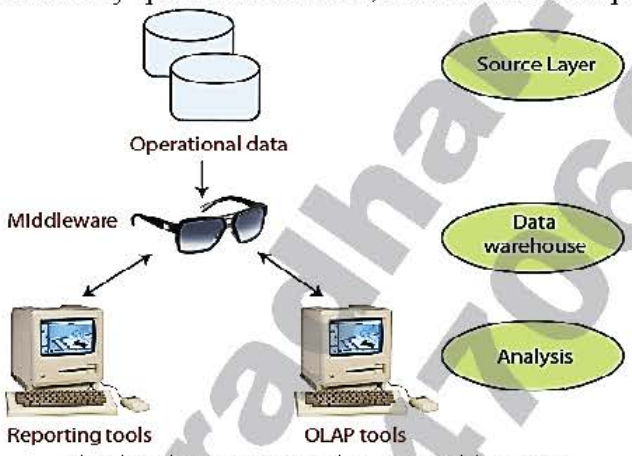
Q. Mention the types of data warehouse Architecture?

Ans. There is mainly three types of data warehouse architecture:



Single-Tier Architecture: Single-Tier architecture is not periodically used in practice. Its purpose is to minimize the amount of data stored to reach this goal; it removes data redundancies.

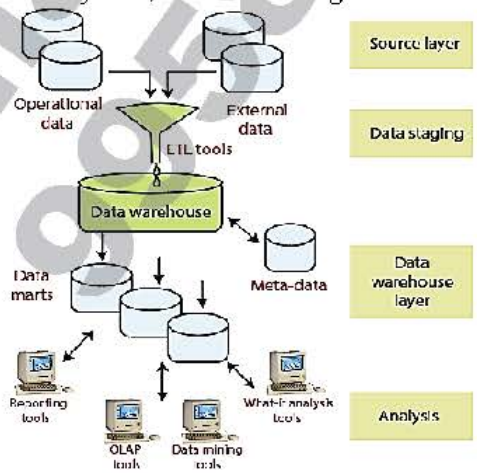
The figure shows the only layer physically available is the source layer. In this method, data warehouses are virtual. This means that the data warehouse is implemented as a multidimensional view of operational data created by specific middleware, or an intermediate processing layer.



Single-Tier Data Warehouse Architecture

The vulnerability of this architecture lies in its failure to meet the requirement for separation between analytical and transactional processing. Analysis queries are agreed to operational data after the middleware interprets them. In this way, queries affect transactional workloads.

Two-Tier Architecture: The requirement for separation plays an essential role in defining the two-tier architecture for a data warehouse system, as shown in fig:



Two-Tier Data Warehouse Architecture

Although it is typically called two-layer architecture to highlight a separation between physically available sources and data warehouses, in fact, consists of four subsequent data flow stages:

Source layer: A data warehouse system uses a heterogeneous source of data. That data is stored initially to corporate relational databases or legacy databases, or it may come from an information system outside the corporate walls.

Data Staging: The data stored to the source should be extracted, cleansed to remove inconsistencies and fill gaps, and integrated to merge heterogeneous sources into one standard schema. The so-named Extraction, Transformation, and Loading Tools (ETL) can combine heterogeneous schemata, extract, transform, cleanse, validate, filter, and load source data into a data warehouse.

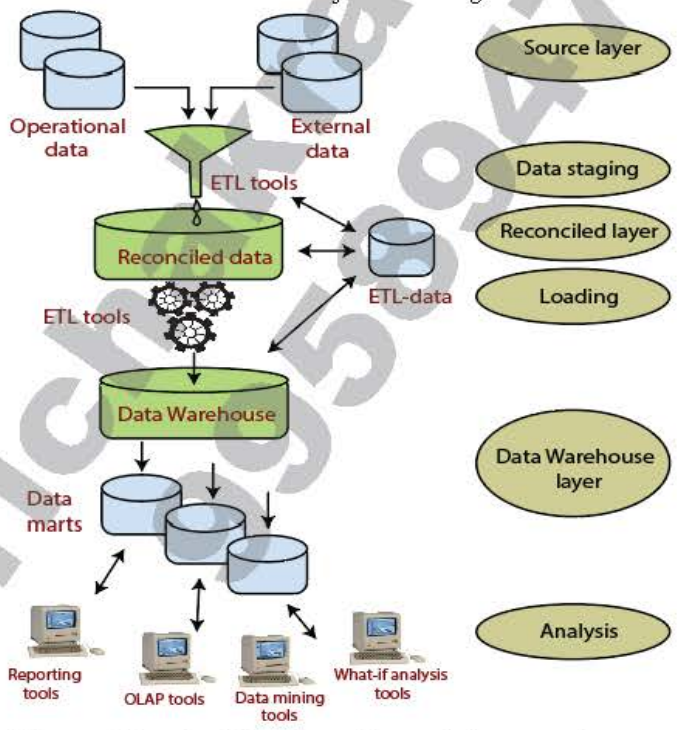
Data Warehouse layer: Information is saved to one logically centralized individual repository: a data warehouse. The data warehouses can be directly accessed, but it can also be used as a source for creating data marts, which partially replicate data warehouse contents and are designed for specific enterprise departments. Meta-data repositories store information on sources, access procedures, data staging, users, data mart schema, and so on.

Analysis: In this layer, integrated data is efficiently, and flexible accessed to issue reports, dynamically analyze information, and simulate hypothetical business scenarios. It should feature aggregate information navigators, complex query optimizers, and customer-friendly GUIs.

Three-Tier Architecture: The three-tier architecture consists of the source layer (containing multiple source system), the reconciled layer and the data warehouse layer (containing both data warehouses and data marts). The reconciled layer sits between the source data and data warehouse.

The main advantage of the reconciled layer is that it creates a standard reference data model for a whole enterprise. At the same time, it separates the problems of source data extraction and integration from those of data warehouse population. In some cases, the reconciled layer is also directly used to accomplish better some operational tasks, such as producing daily reports that cannot be satisfactorily prepared using the corporate applications or generating data flows to feed external processes periodically to benefit from cleaning and integration.

This architecture is especially useful for the extensive, enterprise-wide systems. A disadvantage of this structure is the extra file storage space used through the extra redundant reconciled layer. It also makes the analytical tools a little further away from being real-time.



Three-Tier Architecture for a data warehouse system

MCS-221: Data Warehousing and Data Mining
Guess Paper-2

Q. What is data mart and mention its Advantages?

Ans. A data mart is a simple form of data warehouse focused on a single subject or line of business. With a data mart, teams can access data and gain insights faster, because they don't have to spend time searching within a more complex data warehouse or manually aggregating data from different sources.

The benefits of a data mart: A data mart dedicated to a team or specific line of business offers several benefits:

1. **A single source of truth:** The centralized nature of a data mart helps ensure that everyone in a department or organization makes decisions based on the same data. This is a major benefit, because the data and the predictions based on that data can be trusted, and stakeholders can focus on making decisions and taking action, as opposed to arguing about the data itself
2. **Quicker access to data:** Specific business teams and users can rapidly access the subset of data they need from the enterprise data warehouse and combine it with data from various other sources. Once the connections to their desired data sources are established, they can get live data from a data mart whenever needed without having to go to IT to obtain periodic extracts. Business and IT teams both gain improved productivity as a result
3. **Faster insights leading to faster decision making:** While a data warehouse enables enterprise-level decision-making, a data mart allows data analytics at the department level. Analysts can focus on specific challenges and opportunities in areas such as finance and HR and move more rapidly from data to insights, which enables them to make better and faster decisions
4. **Simpler and faster implementation:** Setting up an enterprise data warehouse to cater to the needs of your entire organization can require significant time and effort. A data mart, in contrast, is focused on serving the needs of specific business teams, requiring access to fewer data sets. It therefore is much simpler and faster to implement
5. **Creating agile and scalable data management:** Data marts provide an agile data management system that works in tandem with business needs, including being able to use information gathered in past projects to help with current tasks. Teams can update and change their data mart based on new and evolving analytics project
6. **Transient analysis:** Some data analytics projects are short-lived—for example, completing a specific analysis of online sales for a two-week promotion prior to a team meeting. Teams can rapidly set up a data mart to accomplish such a project.

Q. What is dimensional Modelling, mention its Advantages also?

Ans. Dimensional modeling represents data with a cube operation, making more suitable logical data representation with OLAP data management. The perception of Dimensional Modeling was developed by Ralph Kimball and is consist of "fact" and "dimension" tables.

In dimensional modeling, the transaction record is divided into either "facts," which are frequently numerical transaction data, or "dimensions," which are the reference information that gives context to the facts. For example, a sale transaction can be damage into facts such as the number of products ordered and the price paid for the products, and into dimensions such as order date, user name, product number, order ship-to, and bill-to locations, and salesman responsible for receiving the order.

Objectives of Dimensional Modeling: The purposes of dimensional modeling are:

- To produce database architecture that is easy for end-clients to understand and write queries.

- To maximize the efficiency of queries. It achieves these goals by minimizing the number of tables and relationships between them.

Advantages of Dimensional Modeling: Following are the benefits of dimensional modeling are:

1. **Dimensional modeling is simple:** Dimensional modeling methods make it possible for warehouse designers to create database schemas that business customers can easily hold and comprehend. There is no need for vast training on how to read diagrams, and there is no complicated relationship between different data elements.
2. **Dimensional modeling promotes data quality:** The star schema enable warehouse administrators to enforce referential integrity checks on the data warehouse. Since the fact information key is a concatenation of the essentials of its associated dimensions, a factual record is actively loaded if the corresponding dimensions records are duly described and also exist in the database.

By enforcing foreign key constraints as a form of referential integrity check, data warehouse DBAs add a line of defence against corrupted warehouses data.

3. **Performance optimization is possible through aggregates:** As the size of the data warehouse increases, performance optimization develops into a pressing concern. Customers who have to wait for hours to get a response to a query will quickly become discouraged with the warehouses. Aggregates are one of the easiest methods by which query performance can be optimized.

Disadvantages of Dimensional Modelling

- To maintain the integrity of fact and dimensions, loading the data warehouses with a record from various operational systems is complicated.
- It is severe to modify the data warehouse operation if the organization adopting the dimensional technique changes the method in which it does business.

Elements of Dimensional Modeling

Fact: It is a collection of associated data items, consisting of measures and context data. It typically represents business items or business transactions.

Dimensions: It is a collection of data which describe one business dimension. Dimensions decide the contextual background for the facts, and they are the framework over which OLAP is performed.

Measure: It is a numeric attribute of a fact, representing the performance or behaviour of the business relative to the dimensions.

Considering the relational context, there are two basic models which are used in dimensional modeling:

- Star Model
- Snowflake Model

The star model is the underlying structure for a dimensional model. It has one broad central table (fact table) and a set of smaller tables (dimensions) arranged in a radial design around the primary table. The snowflake model is the conclusion of decomposing one or more of the dimensions.

Fact Table: Fact tables are used to data facts or measures in the business. Facts are the numeric data elements that are of interest to the company.

Characteristics of the Fact table:

- The fact table includes numerical values of what we measure. For example, a fact value of 20 might means that 20 widgets have been sold.
- Each fact table includes the keys to associated dimension tables. These are known as foreign keys in the fact table.
- Fact tables typically include a small number of columns.

When it is compared to dimension tables, fact tables have a large number of rows.

Dimension Table: Dimension tables establish the context of the facts. Dimensional tables store fields that describe the facts.

Characteristics of the Dimension table: Dimension tables contain the details about the facts. That, as an example, enables the business analysts to understand the data and their reports better.The

dimension tables include descriptive data about the numerical values in the fact table. That is, they contain the attributes of the facts.

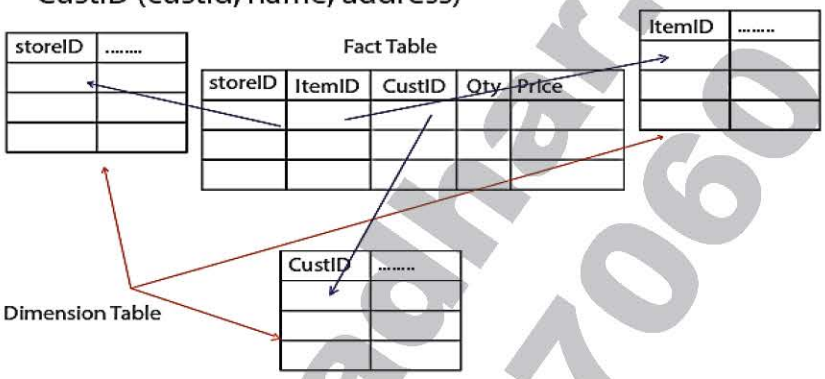
For example, the dimension tables for a marketing analysis function might include attributes such as time, marketing region, and product type.

Since the record in a dimension table is demoralized, it usually has a large number of columns. The dimension tables include significantly fewer rows of information than the fact table.

The attributes in a dimension table are used as row and column headings in a document or query results display.

Example: A city and state can view a store summary in a fact table. Item summary can be viewed by brand, color, etc. Customer information can be viewed by name and address.

Sales (StoreID, ItemID, CustID, qty, price)
StoreID (storeid, city, state)
ItemID (itemid, category, brand, color, size)
CustID (custid, name, address)



Fact Table

Time ID	Product ID	Customer ID	Unit Sold
4	17	2	1
8	21	3	2
8	4	1	1

In this example, Customer ID column in the facts table is the foreign keys that join with the dimension table. By following the links, we can see that row 2 of the fact table records the fact that customer 3, Gaurav, bought two items on day 8.

Dimension Tables

Customer ID	Name	Gender	Income	Education	Region
1	Rohan	Male	2	3	4
2	Sandeep	Male	3	5	1
3	Gaurav	Male	1	7	3

Hierarchy: A hierarchy is a directed tree whose nodes are dimensional attributes and whose arcs model many to one association between dimensional attributes team. It contains a dimension, positioned at the tree's root, and all of the dimensional attributes that define it.

Q. How Snowflake schema is different from Star schema explain in detail?

Ans. A snowflake schema is equivalent to the star schema. "A schema is known as a snowflake if one or more dimension tables do not connect directly to the fact table but must join through other dimension tables."

The snowflake schema is an expansion of the star schema where each point of the star explodes into more points. It is called snowflake schema because the diagram of snowflake schema resembles a snowflake. Snowflaking is a method of normalizing the dimension tables in STAR schemas. When we normalize all the dimension tables entirely, the resultant structure resembles a snowflake with the fact table in the middle. Snowflaking is used to develop the performance of specific queries.

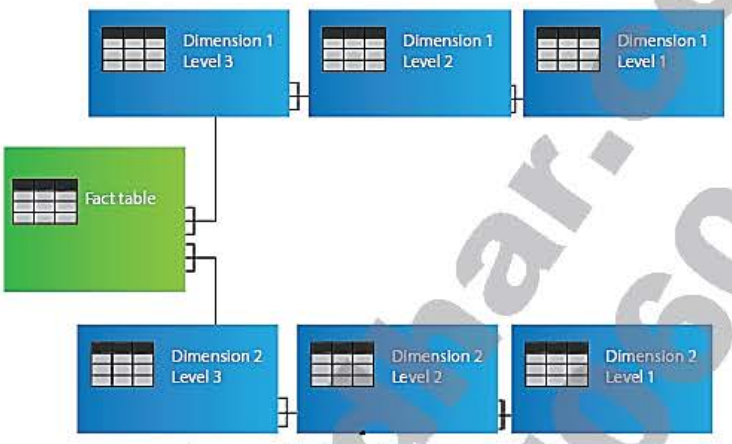
The schema is diagrammed with each fact surrounded by its associated dimensions, and those dimensions are related to other dimensions, branching out into a snowflake pattern.

The snowflake schema consists of one fact table which is linked to many dimension tables, which can be linked to other dimension tables through a many-to-one relationship.

Tables in a snowflake schema are generally normalized to the third normal form. Each dimension table performs exactly one level in a hierarchy.

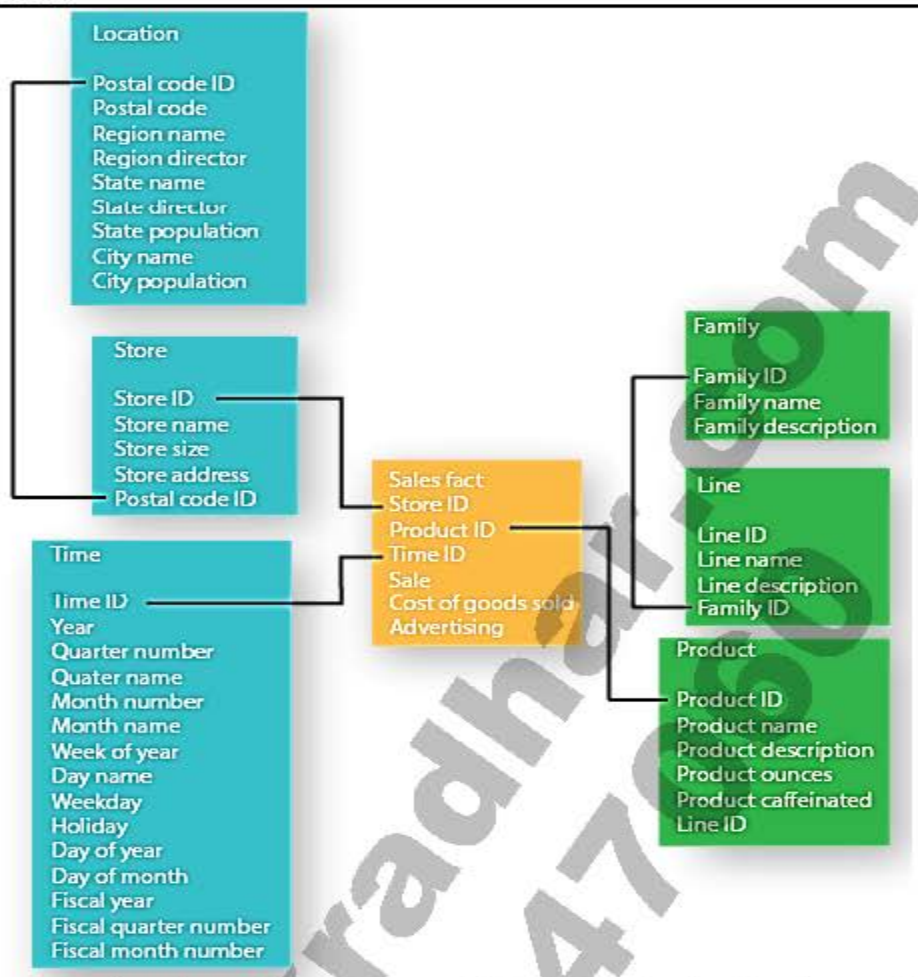
The following diagram shows a snowflake schema with two dimensions, each having three levels.

Snowflake schemas can have any number of dimensions, and each dimension can have any number of levels.



Snowflake Schema

Example: Figure shows a snowflake schema with a Sales fact table, with Store, Location, Time, Product, Line, and Family dimension tables. The Market dimension has two dimension tables with Store as the primary dimension table, and Location as the outrigger dimension table. The product dimension has three dimension tables with Product as the primary dimension table, and the Line and Family table are the outrigger dimension tables.



A star schema store all attributes for a dimension into one demoralized table. This needed more disk space than a more normalized snowflake schema. Snow flaking normalizes the dimension by moving attributes with low cardinality into separate dimension tables that relate to the core dimension table by using foreign keys. Snow flaking for the sole purpose of minimizing disk space is not recommended, because it can adversely impact query performance.

In snowflake, schema tables are normalized to delete redundancy. In snowflake dimension tables are damaged into multiple dimension tables.

A snowflake schema is designed for flexible querying across more complex dimensions and relationship. It is suitable for many to many and one to many relationships between dimension levels.

Advantage of Snowflake Schema:

- The primary advantage of the snowflake schema is the development in query performance due to minimized disk storage requirements and joining smaller lookup tables.
- It provides greater scalability in the interrelationship between dimension levels and components.
- No redundancy, so it is easier to maintain.

Disadvantage of Snowflake Schema

- The primary disadvantage of the snowflake schema is the additional maintenance efforts required due to the increasing number of lookup tables. It is also known as a multi fact star schema.
- There are more complex queries and hence, difficult to understand.
- More tables more join so more query execution time.

Following is a key difference between Snowflake schema vs Star schema:

Star Schema	Snowflake Schema

Hierarchies for the dimensions are stored in the dimensional table.	Hierarchies are divided into separate tables.
It contains a fact table surrounded by dimension tables.	One fact table surrounded by dimension table which are in turn surrounded by dimension table
In a star schema, only single join creates the relationship between the fact table and any dimension tables.	A snowflake schema requires many joins to fetch the data.
Simple DB Design.	Very Complex DB Design.
Denormalized Data structure and query also run faster.	Normalized Data Structure.
High level of Data redundancy	Very low-level data redundancy
Single Dimension table contains aggregated data.	Data Split into different Dimension Tables.
Cube processing is faster.	Cube processing might be slow because of the complex join.
Offers higher performing queries using Star Join Query Optimization. Tables may be connected with multiple dimensions.	The Snowflake schema is represented by centralized fact table which unlikely connected with multiple dimensions.

Q. Describe fact Constellation Schema?

Ans. Fact Constellation is a schema for representing multidimensional model. It is a collection of multiple fact tables having some common dimension tables. It can be viewed as a collection of several star schemas and hence, also known as *Galaxy schema*. It is one of the widely used schema for Data warehouse designing and it is much more complex than star and snowflake schema. For complex systems, we require fact constellations.

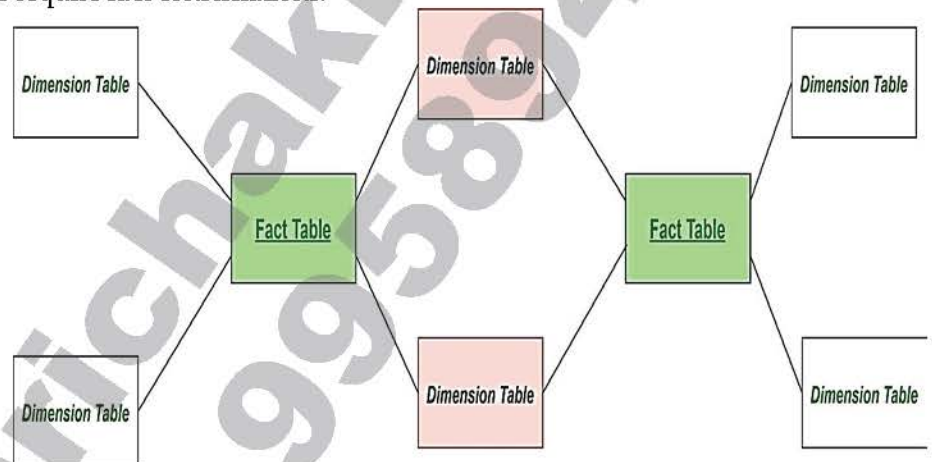
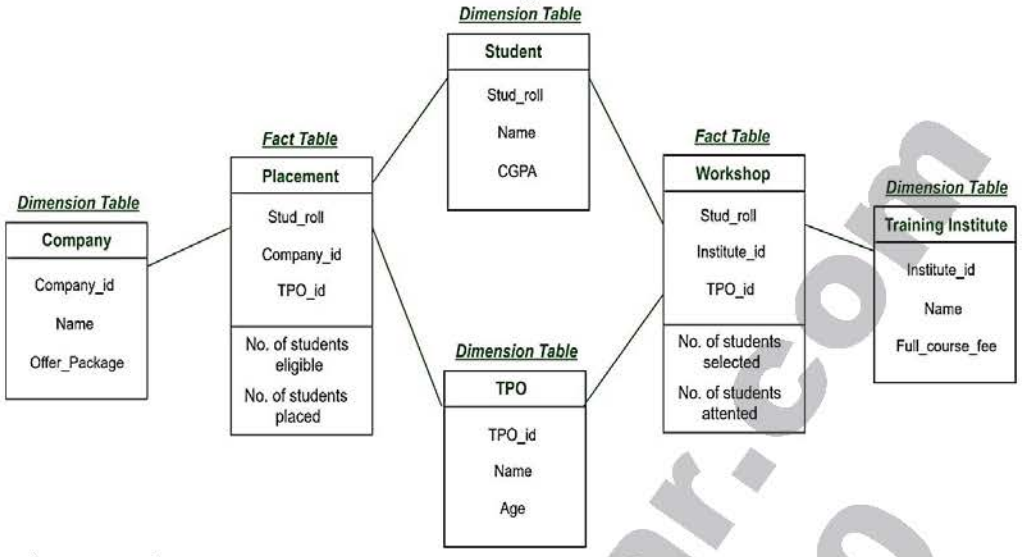


Figure – General structure of Fact Constellation

Here, the pink coloured Dimension tables are the common ones among both the star schemas. Green coloured fact tables are the fact tables of their respective star schemas.

Example:



In above demonstration:

- **Placement** is a *fact table* having attributes: (Stud_roll, Company_id, TPO_id) with facts: (Number of students eligible, Number of students placed).
- **Workshop** is a *fact table* having attributes: (Stud_roll, Institute_id, TPO_id) with facts: (Number of students selected, Number of students attended the workshop).
- **Company** is a *dimension table* having attributes: (Company_id, Name, Offer_package).
- **Student** is a *dimension table* having attributes: (Student_roll, Name, CGPA).
- **TPO** is a *dimension table* having attributes: (TPO_id, Name, Age).

Training Institute is a *dimension table* having attributes: (Institute_id, Name, Full_course_fee).

So, there are two fact tables namely, Placement and Workshop which are part of two different star schemas having dimension tables – *Company*, *Student* and *TPO* in Star schema with fact table *Placement* and dimension tables – *Training Institute*, *Student* and *TPO* in Star schema with fact table *Workshop*. Both the star schema have two dimension tables common and hence, forming a fact constellation or galaxy schema.

Advantage: Provides a flexible schema.

Disadvantage: It is much more complex and hence, hard to implement and maintain.

Q. What is ETL, briefly describe the process of ETL?

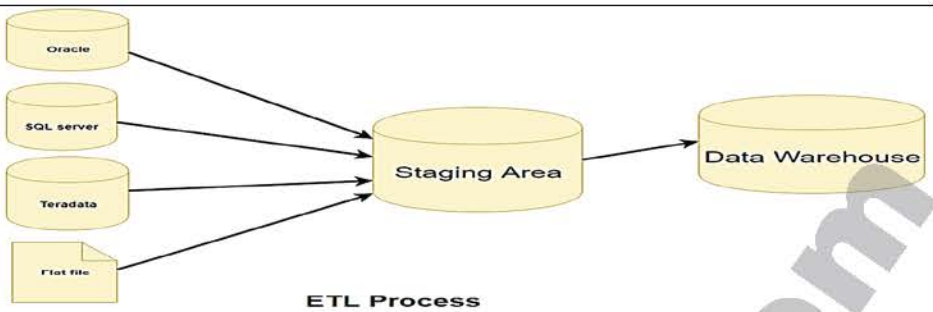
Ans. ETL is a process that extracts the data from different source systems, then transforms the data (like applying calculations, concatenations, etc.) and finally loads the data into the Data Warehouse system. Full form of ETL is Extract, Transform and Load.

It's tempting to think a creating a Data warehouse is simply extracting data from multiple sources and loading into database of a Data warehouse. This is far from the truth and requires a complex ETL process.

The ETL process requires active inputs from various stakeholders including developers, analysts, testers, top executives and is technically challenging.

In order to maintain its value as a tool for decision-makers, Data warehouse system needs to change with business changes. ETL is a recurring activity (daily, weekly, monthly) of a Data warehouse system and needs to be agile, automated, and well documented.

ETL Process in Data Warehouses: ETL is a 3-step process



ETL Process

1. **Step 1) Extraction:** In this step of ETL architecture, data is extracted from the source system into the staging area. Transformations if any are done in staging area so that performance of source system in not degraded. Also, if corrupted data is copied directly from the source into Data warehouse database, rollback will be a challenge. Staging area gives an opportunity to validate extracted data before it moves into the Data warehouse.

Data warehouse needs to integrate systems that have different: DBMS, Hardware, Operating Systems and Communication Protocols. Sources could include legacy applications like Mainframes, customized applications, Point of contact devices like ATM, Call switches, text files, spreadsheets, ERP, data from vendors, partners amongst others.

Hence one need a logical data map before data is extracted and loaded physically. This data map describes the relationship between sources and target data.

Three Data Extraction methods:

- Full Extraction
- Partial Extraction- without update notification.
- Partial Extraction- with update notification

Irrespective of the method used, extraction should not affect performance and response time of the source systems. These source systems are live production databases. Any slow down or locking could affect company's bottom line.

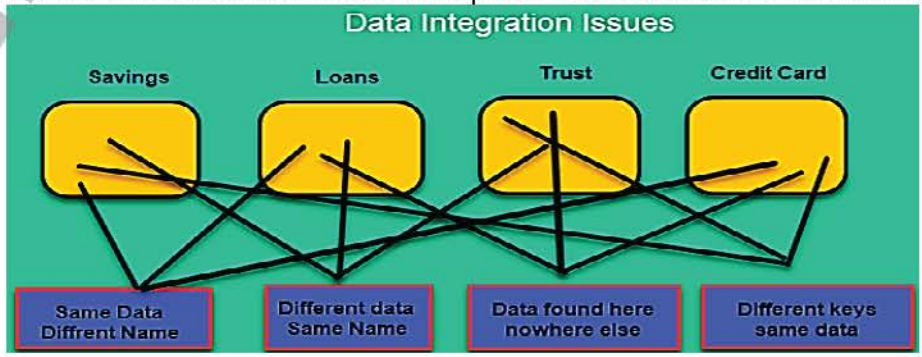
Some validations are done during Extraction:

- Reconcile records with the source data
- Make sure that no spam/unwanted data loaded
- Data type check
- Remove all types of duplicate/fragmented data
- Check whether all the keys are in place or not

2. **Step 2) Transformation:** Data extracted from source server is raw and not usable in its original form. Therefore it needs to be cleansed, mapped and transformed. In fact, this is the key step where ETL process adds value and changes data such that insightful BI reports can be generated.

   It is one of the important ETL concepts where you apply a set of functions on extracted data. Data that does not require any transformation is called as direct move or pass through data.

   In transformation step, you can perform customized operations on data. For instance, if the user wants sum-of-sales revenue which is not in the database. Or if the first name and the last name in a table is in different columns. It is possible to concatenate them before loading.



Following are Data Integrity Problems:

- Different spelling of the same person like Jon, John, etc.
- There are multiple ways to denote company name like Google, Google Inc.
- Use of different names like Cleveland, Cleveland.
- There may be a case that different account numbers are generated by various applications for the same customer.
- In some data required files remains blank
- Invalid product collected at POS as manual entry can lead to mistakes.

Validations are done during this stage

- Filtering – Select only certain columns to load
- Using rules and lookup tables for Data standardization
- Character Set Conversion and encoding handling
- Conversion of Units of Measurements likes Date Time Conversion, currency conversions, numerical conversions, etc.
- Data threshold validation check. For example, age cannot be more than two digits.
- Data flow validation from the staging area to the intermediate tables.
- Required fields should not be left blank.
- Cleaning (for example, mapping NULL to 0 or Gender Male to "M" and Female to "F" etc.)
- Split a column into multiples and merging multiple columns into a single column.
- Transposing rows and columns,
- Use lookups to merge data
- Using any complex data validation (e.g., if the first two columns in a row are empty then it automatically reject the row from processing)

Step 3) Loading: Loading data into the target data warehouse database is the last step of the ETL process. In a typical Data warehouse, huge volume of data needs to be loaded in a relatively short period (nights). Hence, load process should be optimized for performance.

In case of load failure, recover mechanisms should be configured to restart from the point of failure without data integrity loss. Data Warehouse admins need to monitor, resume, cancel loads as per prevailing server performance.

Types of Loading:

- **Initial Load** — populating all the Data Warehouse tables
- **Incremental Load** — applying ongoing changes as when needed periodically.
- **Full Refresh** —erasing the contents of one or more tables and reloading with fresh data.

Load verification:

- Ensure that the key field data is neither missing nor null.
- Test modeling views based on the target tables.
- Check that combined values and calculated measures.
- Data checks in dimension table as well as history table.
- Check the BI reports on the loaded fact and dimension table.

Q. Explain Physical and Logical Extraction in detail?

Ans. Extraction is the operation of extracting data from a source system for further use in a data warehouse environment. This is the first step of the ETL process. After the extraction, this data can be transformed and loaded into the data warehouse.

The source systems for a data warehouse are typically transaction processing applications. For example, one of the source systems for a sales analysis data warehouse might be an order entry system that records all of the current order activities.

Designing and creating the extraction process is often one of the most time-consuming tasks in the ETL process and, indeed, inthe entire data warehousing process. The source systems might be very complex and poorly documented, and thus determining which data needs to be extracted can be difficult. The data has to be extracted normally not only once, but several times in a periodic manner

to supply all changed data to the warehouse and keep it up-to-date. Moreover, the source system typically cannot be modified, nor can its performance or availability be adjusted, to accommodate the needs of the data warehouse extraction process.

The estimated amount of the data to be extracted and the stage in the ETL process (initial load or maintenance of data) may also impact the decision of how to extract, from a logical and a physical perspective. Basically, there are two methods to extract data that is logically and physically.

1. **Logical Extraction Methods:** There are two kinds of logical extraction:

- **Full Extraction:** The data is extracted completely from the source system. Since this extraction reflects all the data currently available on the source system, there's no need to keep track of changes to the data source since the last successful extraction. The source data will be provided as-is and no additional logical information (for example, timestamps) is necessary on the source site. An example for a full extraction may be an export file of a distinct table or a remote SQL statement scanning the complete source table.
- **Incremental Extraction:** At a specific point in time, only the data that has changed since a well-defined event back in history will be extracted. This event may be the last time of extraction or a more complex business event like the last booking day of a fiscal period. To identify this delta change there must be a possibility to identify all the changed information since this specific time event. This information can be either provided by the source data itself like an application column, reflecting the last-changed timestamp or a change table where an appropriate additional mechanism keeps track of the changes besides the originating transactions. In most cases, using the latter method means adding extraction logic to the source system.

Many data warehouses do not use any change-capture techniques as part of the extraction process. Instead, entire tables from the source systems are extracted to the data warehouse or staging area, and these tables are compared with a previous extract from the source system to identify the changed data. This approach may not have significant impact on the source systems, but it clearly can place a considerable burden on the data warehouse processes, particularly if the data volumes are large.

2. **Physical Extraction Methods:** Depending on the chosen logical extraction method and the capabilities and restrictions on the source side, the extracted data can be physically extracted by two mechanisms. The data can either be extracted online from the source system or from an offline structure. Such an offline structure might already exist or it might be generated by an extraction routine.

There are the following methods of physical extraction:

- **Online Extraction:** The data is extracted directly from the source system itself. The extraction process can connect directly to the source system to access the source tables themselves or to an intermediate system that stores the data in a preconfigured manner (for example, snapshot logs or change tables). Note that the intermediate system is not necessarily physically different from the source system.
- With online extractions, you need to consider whether the distributed transactions are using original source objects or prepared source objects.
- **Offline Extraction:** The data is not extracted directly from the source system but is staged explicitly outside the original source system. The data already has an existing structure (for example, redo logs, archive logs or transportable tablespaces) or was created by an extraction routine.

Consider the following structures:

- Flat files
- Data in a defined, generic format. Additional information about the source object is necessary for further processing.
- Dump files
- Oracle-specific format. Information about the containing objects is included.
- Redo and archive logs
- Information is in a special, additional dump file.

- Transportable tablespaces.

Q. Describe data Loading in ETL?

Ans. Data loading refers to the "load" component of ETL. After data is retrieved and combined from multiple sources (extracted), cleaned and formatted (transformed), it is then loaded into a storage system, such as a cloud data warehouse.

ETL aids in the data integration process that standardizes diverse and disparate data types to make it available for querying, manipulation, or reporting for many different individuals and teams. Because today's organizations are increasingly reliant upon their own data to make smarter, faster business decisions, ETL needs to be scalable and streamlined to provide the most benefit.

Benefits of data loading:

- Before ETL evolved into its current state, organizations had to load data manually or else use several different ETL vendors for each different database or source. Understandably, this made the process slower and more complicated than it needed to be — reinforcing data silos rather than breaking them down.
- Today, the ETL process — including data loading — is designed for speed, efficiency, and flexibility. But more importantly, it can scale to meet the growing data demands of most enterprises. ETL easily accommodates proliferation of data sources as technologies like IoT and connected devices continue to gain popularity. And it can handle any number of data types and formats, whether structured, semi-structured, or unstructured.

Challenges with data loading: Many ETL solutions are cloud-based, which accounts for their speed and scalability. But large enterprises with traditional, on-premise infrastructure and data management processes often use custom built scripts to collect and load their own data into storage systems through customized configurations. This can:

- Slow down analysis. Each time a data source is added or changed, the system has to be reconfigured, which takes time and hampers the ability to make quick decisions.
- Increase the likelihood of errors. Changes and reconfigurations open up the door for human error, duplicate or missing data, and other problems.
- Require specialized knowledge. In-house IT teams often lack the skill (and bandwidth) needed to code and monitor ETL functions themselves.
- Require costly equipment. In addition to investment in the right human resources, organizations have to purchase, house, and maintain hardware and other equipment to run the process on site.

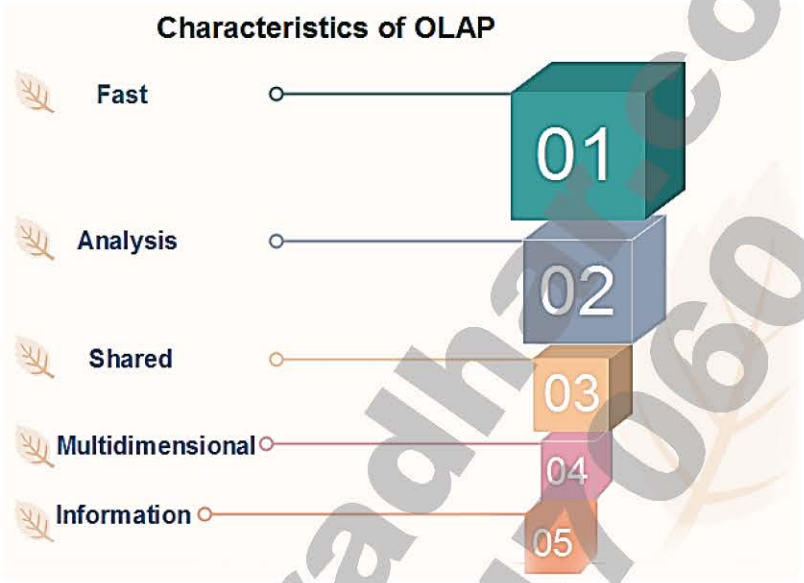
Methods for data loading: Since data loading is part of the larger ETL process, organizations need a proper understanding of the types of ETL tools and methods available, and which one(s) work best for their needs, budget, and structure.

- **Cloud-based:** ETL tools in the cloud are built for speed and scalability, and often enable real-time data processing. They also include the ready-made infrastructure and expertise of the vendor, who can advise on best practices for each organization's unique setup and needs.
- **Batch processing:** ETL tools that work off batch processing move data at the same scheduled time every day or week. It works best for large volumes of data and for organizations that don't necessarily need real-time access to their data.
- **Open source:** Many open-source ETL tools are quite cost-effective as their code base is publicly accessible, modifiable, and shareable. While a good alternative to commercial solutions, these tools can still require some customization or hand-coding.

MCS-221: Data Warehousing and Data Mining
Guess Paper-1

Q. What are the characteristics of OLAP?

Ans. In the FASMI characteristics of OLAP methods, the term derived from the first letters of the characteristics are:



Fast: It defines which the system targeted to deliver the most feedback to the client within about five seconds, with the elementary analysis taking no more than one second and very few taking more than 20 seconds.

Analysis: It defines which the method can cope with any business logic and statistical analysis that is relevant for the function and the user, keep it easy enough for the target client. Although some preprogramming may be needed we do not think it acceptable if all application definitions have to be allow the user to define new Adhoc calculations as part of the analysis and to document on the data in any desired method, without having to program so we excludes products (like Oracle Discoverer) that do not allow the user to define new Adhoc calculation as part of the analysis and to document on the data in any desired product that do not allow adequate end user-oriented calculation flexibility.

Share: It defines which the system tools all the security requirements for understanding and, if multiple write connection is needed, concurrent update location at an appropriated level, not all functions need customer to write data back, but for the increasing number which does, the system should be able to manage multiple updates in a timely, secure manner.

Multidimensional: This is the basic requirement. OLAP system must provide a multidimensional conceptual view of the data, including full support for hierarchies, as this is certainly the most logical method to analyze business and organizations.

Information: The system should be able to hold all the data needed by the applications. Data sparsity should be handled in an efficient manner.

The main characteristics of OLAP are as follows:-

- **Multidimensional conceptual view:** OLAP systems let business users have a dimensional and logical view of the data in the data warehouse. It helps in carrying slice and dice operations.

- **Multi-User Support:** Since the OLAP techniques are shared, the OLAP operation should provide normal database operations, containing retrieval, update, adequacy control, integrity, and security.
- **Accessibility:** OLAP acts as a mediator between data warehouses and front-end. The OLAP operations should be sitting between data sources (e.g., data warehouses) and an OLAP front-end.
- **Storing OLAP results:** OLAP results are kept separate from data sources.
- Uniform documenting performance: Increasing the number of dimensions or database size should not significantly degrade the reporting performance of the OLAP system.
- OLAP provides for distinguishing between zero values and missing values so that aggregates are computed correctly.
- OLAP system should ignore all missing values and compute correct aggregate values.
- OLAP facilitate interactive query and complex analysis for the users.
- OLAP allows users to drill down for greater details or roll up for aggregations of metrics along a single business dimension or across multiple dimension.
- OLAP provides the ability to perform intricate calculations and comparisons.
- OLAP presents results in a number of meaningful ways, including charts and graphs.

Q. Mention the OLAP Functions?

Ans. The OLAP functions extend the syntax of the SQL analytic functions. This syntax is familiar to SQL developers and DBAs, so you can adopt it more easily than proprietary OLAP languages and APIs. Using the OLAP functions, you can create all standard calculated measures, including rank, share, prior and future periods, period-to-date, parallel period, moving aggregates, and cumulative aggregates.

The OLAP functions are grouped into these categories:

- Aggregate Functions
- Analytic Functions
- Hierarchical Functions
- Lag Functions
- OLAP DML Functions
- Rank Functions
- Share Functions
- Window Functions

Aggregate Functions:

- AVERAGE_RANK
  AVG
  COUNT
  DENSE_RANK
  MAX
  MIN
  RANK
  SUM

Analytic Functions:

- AVERAGE_RANK
  AVG
  COUNT
  DENSE_RANK
  LAG
  LAG_VARIANCE
  LEAD_VARIANCE_PERCENT

MAX
MIN
RANK
ROW_NUMBER
SUM

Hierarchical Functions:

- HIER_ANCESTOR
  HIER_CHILD_COUNT
  HIER_DEPTH
  HIER_LEVEL
  HIER_ORDER
  HIER_PARENT
  HIER_TOP

Lag Functions:

- LAG
  LAG_VARIANCE
  LAG_VARIANCE_PERCENT
  LEAD
  LEAD_VARIANCE
  LEAD_VARIANCE_PERCENT

OLAP DML Functions:

OLAP_DML_EXPRESSION

Rank Functions:

- AVERAGE_RANK
  DENSE_RANK
  RANK
  ROW_NUMBER

Share Functions:

- SHARE

Window Functions:

- AVG
  COUNT
  MAX
  MIN
  SUM

Q. Mention the types of data marts?

Ans.A *data mart* is a simple form of a data warehouse that is focused on a single subject (or functional area), such as Sales or Finance or Marketing.

Data marts are often built and controlled by a single department within an organization. Given their single-subject focus, data marts usually draw data from only a few sources. The sources could be internal operational systems, a central data warehouse, or external data.

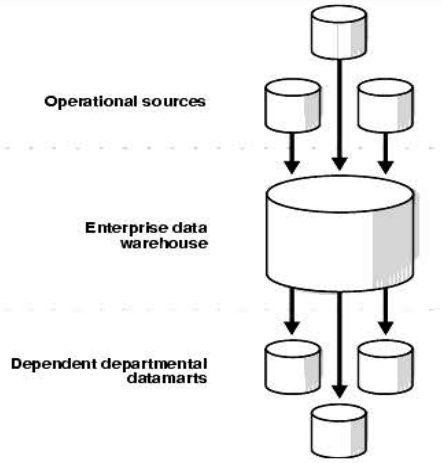
Three basic types of data marts are dependent, independent, and hybrid.

The categorization is based primarily on the data source that feeds the data mart.

- ***Dependent data marts*** draw data from a central data warehouse that has already been created.
- ***Independent data marts,*** in contrast, are standalone systems built by drawing data directly from operational or external sources of data or both.
- ***Hybrid data marts*** can draw data from operational systems or data warehouses.

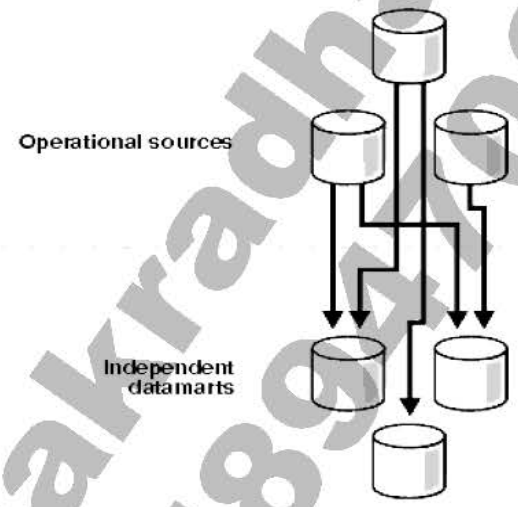
Dependent Data Marts:A dependent data mart allows you to unite your organization's data in one data warehouse.

Dependent Data Mart



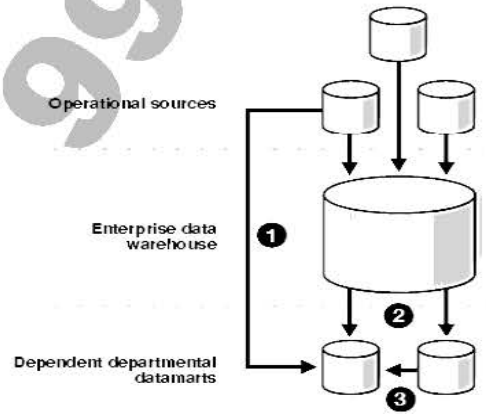
Independent Data Marts: An independent data mart is created without the use of a central data warehouse. This could be desirable for smaller groups within an organization. It is not, however, the focus of this Guide. See the *Data Mart Suites* documentation for further details regarding this architecture.

Independent Data Marts



Hybrid Data Marts:A hybrid data mart allows you to combine input from sources other than a data warehouse. This could be useful for many situations, especially when you need ad hoc integration, such as after a new group or product is added to the organization.

Hybrid Data Mart



Q. What is Hadoop and how it is different from data Warehouse?

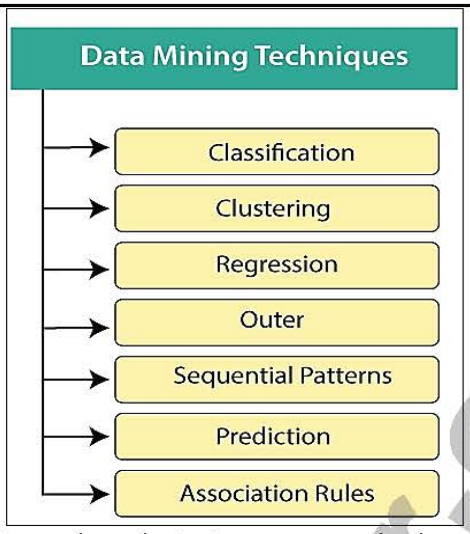
Ans. Hadoop is an open-source, a Java-based programming framework that supports the processing and storage of extremely large data sets in a distributed computing environment.
Hadoop is made up of 4 modules:

- **Distributed File-System:** Distributed File System allows data to be stored in an easily accessible format, across a large number of linked storage devices.
- **Map Reduce:**Map Reduce is the combination of two operations – reading data from the database and putting it into a format suitable for analysis (map) and performing mathematical operations (reduce).
- **Hadoop Common:**Hadoop Common provides the tools needed for the data stored in HDFS (Hadoop Distributed File System)
- **YARN:** YARN manages resources of the systems storing the data and running the analysis.

Below is the list of points describe the Comparisons Between Data Warehouse and Hadoop.

Basis For Comparison	**Data Warehouse**	**Hadoop**
Data	In Data Warehouse we analyze structured and processed data	In Hadoop, we can process any kind of data including structured/unstructured/semi-structured and raw
Processing	Its processing is based on schema-on-write concepts	Its processing is based on schema-on-read concepts
Storage	Suitable for data with small volume and it's too much expensive for large volume data	It works well with large data sets having huge volume, velocity, and variety
Agility	It is less agile and of fixed configuration	It is highly agile, configure and reconfigure as needed
Security	Data Warehouse technologies have been around for decades. Thus in term of security, we can rely on Data Warehouse	While Hadoop technologies are relatively new as compared to Data Warehouse, so security is a big concern here
Users	Business professionals usually use data warehouse	Hadoop is quite famous in the field of data science and data engineering

Q. Explain the data mining Techniques?
Ans. Data mining includes the utilization of refined data analysis tools to find previously unknown, valid patterns and relationships in huge data sets. These tools can incorporate statistical models, machine learning techniques, and mathematical algorithms, such as neural networks or decision trees. Thus, data mining incorporates analysis and prediction.



Classification: This technique is used to obtain important and relevant information about data and metadata. This data mining technique helps to classify data in different classes.

Data mining techniques can be classified by different criteria, as follows:

- Classification of Data mining frameworks as per the type of data sources mined: This classification is as per the type of data handled. For example, multimedia, spatial data, text data, time-series data, World Wide Web, and so on..
- Classification of data mining frameworks as per the database involved: This classification based on the data model involved. For example. Object-oriented database, transactional database, relational database, and so on..
- Classification of data mining frameworks as per the kind of knowledge discovered: This classification depends on the types of knowledge discovered or data mining functionalities. For example, discrimination, classification, clustering, characterization, etc. some frameworks tend to be extensive frameworks offering a few data mining functionalities together...
- Classification of data mining frameworks according to data mining techniques used: This classification is as per the data analysis approach utilized, such as neural networks, machine learning, genetic algorithms, visualization, statistics, data warehouse-oriented or database-oriented,etc.

  The classification can also take into account, the level of user interaction involved in the data mining procedure, such as query-driven systems, autonomous systems, or interactive exploratory systems.

Clustering: Clustering is a division of information into groups of connected objects. Describing the data by a few clusters mainly loses certain confine details, but accomplishes improvement. It models data by its clusters. Data modelling puts clustering from a historical point of view rooted in statistics, mathematics, and numerical analysis. From a machine learning point of view, clusters relate to hidden patterns, the search for clusters is unsupervised learning, and the subsequent framework represents a data concept. From a practical point of view, clustering plays an extraordinary job in data mining applications. For example, scientific data exploration, text mining, information retrieval, spatial database applications, CRM, Web analysis, computational biology, medical diagnostics, and much more.

In other words, we can say that Clustering analysis is a data mining technique to identify similar data. This technique helps to recognize the differences and similarities between the data. Clustering is very similar to the classification, but it involves grouping chunks of data together based on their similarities.

Regression: Regression analysis is the data mining process is used to identify and analyze the relationship between variables because of the presence of the other factor. It is used to define the probability of the specific variable. Regression, primarily a form of planning and modelling. For

example, we might use it to project certain costs, depending on other factors such as availability, consumer demand, and competition. Primarily it gives the exact relationship between two or more variables in the given data set.

Association Rules: This data mining technique helps to discover a link between two or more items. It finds a hidden pattern in the data set.

Association rules are if-then statements that support to show the probability of interactions between data items within large data sets in different types of databases. Association rule mining has several applications and is commonly used to help sales correlations in data or medical data sets.

The way the algorithm works is that you have various data, For example, a list of grocery items that you have been buying for the last six months. It calculates a percentage of items being purchased together.

These are three major measurements technique:

- **Lift:**This measurement technique measures the accuracy of the confidence over how often item B is purchased. (Confidence) / (item B)/ (Entire dataset)
- **Support:**This measurement technique measures how often multiple items are purchased and compared it to the overall dataset. (Item A + Item B) / (Entire dataset)
- **Confidence:**This measurement technique measures how often item B is purchased when item A is purchased as well. (Item A + Item B)/ (Item A)

Outer detection: This type of data mining technique relates to the observation of data items in the data set, which do not match an expected pattern or expected behaviour. This technique may be used in various domains like intrusion, detection, fraud detection, etc. It is also known as Outlier Analysis or Outlier mining. The outlier is a data point that diverges too much from the rest of the dataset. The majority of the real-world datasets have an outlier. Outlier detection plays a significant role in the data mining field. Outlier detection is valuable in numerous fields like network interruption identification, credit or debit card fraud detection, detecting outlying in wireless sensor network data, etc.

Sequential Patterns: The sequential pattern is a data mining technique specialized for evaluating sequential data to discover sequential patterns. It comprises of finding interesting subsequence's in a set of sequences, where the stake of a sequence can be measured in terms of different criteria like length, occurrence frequency, etc.

In other words, this technique of data mining helps to discover or recognize similar patterns in transaction data over some time.

Prediction: Prediction used a combination of other data mining techniques such as trends, clustering, classification, etc. It analyzes past events or instances in the right sequence to predict a future event.

Q. Briefly explain some data mining tools?

Ans. Data Mining is the set of techniques that utilize specific algorithms, statical analysis, artificial intelligence, and database systems to analyze data from different dimensions and perspectives.

Data Mining tools have the objective of discovering patterns/trends/groupings among large sets of data and transforming data into more refined information.

It is a framework, such as Rstudio or Tableau that allows you to perform different types of data mining analysis.

These are the most popular data mining tools:

1. **Range Data Mining:** Orange is a perfect machine learning and data mining software suite. It supports the visualization and is a software-based on components written in Python computing language and developed at the bioinformatics laboratory at the faculty of computer and information science, Ljubljana University, Slovenia.
   As it is a software-based on components, the components of Orange are called "widgets." These widgets range from pre-processing and data visualization to the assessment of algorithms and predictive modeling.

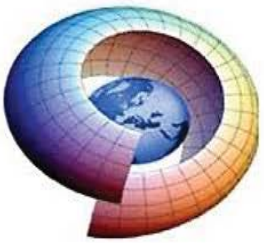
Widgets deliver significant functionalities such as:

- Displaying data table and allowing to select features
- Data reading
- Training predictors and comparison of learning algorithms
- Data element visualization, etc.

Besides, Orange provides a more interactive and enjoyable atmosphere to dull analytical tools. It is quite exciting to operate.

2. **SAS Data Mining:** SAS stands for Statistical Analysis System. It is a product of the SAS Institute created for analytics and data management. SAS can mine data, change it, manage information from various sources, and analyze statistics. It offers a graphical UI for non-technical users.
   SAS data miner allows users to analyze big data and provide accurate insight for timely decision-making purposes. SAS has distributed memory processing architecture that is highly scalable. It is suitable for data mining, optimization, and text mining purposes.

DataMelt Data Mining



3. **DataMelt Data Mining:** DataMelt is a computation and visualization environment which offers an interactive structure for data analysis and visualization. It is primarily designed for students, engineers, and scientists. It is also known as DMelt. DMelt is a multi-platform utility written in JAVA. It can run on any operating system which is compatible with JVM (Java Virtual Machine). It consists of Science and mathematics libraries.
   - **Scientific libraries:** Scientific libraries are used for drawing the 2D/3D plots.
   - **Mathematical libraries:** Mathematical libraries are used for random number generation, algorithms, curve fitting, etc.

   DMelt can be used for the analysis of the large volume of data, data mining, and statistical analysis. It is extensively used in natural sciences, financial markets, and engineering.

4. **Rattle:** Ratte is a data mining tool based on GUI. It uses the R stats programming language. Rattle exposes the statical power of R by offering significant data mining features. While rattle has a comprehensive and well-developed user interface, It has an integrated log code tab that produces duplicate code for any GUI operation.

The data set produced by Rattle can be viewed and edited. Rattle gives the other facility to review the code, use it for many purposes, and extend the code without any restriction.

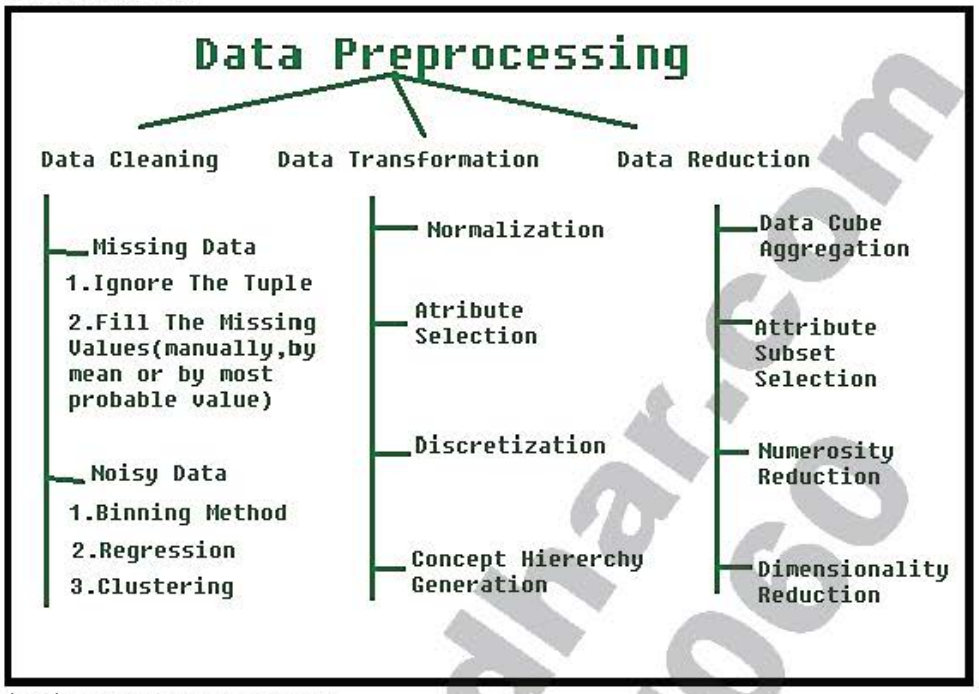
5. **Rapid Miner:** Rapid Miner is one of the most popular predictive analysis systems created by the company with the same name as the Rapid Miner. It is written in JAVA programming language. It offers an integrated environment for text mining, deep learning, machine learning, and predictive analysis.

The instrument can be used for a wide range of applications, including company applications, commercial applications, research, education, training, application development, machine learning.

Rapid Miner provides the server on-site as well as in public or private cloud infrastructure. It has a client/server model as its base. A rapid miner comes with template-based frameworks that enable fast delivery with few errors(which are commonly expected in the manual coding writing process).

Q. Explain the concept of data Preprocessing?

Ans. Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.



Steps Involved in Data Preprocessing:

1. **Data Cleaning:** The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

A. **Missing Data:** This situation arises when some data is missing in the data. It can be handled in various ways.

Some of them are:

- **Ignore the tuples:** This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.
- **Fill the Missing values:** There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

B. **Noisy Data:** Noisy data is a meaningless data that can't be interpreted by machines.It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways:

- **Binning Method:** This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.
- **Regression:** Here data can be made smooth by fitting it to a regression function.The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).
- **Clustering:** This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters.

2. **Data Transformation:** This step is taken in order to transform the data in appropriate forms suitable for mining process. This involves following ways:

- **Normalization:** It is done in order to scale the data values in a specified range (-1.0 to 1.0 or 0.0 to 1.0)
- **Attribute Selection:** In this strategy, new attributes are constructed from the given set of attributes to help the mining process.
- **Discretization:** This is done to replace the raw values of numeric attribute by interval levels or conceptual levels.

- **Concept Hierarchy Generation:** Here attributes are converted from lower level to higher level in hierarchy. For Example-The attribute "city" can be converted to "country".

3. **Data Reduction:** Since data mining is a technique that is used to handle huge amount of data. While working with huge volume of data, analysis became harder in such cases. In order to get rid of this, we uses data reduction technique. It aims to increase the storage efficiency and reduce data storage and analysis costs.

The various steps to data reduction are:

a) **Data Cube Aggregation:** Aggregation operation is applied to data for the construction of the data cube.

b) **Attribute Subset Selection:** The highly relevant attributes should be used, rest all can be discarded. For performing attribute selection, one can use level of significance and p- value of the attribute. The attribute having p-value greater than significance level can be discarded.

c) **Numerosity Reduction:** This enable to store the model of data instead of whole data, for example: Regression Models.

d) **Dimensionality Reduction:** This reduce the size of data by encoding mechanisms.It can be lossy or lossless. If after reconstruction from compressed data, original data can be retrieved, such reduction are called lossless reduction else it is called lossy reduction.

The two effective methods of dimensionality reduction are:

- Wavelet transforms.
- PCA (Principal Component Analysis).